AKADEMIA EKONOMICZNA

WE WROCŁAWIU

WYDZIAŁ GOSPODARKI NARODOWEJ

Andrzej Gospodarowicz

ZASTOSOWANIE METODY PRIORYTETÓW DO ROZWIĄZYWANIA ZAGADNIEŃ Z TEORII PRZEDSIĘWZIĘĆ CZASOWYCH

praca doktorska

Promotor:

Prof. dr hab. Zdzisław Hellwig

WROCŁAW 1975

SPIS TREŚCI

str.

WSTEP

W badaniach nad optymalizacją działania skomplikowanych systemów składających się z ludzi i urządzeń technicznych jeszcze kilkanaście lat temu przeważały analizy wykorzystujące optymalizację w zakresie związków ilościowych charakteryzujących system.
Obecnie równie ważną pozycję zaczynają zajmować optymalizacje działania takich systemów w czasie.

W ostatnim 15-leciu w obrębie badań operacyjnych kształtuje się nowa teoria, która w literaturze radzieckiej znana jest jako t e o r i a r a s p i s a n i j , zaś w literaturze anglosaskiej jako s c h e d u l i n g t h e o r y . W literaturze polskiej na określenie tej teorii stosowane są różne pojęcia, /np. teoria harmonogramowania, teoria rozkładów / . W pracy niniejszej będziemy używali terminu zaproponowanego przez Z.Hellwiga, a mianowicie t e o r i a p r z e d s i ę w z i ę ć c z a s o w y c h .
Teoria ta zrodziła się na bazie planowania produkcji. Zaczątki tej teorii znajdujemy w pracach K.Adamieckiego, który badał zależności między czynnościami w przedsięwzięciu oraz zajmował się teorią koordynacji procesów produkcyjnych za pomocą harmonogramów [1] , [2] .

Pierwsze próby ustalenia optymalnego harmonogramu produkcji dla takich odcinków produkcyjnych jakoddział lub wydział były podjęte przez B.Johnsona. W swojej pracy opublikowanej po raz pierwszy w 1954 r. przedstawił on bardzo prosty algorytm rozwiązania takiego zadania dla przypadku, kiedy odcinek produkcyjny składa się z dwóch stanowisk [79] . Próby uogólnienia algorytmu Johnsona nie przyniosły jak do tej pory pozytywnych rezultatów.

Krąg zagadnień, którymi zajmuje się teoria przedsięwzięć czasowych nie ogranicza się tylko do problemów występujących w planowaniu produkcji. Z zagadnieniami tymi spotykamy się wszędzie tam, gdzie zachodzi konieczność uporządkowania w czasie pewnych czynności wykonywanych przy pomocy ograniczonego zbioru wykonawców i środków. Przykładowo można tu wymienić takie zagadnienia jak: układanie rozkładu jazdy pociągów, układanie harmonogramu zajęć szkolnych, układanie planu studiów dla wyższej uczelni, planowanie urlopów itp. Ponadto z zagadnieniami tymi spotykamy się często w wojsku. Zagadnienia rozważane w teorii przedsięwzięć czasowych mają charakter kombinatoryczny i w każdym konkretnym przypadku zadanym warunkom i ograniczeniom może odpowiadać bardzo duża liczba rozwiązań dopuszczalnych.

Istniejące metody rozwiązania tego typu zagadnień ze względu na dokładność otrzymanego rozwiązania można podzielić na dwie grupy:

- metody za pomocą których znajdujemy rozwiązanie optymalne ze względu na określone kryterium,
- metody za pomocą których otrzymujemy rozwiązanie przybliżone.

Metody należące do pierwszej grupy mają zasadniczą wadę. Przy ich pomocy można rozwiązywać niewielkie zadania, które najczęściej nie znajdują odzwierciedlenia w praktyce. Zadania spotykane w praktyce charakteryzują się dużą ilością zmiennych i ograniczeń /tzw. zadania o dużych rozmiarach/. W chwili obecnej takie zadanie można rozwiązać jedynie przy pomocy metod przybliżonych.

Głównym celem pracy jest podanie schematu postępowania zapewniającego generowanie rozwiązań przy pomocy komputera dla szerokiej klasy zadań o dużych rozmiarach. Praca składa się z sześciu rozdziałów oraz Aneksu.

W rozdziale pierwszym podjęto próbę scharakteryzowania w kategoriach abstrakcyjnych podstawowego problemu będącego przedmiotem rozważań teorii przedsięwzięć czasowych. Ze względu na różnorodność zadań rozpatrywanych w teorii przedsięwzięć czasowych zaproponowano dokonanie podziału tych zadań biorąc za podstawę rodzaj tzw. ograniczeń strukturalnych. Uwzględniając trzy rodzaje ograniczeń strukturalnych omawiane zadania podzielono na:

- zadania z szeregową kolejnością czynności,
- zadania z szeregowo-równoległą kolejnością czynności,
- zadania z częściową kolejnością czynności.

W rozdziale pierwszym dokonano także analizy sposobu rozwiązania zadań o dużych rozmiarach.
Wyróżniono przy tym metody heurystyczne, proponując do rozwiązywania zadań o dużych rozmiarach pewną metodę heurystyczną, a właściwie pewien schemat postępowania nazwany w tej pracy m e t o d ą p r i o r y t e t ó w.

Proponowana metoda priorytetów korzysta z metod tzw. programowania heurystycznego. Programowanie heurystyczne nie posiada w literaturze polskiej żadnych opracowań monograficznych. Stąd też ważniejsze pojęcia tego programowania omówiono w rozdziale drugim.

Centralne zagadnienia w metodzie proponowanej w pracy stanowią: problem konfliktów oraz problem priorytetu. Zagadnienia te szczegółowo omówione są w rozdziale trzecim.

Trzy dalsze rozdziały poświęcone są zastosowaniu metody priorytetów do rozwiązania podstawowych zadań teorii przedsięwzięć czasowych.

Zastosowanie tej metody do ustalania harmonogramu produkcji wydziału obróbczego lub przygotowawczego przedstawiono w rozdziale czwartym, zaś w rozdziale piątym zastosowano ją do ustalania harmonogramu produkcji dla wydziału montażowego.

W rozdziale szóstym przedstawiono zastosowanie metody priorytetów do układania harmonogramu zajęć w szkole wyższej. Rozdział ten zawiera także opis eksperymentów komputerowych prowadzonych na realnych danych z Akademii Ekonomicznej we Wrocławiu w celu weryfikacji zaproponowanego algorytmu.

Formułując zadania omawiane w rozdziale czwartym i piątym starano się zwrócić uwagę na istotne ograniczenia występujące w zadaniach spotykanych w praktyce. Bardzo pomocne w tym względzie oraz przy konstruowaniu algorytmów było zapoznanie się z problemami praktycznymi w Jelczańskich Zakładach Samochodowych oraz uczestnictwo w pracach Zespołu

Integracji kierowanego przez prof.dr Z.Hellwiga. Zespół ten wraz z innymi Zespołami Zakładów Naukowo-Badawczych Akademii Ekonomicznej we Wrocławiu realizuje na zlecenie "Mera-Elwro" system "KOSIP".

Aneks zawiera zapis w języku FORTRAN algorytmów realizowanych na komputerze Odra 1305, fragmenty wydruków oraz wzory dokumentów dla zapisu danych wejściowych.

Formalnie cała praca dzieli się na rozdziały, które dzielą się na paragrafy, a te z kolei na punkty. Numeracja wzorów dokonywana jest w sposób ciągły w całej pracy. Pozycje literatury numerowane są w nawiasach kwadratowych i posiadają numerację ciągłą dla całej pracy.

1. WPROWADZENIE DO TEORII PRZEDSIĘWZIĘĆ CZASOWYCH

1.1. Uwagi ogólne

Sformułowanie przedmiotu i metod teorii przedsięwzięć czasowych zostało podjęte po raz pierwszy przez Bellmana w roku 1956 /por. [22] /. Wraz z rozwojem metod z teorii przedsięwzięć czasowych następne próby w tym zakresie były podejmowane m.in. przez Szkurbę [176] , Conway'a, Maxwella i Millera [41] , Moisiejewa [134] oraz Siemionowa i Portugala [164] . W ostatnim okresie teoria przedsięwzięć czasowych rozwija się bardzo intensywnie. Formułowane są nowe zadania /z reguły bardziej złożone/ i proponowane są metody ich rozwiązania /najczęściej przybliżone/ [35] , [45] , [52] , [64] , [132] , [135] , [161] , [171] . Na obecnym etapie rozwoju teorii przedsięwzięć czasowych nie wydaje się już możliwe omówienie wszystkich zadań stanowiących przedmiot rozważań teorii. Właśnie dlatego w pracy podjęto próbę podziału zadań teorii przedsięwzięć czasowych na trzy grupy a następnie w każdej grupie omówiono zadania najbardziej typowe.

Formułując ogólną problematykę omawianej teorii można by, jak to czynią niektórzy [35] , [64] , używać terminów i określeń zaczerpniętych z planowania produkcji. Wydaje się jednak, że postępowanie takie prowadzi do utożsamiania tej teorii tylko z zagadnieniami planowania produkcji, co jak wiadomo nie jest słuszne.

Przedstawione poniżej ważniejsze problemy teorii przedsięwzięć czasowych zostały opisane w kategoriach abstrakcyjnych, które łatwo można przetransponować na język konkretnych zastosowań.

1.2. O p i s p o d s t a w o w e g o p r o b l e m u

1.2.1. Ogólne sformułowanie

Niech będą dane następujące zbiory skończone: zbiór B o liczebności m_1 zwany w dalszym ciągu zbiorem o b i e k t ó w, zbiór D o liczebności m_3 zwany dalej zbiorem c z y n n o ś c i oraz zbiór Z o liczebności m_2 zwany zbiorem z a s o b ó w .
Elementy zbioru B ponumerujemy w dowolny sposób kolejnymi liczbami naturalnymi od 1 do m_1, oznaczając je symbolem b_i, natomiast elementy zbioru Z liczbami od 1 do m_2 oznaczając poszczególne elementy symbolem z_k.
Elementom $z_k \in Z$ są przyporządkowane pewne dodatnie liczby rzeczywiste $o_k(t)$ zwane dysponowaną ilością k-tego zasobu w chwili t.
Każdemu elementowi $b_i \in B$ przyporządkowany jest z kolei pewien podzbiór zbioru D. Podzbiór ten oznaczać będziemy symbolem D_i a jego liczebność symbolem n_i. Podobnie jak w przypadku zbiorów B oraz Z elementy zbioru D_i $/i=1,2,\ldots,m_1/$ ponumerujemy liczbami naturalnymi od 1 do n_i i oznaczymy symbolem d_{ij}, gdzie d_{ij} oznacza j-tą czynność należącą do zbioru D_i. Zbiory D_i $/i=1,2,\ldots,m_1/$

nie muszą być rozłączne ale musi być spełniony warunek taki, że

$$D = D_1 \cup D_2 \cup \dots \cup D_{m_1}$$

O fakcie przyporządkowania każdemu obiektowi b_i pewnego podzbioru czynności D_i mówi się w języku profesjonalnym, że dla obiektu wykonywane są pewne kompleksy czynności. Zorganizowane działanie zmierzające do osiągnięcia określonego celu realizowane na zbiorze obiektów poprzez wykonywanie dla nich pewnych k o m p l e k s ó w c z y n n o ś c i jest nazywane p r z e d s i ę w z i ę c i e m [123]. Każdemu elementowi $d_{ij} \in D_i$ /$i=1,2,\dots,m_1$/ przyporządkowana jest pewna wielkość t_{ij} zwana c z a s e m t r w a n i a oraz wielkość $q_{ijk}(t)$ zwana p o t r z e b n ą i l o ś c i ą k - t e g o z a s o b u w c h w i l i t. Wielkości t_{ij} oraz $q_{ijk}(t)$ będą też nazywane w dalszym ciągu c h a r a k t e r y s t y k a m i c z y n n o ś c i d_{ij}. Każdej czynności $d_{ij} \in D_i$ /$i=1,2,\dots,m_1$/ można przyporządkować pewien wektor X

$$X = (x_1, x_2, \dots, x_z) \tag{1}$$

gdzie poszczególne składowe tego wektora to charakterystyki czynności d_{ij}; tak np. $x_1 = t_{ij}$, $x_2 = q_{ijk}(t)$, x_3 - uciążliwość czynności mierzona w określonych jednostkach itd.

Na zbiorze D_i /$i=1,2,\dots,m_1$/ zadana jest pewna relacja $R \subset D_i \times D_i$, zwana dalej r e l a c j ą p o p r z e d z a n i a. Jeżeli $d_{ij_1}, d_{ij_2} \in D_i$ są w relacji R /symbolicznie piszemy wówczas $\langle d_{ij_1}, d_{ij_2} \rangle \in R$ lub $d_{ij_1} \; R \; d_{ij_2}$ / wtedy mówimy, że czynność d_{ij_1} poprzedza czynność d_{ij_2}.

Zapis symboliczny $d_{ij_1} \; R \; d_{ij_2}$ interpretuje się następująco: "wykonanie czynności d_{ij_1} poprzedza w czasie wykonanie czynności d_{ij_2}". Relacja R posiada następujące własności:

a/ przeciwzwrotność tzn. dla każdego $d_{ij} \in D_i$ $\neg(d_{ij} \; R \; d_{ij})$

b/ przeciwsymetrię tzn. dla każdych dowolnych $d_{ij_1}, d_{ij_2} \in D$
jeżeli $d_{ij_1} \; R \; d_{ij_2}$ wtedy $\neg(d_{ij_2} \; R \; d_{ij_1})$

c/ przechodniość tzn. dla każdych $d_{ij_1}, d_{ij_2}, d_{ij_3} \in D_i$
jeżeli $d_{ij_1} \; R \; d_{ij_2}$ i $d_{ij_2} \; R \; d_{ij_3}$ to $d_{ij_1} \; R \; d_{ij_3}$.

Poniżej wyjaśnimy na czym polega programowanie przedsięwzięć. Realizacja przedsięwzięcia wymaga wypełnienia pewnych ograniczeń, które podzielimy na ograniczenia podstawowe i ograniczenia dodatkowe.

Ograniczenia podstawowe występują we wszystkich zadaniach będących przedmiotem rozważań teorii przedsięwzięć czasowych, natomiast ograniczenia dodatkowe dotyczą konkretnych zadań.

Do podstawowych ograniczeń zalicza się dwa ograniczenia. Pierwsze zwane jest w literaturze ograniczeniem b i l a n s o w y m , drugie o g r a n i c z e n i e m s t r u k t u r a l n y m [56] .

Zanim sformułujemy ograniczenie bilansowe przyjmiemy definicję oraz wprowadzimy pewne oznaczenia.

Powiemy, że czynność $d_{ij} \in D_i$ $/i=1,2,\ldots,m_1/$ o d b y w a s i ę w c z a s i e t, jeżeli dla d_{ij} zachodzi następujący warunek: $t^o_{ij} \leq t \leq t^1_{ij}$, gdzie symbolem t^o_{ij} oznaczać będziemy moment rozpoczęcia czynności d_{ij} natomiast symbolem t^1_{ij} moment zakończenia czynności d_{ij}, przy czym wielkość t^1_{ij} jest wyliczana następująco:

$t^1_{ij} = t^o_{ij} + t_{ij}$.

Symbolem t^r oznaczymy p o c z ą t k o w y m o m e n r o z p o c z ę c i a c z y n n o ś c i należących do zbiorów D_i $/i=1,2,\ldots,m_1/$

$$t^r = \min_{i,j} t^o_{ij} \qquad i=1,2,\ldots,m_1 \;,\; j=1,2,\ldots,n_i \qquad (2)$$

Symbolem t^z oznaczymy natomiast o s t a t e c z n y m o m e n t z a k o ń c z e n i a c z y n n o ś c i należących do zbiorów D_i $/i=1,2,\ldots,m_1/$

$$t^z = \max_{i,j} t^1_{ij} \qquad i=1,2,\ldots,m_1, \;\; j=1,2,\ldots,n_i \qquad (3)$$

Tak jak zapowiedziano wcześniej zdefiniujemy obecnie ograniczenie bilansowe.

$$\sum_{i,j} q_{ijk}(t) \leq c_k(t) \qquad \text{dla} \quad t^r \leq t \leq t^z \qquad (4)$$
$$k=1,2,\ldots,m_2$$

gdzie sumowanie odbywa się po tych wskaźnikach i oraz j, które spełniają następujący warunek: czynność d_{ij} odbywa się w czasie t oraz jest jej przyporządkowany k-ty zasób.
Ograniczenie bilansowe bywa interpretowane następująco: wielkości t_{ij}^{o} muszą być ustalone w ten sposób aby zapotrzebowanie na dany zasób w każdym momencie t / $t^{r} \leq t \leq t^{z}$/ nie przekraczało dysponowanej ilości zasobu.
Ograniczenie strukturalne można sformułować następująco:
Dla czynności $d_{ij} \in D_i$ /$i=1,2,\ldots,m_1$/ moment rozpoczęcia musi być równy lub większy od momentów zakończenia wszystkich tych czynności, które są z czynnością d_{ij} w relacji R czyli inaczej od momentów zakończenia wszystkich czynności, które poprzedzają czynność d_{ij}. Wśród mnogości ograniczeń dodatkowych zwrócimy uwagę na niektóre z nich. Będziemy je numerować kolejnymi liczbami naturalnymi.

1/ $c_k(t) = \text{const.}$ dla $k = 1,2,\ldots,m_2$

2/ $q_{ijk}(t) = \text{const.}$ dla $t_{ij}^{o} \leq t \leq t_{ij}^{1}$

W dalszych rozważaniach przyjmujemy, że te dwa ograniczenia są zadane.

3/ Dla wszystkich lub niektórych czynności $d_{ij} \in D_i$ /$i=1,2,\ldots,m_1$/ podana jest pewna wielkość τ_{ij}^{o}, od której wielkość t_{ij}^{o} musi być większa lub równa.

4/ Dla wszystkich lub niektórych czynności $d_{ij} \in D_i$ /$i=1,2,\ldots,m_1$/ podana jest wielkość τ_{ij}^{1}, której wielkość t_{ij}^{1} nie może przekroczyć.

Programowanie przedsięwzięć polega na tym, aby każdej czynności $d_{ij} \in D_i$ $/i=1,2,\dots,m_1/$ przyporządkować wielkość t^o_{ij} przy czym wielkość t^o_{ij} musi spełniać wszystkie ograniczenia tak podstawowe jak i dodatkowe. Wielkości t^o_{ij} można zapisać w macierzy T^o

$$T^o = [t^o_{ij}] \qquad \begin{array}{l} i=1,2,\dots,m_1 \\ j=1,2,\dots,m_3 \end{array} \qquad (5)$$

$$\text{gdzie} \quad m_3 = \max_i \; n_i$$

przy czym $t^o_{ij} \geqslant 0$ dla $i=1,2,\dots,m_1$ oraz $1 \leqslant j \leqslant n_i$

$t^o_{ij} = -1$ dla $i=1,2,\dots,m_1$ oraz $n_i < j \leqslant m_3$

Macierz T^o spełniającą wszystkie ograniczenia podstawowe i dodatkowe będziemy nazywali *rozwiązaniem dopuszczalnym* lub *harmonogramem czynności*. Przy zadanych ograniczeniach /szczególnie dotyczy to ograniczeń dodatkowych podanych w p.3 oraz 4/ może nie istnieć rozwiązanie dopuszczalne. Jeżeli rozwiązanie dopuszczalne istnieje wtedy należy znaleźć taką macierz T^o, która maksymizowała by zadaną funkcję F zwaną *funkcją-kryterium*. Macierz T^o będzie się wtedy nazywała *rozwiązaniem optymalnym* lub *harmonogramem optymalnym*.

Tak więc można powiedzieć, że programowanie przedsięwzięć polega na rozłożeniu w *czasie* czynności należących

do zbiorów D_i $/i=1,2,\dots,m_1/$ w ten sposób, aby były spełnione wszystkie ograniczenia oraz aby wartość funkcji kryterium osiągnęła ekstremum.
Określając dla poszczególnych czynności momenty ich rozpoczęcia organizujemy w ten sposób pewne działanie zmierzające do osiągnięcia określonego celu czyli inaczej organizujemy pewne przedsięwzięcie. Sformułowanie zadań występujących podczas organizowania przedsięwzięcia oraz metody ich rozwiązania stanowią przedmiot zainteresowania teorii przedsięwzięć czasowych.
Aby zaakcentować, że centralną wielkością w tej teorii jest czas, przyjęto zgodnie z sugestią Z.Hellwiga, że termin przedsięwzięcie będzie nazywany w tej pracy przedsięwzięciem czasowym.

1.2.2. Klasyfikacja zagadnień

W każdym konkretnym zadaniu relacja R zadana na zbiorze D_i $/i=1,2,\dots,m_1/$ oprócz własności wymienionych wcześniej może posiadać szereg dodatkowych własności. Wśród własności dodatkowych można wyróżnić takie trzy własności, z których każda jest charakterystyczna dla dużej grupy zagadnień.

W pierwszej grupie zagadnień relacja R ma własność następującą: dla każdego elementu $d_{ij} \in D_i$ $/i=1,2,\dots,m_1/$ istnieje przynajmniej jeden element $d'_{ij} \in D_i$ taki, że $d'_{ij} \neq d_{ij}$ oraz zachodzi relacja $d_{ij}\, R\, d'_{ij}$ lub $d'_{ij}\, R\, d_{ij}$.

W drugiej grupie relacja R ma własność spójności tzn. dla dowolnych $d_{ij_1}, d_{ij_2} \in D_i$ $/i=1,2,\ldots,m_1/$ jeżeli $d_{ij_1} \neq d_{ij_2}$ to d_{ij_1} R d_{ij_2} lub d_{ij_2} R d_{ij_1}. Widać tu wyraźnie, że spełniona jest jednocześnie własność charakterystyczna dla pierwszej grupy.

Trzecia grupa charakteryzuje się tym, że w zbiorze D_i $/i=1,2,\ldots,m_1/$ można znaleźć takie elementy $d_{ij} \in D_i$, dla których własność pierwszej grupy nie jest spełniona. Elementy takie będziemy nazywać czynnościami wyizolowanymi.

W każdej z podanych trzech grup zagadnień relacja R ustala w zbiorze D_i $/i=1,2,\ldots,m_1/$ pewien porządek czyli pewną kolejność. Będziemy mówili, że w pierwszej grupie zagadnień w zbiorze D_i $/ i=1,2,\ldots,m_1/$ zadany jest porządek B, w drugiej grupie porządek A oraz w trzeciej grupie porządek C.

Jeżeli w zbiorze D_i składającym się z n_i elementów zadany jest porządek A wtedy zbiór ten jest izomorficzny /co do porządku/ z liczbami $1,2,\ldots,n_i$. Powiemy też wówczas, że kolejność wykonywania czynności należących do zbioru D_i jest szeregowa tzn. w każdym momencie t może być wykonywana tylko jedna czynność ze zbioru D_i. Gdy w pewnym zbiorze D_i zadany jest porządek B wtedy można powiedzieć, że w zbiorze D_i istnieją pewne podzbiory czynności, które należy wykonywać szeregowo, jednak w każdym momencie t może być wykonana więcej jak jedna czynność ze zbioru D_i. Można więc mówić, że czynności są wykonywane w kolejności szeregowo-równoległej. Porządek C zadany w pewnym zbiorze D_i tym się różni od porządku B, że w zbiorze tym istnieją czynności wyizolowane. W tym przypadku będziemy mówili, że w zbiorze D_i zadana jest częściowa kolejność.

Powiemy, że:

1/ zadanie będzie należeć do **zadań z szeregową kolejnością czynności** zwanych też w dalszym ciągu **zadaniami A** wtedy, gdy mamy do czynienia z zadaniem, w którym w zbiorze D_i $/i=1,2,\ldots,m_1/$ zadany jest porządek A,

2/ zadanie będzie należeć do **zadania z szeregowo-równoległą kolejnością czynności** zwanych też w dalszym ciągu **zadaniami B** wtedy, gdy przedmiotem rozważań jest zadanie, w którym w zbiorze D_i $/i=1,2,\ldots,m_1/$ zadany jest porządek B,

3/ zadanie będzie należeć do **zadań z częściową kolejnością czynności** zwanych w dalszym ciągu **zadaniami C** wtedy, gdy przedmiotem rozważań jest zadanie, w którym w zbiorze D_i $/i=1,2,\ldots,m_1/$ zadany jest porządek C.

Powyżej rozpatrywane były zadania, w których we wszystkich zbiorach D_i zadany jest jednakowy porządek czynności.
Teoretycznie może wystąpić zadanie, w którym w pewnych zbiorach D_i jest zadany np. porządek A zaś w pozostałych zbiorach D_i porządek B.

Zauważmy, że porządek A jest szczególnym przypadkiem porządku B, a z kolei porządek B szczególnym przypadkiem porządku C. Ogólnie można przyjąć, że z zadaniem A mamy do czynienia wtedy, gdy w każdym zbiorze D_i $/i=1,2,\ldots,m_1/$

zadany jest porządek A. O zadaniu B można mówić wtedy, gdy wśród zbiorów D_i / $i=1,2,\ldots,m_1$/ wystąpi przynajmniej jeden zbiór, w którym zadany jest porządek B a w pozostałych porządek A, natomiast o zadaniu C mówimy wtedy, gdy wśród zbiorów D_i /$i=1,2,\ldots,m_1$/ wystąpi przynajmniej jeden zbiór, w którym zadany jest porządek C, w pozostałych natomiast może wystąpić zarówno porządek A jak i porządek B.

1.2.3. Interpretacja zagadnień

Dla jaśniejszego przedstawienia trzech grup zagadnień, porządek w zbiorze D_i, a więc kryterium podziału tych zagadnień, zostanie opisany w języku teorii grafów.

Jak wiadomo zbiór łuków grafu stanowi graficzny obraz zbioru par uporządkowanych iloczynu kartezjańskiego pewnego zbioru elementów spełniających zadaną relację a zbiór wierzchołków grafu jest obrazem graficznym elementów rozpatrywanego zbioru. Graf S, w którym zbiór wierzchołków jest np. obrazem graficznym elementów zbioru D_i a zbiór łuków jest obrazem zbioru par uporządkowanych spełniających relację R będziemy oznaczać symbolem $S(D_i,R)$.
Na podstawie własności "a", "b", "c" można powiedzieć, że zbiór D_i /$i=1,2,\ldots,m_1$/ wraz z zadaną na nim relacją R określa klasę grafów skończonych skierowanych oraz nie zawierających cykli. W związku z tym, że w każdej z trzech wcześniej podanych grup zagadnień relacja R posiada dodatkowo inne własności, w każdej z tych grup obrazem graficznym zbioru D_i /$i=1,2,\ldots,m_1$/ oraz relacji R będzie inny graf

należący jednak do tej klasy grafów.

Dla wygody i jasności dalszych rozważań przy definiowaniu grafu będziemy wykorzystywali relację bezpośredniego poprzedzania $R^{*} \subset R$. Będziemy mówili, że $d_{ij_1}, d_{ij_2} \in D_i$ $/i=1,2,\dots,m_1/$ są w relacji R^{*} jeżeli:

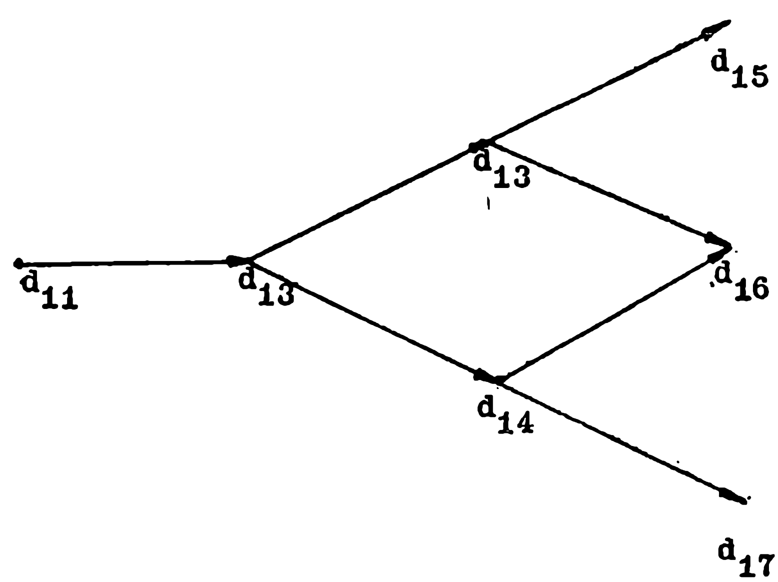
- $d_{ij_1} \; R \; d_{ij_2}$
- nie istnieje takie $d_{ij} \in D_i$ $/i=1,2,\dots,m_1/$, gdzie $d_{ij} \neq d_{ij_1}$ i $d_{ij} \neq d_{ij_2}$, dla którego $d_{ij} \; R \; d_{ij_2}$ oraz $d_{ij_1} \; R \; d_{ij}$.

Jak wiadomo relacja R posiada własność przechodniości. Tak więc jeżeli tylko są pary uporządkowane, czyli w grafie $S(D_i,R)$ łuki $\langle d_{ij_1}, d_{ij_2} \rangle$ i $\langle d_{ij_2}, d_{ij_3} \rangle$, to musi wystąpić łuk wypadkowy $\langle d_{ij_1}, d_{ij_3} \rangle$. Ogólnie, jeżeli można utworzyć następujący ciąg par uporządkowanych $\langle d_{ij_1}, d_{ij_2} \rangle$, $\langle d_{ij_2}, d_{ij_3} \rangle$, ..., $\langle d_{ij_{s-2}}, d_{ij_{s-1}} \rangle$ $\langle d_{ij_{s-1}}, d_{ij_s} \rangle$, to musi wystąpić łuk wypadkowy $\langle d_{ij_1}, d_{ij_s} \rangle$. O takim ciągu par uporządkowanych mówi się, że w grafie $S(D_i,R)$ z d_{ij_1} do d_{ij_2} istnieje ścieżka.

Definiując graf przy pomocy relacji R^{*} ograniczamy znacznie ilość połączeń między wierzchołkami czyli łuków, ponieważ redukujemy łuki wypadkowe. Łatwo zauważyć, że dwa dowolne elementy $d_{ij_1}, d_{ij_s} \in D_i$ są w relacji R jeżeli w grafie $S(D_i,R^{*})$ istnieje ścieżka z d_{ij_1} do d_{ij_s}

W drugiej z wymienionych wyżej grup problemów relację R określa graf zwany siecią. Dla dowolnego zbioru D_i $/i=1,2,\ldots,m_1/$ pojęcie grafu zwanego siecią zdefiniujemy następująco [8] .

Siecią nazywa się graf skończony $S(D_i,R^*)$ bez pętli i cykli, zorientowany w jednym kierunku od wierzchołków zwanych wejściami grafu do wierzchołków zwanych wyjściami. Wierzchołek stanowiący wejście grafu jest to taki wierzchołek, dla którego półstopień wejścia czyli liczba łuków wchodzących do tego wierzchołka jest równa zeru. Wierzchołek stanowiący wyjście to taki, dla którego półstopień wyjścia czyli liczba łuków wychodzących z tego wierzchołka równa się zeru. Zauważmy, że zgodnie z podaną definicją szczególnym przypadkiem sieci jest sieć mająca jedno wejście i jedno wyjście. Przykład sieci dla relacji R zadanej na siedmioelementowym zbiorze $D_1 = \{d_{11}, d_{12}, d_{13}, d_{14}, d_{15}, d_{16}, d_{17}\}$ przedstawiony jest na rys. 1

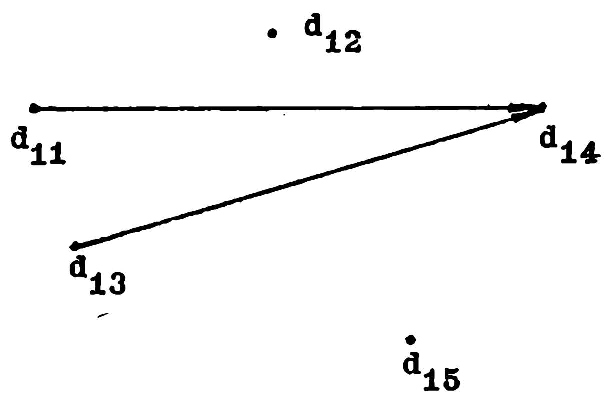


Rys.1

Gdy mamy do czynienia z pierwszą grupą zagadnień, relacja R ma dodatkowo własność spójności czyli dla każdych $d_{ij_1}, d_{ij_2} \in D_i$ zachodzi d_{ij_1} R d_{ij_2} lub d_{ij_2} R d_{ij_1}. Powiemy wtedy, że wszystkie elementy zbioru D_i /$i=1,2,\ldots,m$/ mogą być "porównywane" za pomocą tej relacji. Zbiór taki zawiera pewien element np. d_{i_1}, który poprzedza wszystkie pozostałe elementy / będziemy go nazywali elementem najwcześniejszym/, następny element np. d_{i_2}, który poprzedza wszystkie elementy oprócz d_{i_1} oraz d_{i_2} itd. aż do elementu, który nie poprzedza żadnego elementu /element ostatni/. Graf będący obrazem graficznym takiej relacji zadanej na zbiorze D_i /$i=1,2,\ldots,m_i$/ jest siecią mającą jedno wejście /stanowi je element najwcześniejszy/ oraz jedno wyjście /element ostatni/ oraz w grafie tym od wierzchołka stanowiącego wejście grafu do wierzchołka stanowiącego wyjście grafu istnieje tylko jedna ścieżka, przy czym do ścieżki tej należą wszystkie wierzchołki grafu. Graf taki będziemy nazywali siecią liniową. Natomiast siecią będziemy nazywali taką sieć, która nie jest siecią liniową. Na rys. 2 przedstawiono pewną sieć liniową dla relacji zadanej na zbiorze
$D_1 = \{d_{11}, d_{12}, d_{13}, d_{14}\}$

Rys.2

W przypadku gdy przedmiotem rozważań są zagadnienia należące do trzeciej grupy wówczas relację R można przedstawić za pomocą grafu skończonego skierowanego, nie zawierającego cykli oraz posiadającego wierzchołki izolowane tj. takie, które nie są połączone łukiem z innym wierzchołkiem. Graf taki będziemy nazywali grafem swobodnym. Przykład grafu swobodnego dla relacji R zadanej na zbiorze $D_1 = \{d_{11}, d_{12}, d_{13}, d_{14}, d_{15}\}$ ilustruje rys.3



Rys.3

Przyjmując, że rozpatrujemy zadania, w których porządek w każdym zbiorze D_i jest jednakowy, powiemy że z zadaniem A mamy do czynienia wtedy, gdy porządek w zbiorze D_i /$i=1,2,\dots,m_1$/ można przedstawić za pomocą sieci liniowej, z zadaniem B gdy porządek przedstawiony został za pomocą sieci oraz z zadaniem C gdy porządek przedstawiono za pomocą grafu swobodnego.

1.3. Sposób rozwiązania

Opisany wyżej ogólny problem, którym zajmuje się teoria przedsięwzięć czasowych jest prosty w sformułowaniu, nie jest jednak prosty do rozwiązania. Problem ma charakter kombinatoryczny i przy zadanych ograniczeniach może istnieć bardzo duża ale skończona liczba rozwiązań, spośród których należy wybrać takie, dla którego pewna funkcja kryterium F osiąga ekstremum.

Należy jednak podkreślić, że zadania z teorii przedsięwzięć czasowych są typowymi zadaniami wieloekstremalnymi [27], [125]. W zadaniach takich mamy jak wiadomo do czynienia z szeregiem lokalnych ekstremów funkcji F. Znaleziona macierz T^o będzie wtedy rozwiązaniem optymalnym, gdy wartość funkcji $F(T^o)$ osiąga ekstremum globalne.

Większość zadań z teorii przedsięwzięć czasowych należałoby rozpatrywać z punktu widzenia kilku funkcji kryterium i wyznaczać rozwiązanie "godzące" je[1]. Takie podejście komplikuje najczęściej strukturę matematyczną problemu, która w przypadku omawianych zadań jest jak wiadomo bardzo złożona. W dostępnej literaturze przedmiotu realne zadania z teorii przedsięwzięć czasowych bada się pod kątem jednej funkcji kryterium, pozostałe traktując jako ograniczenie.

Radziecki matematyk N.Mojisiejew bardzo trafnie zauważył, że główną trudność przy budowie algorytmów obliczeniowych stanowi fakt niepełnego uporządkowania zbioru możliwych

---

[1] Teoretyczne problemy optymalizacji wielokryteriowej były przedmiotem pracy doktorskiej przygotowanej w Instytucie Cybernetyki Ekonomicznej przez E.Konarzewską pod kierunkiem W.Bukietyńskiego

przyporządkowań wielkości t_{ij}^{0} poszczególnym czynnościom [134 str. 171] .

Ogólnie, w literaturze, o zadaniach z teorii przedsięwzięć czasowych mówi się, że są to ekstremalne zadania o charakterze kombinatorycznym, zwane też krótko zadaniami kombinatorycznymi [104] . Zadania kombinatoryczne stanowią klasę zadań programowania dyskretnego i oprócz omawianego problemu zaliczyć do nich można m.in. zagadnienie przydziału, zagadnienie komiwojażera, zagadnienie pokrycia. Do rozwiązywania zadań teorii przedsięwzięć czasowych znajdują więc zastosowanie metody wypracowane w teorii programowania dyskretnego.

W literaturze spotyka się różnorodność metod rozwiązywania konkretnych zadań rozpatrywanych w teorii programowania dyskretnego. Z tego względu bardzo często podejmowane są próby klasyfikacji tych metod [18] , [36] , [104] , [114] , [125]

W pracy niniejszej przyjmiemy klasyfikację zaproponowaną przez Korbuta i Finkelsztejna w monografii [104] .

Metody służące do rozwiązywania zadań omawianych w niniejszej pracy dzielą się na:

1° metody płaszczyzn odcinających /metody odcięcia/,

2° metody kombinatoryczne,

3° metody przybliżone.

Metody należące do pierwszej grupy wykorzystują programowanie liniowe. Rozwiązywanie zadań przy pomocy tych metod polega na znalezieniu ekstremum całkowitoliczbowego funkcji liniowej zadanej na skończonym zbiorze elementów. Odbywa się to w ten sposób, że w miejsce zbioru dyskretnego, stanowiącego wyjściowy obszar rozwiązań dopuszczalnych, przyjmuje się zbiór wypukły tzn. nie uwzględnia się warunku dyskretności. Dla otrzymanego w ten sposób zadania szuka się ekstremum funkcji liniowej. Jeżeli otrzymane rozwiązanie jest całkowitoliczbowe /tzn. spełnia warunki dyskretności/ - zadanie jest rozwiązane. W przeciwnym wypadku należy dokonać tzw. przejścia do rozwiązania dyskretnego. Przejście to stanowi istotę metod pierwszej grupy i polega na dołączeniu nowych ograniczeń liniowych spełniających dwa następujące warunki [104] :

- otrzymane rozwiązanie niecałkowitoliczbowe nie spełnia tego ograniczenia,
- z góry wiadomo, że spełnia go dowolne rozwiązanie całkowitoliczbowe.

Ograniczenia takie dołączane są kolejno i za każdym razem rozwiązywane jest zadanie. Proces obliczeń przerywany jest w chwili gdy rozwiązanie spełnia warunki całkowitoliczbowości.

Szczególnie duże zasługi w rozwoju tych metod ma R.E. Gomory, który jako pierwszy w 1958 r. podał ideę formułowania ograniczeń uzupełniających oraz opracował pierwszy algorytm rozwiązywania pewnej grupy zadań z teorii przedsięwzięć czasowych [95] .

Powstaje pytanie o efektywności algorytmów, w których znajdują zastosowanie metody płaszczyzn odcinających. Najlepszą odpowiedzią na to pytanie byłoby podanie wzoru określającego liczbę operacji arytmetycznych niezbędnych do rozwiązania zadania w zależności od parametrów zadania. Jak dotychczas nikt nie podał takiego wzoru nawet dla zadań

programowania liniowego bez warunku dyskretności. Wszystkie rozważania o efektywności tych algorytmów bazują na wynikach tych eksperymentów na maszynie cyfrowej.
Ogólnie można powiedzieć, że przy stosowaniu omawianych metod notuje się zarówno sukcesy jak i niepowodzenia. Do zadań dla których zastosowanie metod płaszczyzn odcinających nie daje pozytywnych wyników należą zadania z teorii przedsięwzięć czasowych [104] .
Jak stwierdzają Korbut i Finkelsztejn formułowanie tych zadań w języku programowania liniowego w liczbach całkowitych jest nienaturalne, ponieważ dla stosunkowo niewielkiego zadania występuje ogromna liczba ograniczeń i zmiennych. Przy pomocy tych metod można rozwiązywać zadania średniego rozmiaru tj. zawierające dziesiątki zmiennych i kilkanaście ograniczeń.

Celem niniejszej pracy, jak to już podkreślono we wstępie, jest podanie algorytmów rozwiązywania zadań dużego rozmiaru /rozpatruje się w nich setki i więcej czynności oraz dziesiątki ograniczeń/. Dla takich zadań otrzymanie rozwiązania przy pomocy metod płaszczyzn odcinających jest niemożliwe.
Metody kombinatoryczne tym się różnią od metod płaszczyzn odcinających, że w maksymalnym stopniu wykorzystuje się w nich zarówno dyskretność problemu jak i jego kombinatoryczny charakter. Można powiedzieć, że sens tych metod polega na tym, że z góry odrzuca się te podzbiory rozwiązań, o których wiadomo, że nie zawierają rozwiązania optymalnego.
Do metod tych należy zaliczyć przede wszystkim metodę podziału i ograniczeń, a także metody wykorzystujące ogólną ideę

programowania dynamicznego.

Metoda podziału i ograniczeń po raz pierwszy została opracowana w roku 1960 przez Landa i Doiga i zastosowana do rozwiązywania zadań programowania liniowego w liczbach całkowitych. W ciągu następnych kilku latach daje się zauważyć intensywny rozwój badań nad zastosowaniem tej metody do konkretnych zagadnień z teorii przedsięwzięć czasowych.
W rezultacie powstaje szereg wariantów tej metody a właściwie pewnego schematu postępowania.
Ogólna idea tej metody jest stosunkowo prosta. Zbiór wszystkich możliwych rozwiązań dzieli się w pewien określony sposób na podzbiory rozłączne. Dla każdego otrzymanego podzbioru wyznacza się pewną wielkość, spełniającą rolę dolnego ograniczenia wartości ich funkcji kryterium / w przypadku minimalizacji funkcji kryterium/[1]. Spośród tych podzbiorów wybiera się ten, dla którego wartość dolnego ograniczenia jest najmniejsza. Wybrany podzbiór dzieli się z kolei na mniejsze podzbiory i ponownie określa się wartość dolnego ograniczenia.
Proces podziału podzbiorów rozwiązań trwa do momentu, gdy
- w wybranym podzbiorze wartość dolnego ograniczenia dla wszystkich elementów jest jednakowa lub
- wybrano podzbiór jednoelementowy.

Szczegółowy opis tej metody można znaleźć w pracach [15] , [49] , [104] , [114] , [115] .
Zastosowanie metody podziału i ograniczeń do rozwiązywania

---

[1] W przypadku maksymalizacji funkcji kryterium oblicza się górne ograniczenie.

konkretnych zadań wymaga określenia zasady podziału zbioru rozwiązań na podzbiory, zdefiniowania funkcji-kryterium oraz ustalenia sposobu znajdowania ekstremów funkcji na podzbiorach. W praktyce nie jest to sprawa prosta.

Należy zauważyć, że praktyczna realizacja na maszynie cyfrowej algorytmu opartego na metodzie podziału i ograniczeń, nawet dla stosunkowo niewielkiego zadania, wymaga znacznej pamięci, ponieważ należy przechowywać informacje o podzbiorach wyeliminowanych. Chodzi o to, że jeżeli w trakcie kolejnego podziału podzbioru na mniejsze podzbiory okaże się, że wartość ich dolnego ograniczenia jest większa niż w podzbiorach poprzednio wyeliminowanych wtedy powraca się do nich i wybiera się ten podzbiór, dla którego wartość dolnego ograniczenia wyliczona w danym momencie jest najmniejsza.

Na podstawie wielu pozycji literatury [31] , [46] , [116], w których opisano eksperymenty maszynowe z metodą podziału i ograniczeń stosowaną do zadań z teorii przedsięwzięć czasowych, a także na podstawie badań prowadzonych w tym zakresie w Instytucie Metod Rachunku Ekonomicznego Akademii Ekonomicznej we Wrocławiu przez W.Ostasiewicza na maszynie cyfrowej Odra 1204 /w trakcie przygotowywania rozprawy doktorskiej/, można sformułować tezę, że metoda podziału i ograniczeń nie pozwala rozwiązywać zadań o dużych rozmiarach.

Rozwiązywanie zadań rozpatrywanych w teorii przedsięwzięć czasowych przy pomocy programowania dynamicznego polega na zamianie jednokrotnego rozwiązania na skończony ciąg rozwiązań etapowych przy czym optymalne rozwiązanie na każdym etapie daje możliwość uzyskania optymalnych rozwiązań w następnych etapach. Inaczej mówiąc zadanie o n zmiennych zostaje przekształcone w ciąg zadań o jednej zmiennej, w którym wynik każdego kolejnego zadania jest funkcją wyniku poprzedniego zadania oraz wartości zmiennej tego zadania.

Możliwość wyznaczenia takiego wieloetapowego procesu optymalizacji opiera się na tzw. zasadzie optymalności Bellmana, której ideę można przedstawić w sposób następujący.

Niech rozwiązaniem zadania będzie ustalenie optymalnej kolejności realizacji poszczególnych czynności należących do zbioru D_1, tj. chodzi o znalezienie takiego ciągu czynności $(d_{11}, d_{12}, \ldots, d_{1m_1})$, dla którego funkcja kryterium osiąga wartość optymalną. Przyjmijmy, że wartość funkcji kryterium można także obliczyć dla każdego podciągu postaci $(d_{11}, d_{12}, \ldots, d_{1k})$ gdzie $k=1,2,\ldots,m_1-1$ oraz wszystkich jego permutacji. Podciąg taki będzie optymalnym jeżeli wartość funkcji-kryterium, wśród wszystkich możliwych podciągów składających się z takich samych elementów, jest ekstremalna.

Jeżeli w optymalnym ciągu $(d_{11}, d_{12}, \ldots, d_{1m_1})$ każdy podciąg postaci $(d_{11}, d_{12}, \ldots, d_{1k})$ gdzie $k=1,2,\ldots,m_1-1$, jest optymalny, wtedy mówimy, że dla danego zadania spełniona jest zasada optymalności Bellmana.

Zastosowanie programowania dynamicznego do rozwiązywania zadań z teorii przedsięwzięć czasowych napotyka jednak na znaczne trudności. Po pierwsze nie jest sprawą prostą sprowadzenie danego zadania do modelu wieloetapowego procesu optymalizacji. W każdym konkretnym przypadku trzeba z reguły

stosować szereg indywidualnych podejść, wymagających od badacza zarówno dużej wiedzy teoretycznej jak i doświadczenia.
Po drugie realizacja komputerowa zadań o dużych rozmiarach wymaga wykonania przez maszynę bardzo wielkiej ilości operacji elementarnych /np. dodawania, odejmowania/. Liczba takich operacji P wyraża się następującym wzorem [134, s.35]

$$P = 10^{21} \cdot N$$

gdzie:

1 - ilość czynności

N - przewidziany w harmonogramie czas wykonania wszystkich czynności /w godz.,dniach itp/.

Ułożenie harmonogramu przy zastosowaniu programowania dynamicznego dla 1000 czynności wymagałoby tak długiego nieprzerwanego czasu pracy komputera, który może być tylko porównywany z okresem istnienia naszej galaktyki. Oczywiście nawet zastosowanie super szybkich komputerów niewiele zmieni te proporcje.

Tak więc można powiedzieć, że w chwili obecnej metody kombinatoryczne nie nadają się do rozwiązywania zadań o dużych rozmiarach.

Trzecią grupę metod stanowią metody przybliżone. W przypadku zadań o dużych rozmiarach są one właściwie jedynymi efektywnymi metodami rozwiązania. Mimo to nie należy negować znaczenia pierwszej i drugiej grupy metod. Przy ich pomocy matematycy-numerycy podejmują próby rozwiązywania zadań uproszczonych, badając w ten sposób efektywne metody redukcji wariantów nieperspektywicznych. Jest to więc swego rodzaju poligon doświadczalny.

Znalezienie prostych algorytmów, przy pomocy których można otrzymać optymalne rozwiązanie zadań uproszczonych stanowiłoby niewątpliwie duży krok naprzód w poszukiwaniu podobnych algorytmów dla zadań występujących w realnych warunkach.

Analogicznie jak przy podziale dyscyplin naukowych, metody należące do dwóch pierwszych grup można traktować jako metody podstawowe, natomiast metody należące do trzeciej grupy jako stosowane.

Trzeba podkreślić, że w ostatnim okresie metody stosowane rozwijają się bardzo intensywnie. Stymulatorem rozwoju tych metod są między innymi [104]

- złożoność istniejących metod ścisłych oraz trudności posługiwania się nimi przy rozwiązywaniu zadań dużych rozmiarów,
- niedostatecznie wiarygodne dane wejściowe co powoduje, że w wielu zagadnieniach praktycznych rozwiązania ścisłe tracą swą wartość,
- specyficzna struktura wielu zagadnień.

Wśród metod stosowanych czyli inaczej przybliżonych można wyróżnić zasadniczo dwie grupy. Pierwsza grupa wykorzystuje tzw. poszukiwanie przypadkowe, druga natomiast oparta jest na programowaniu heurystycznym.

Ogólnie ideę metod poszukiwania przypadkowego można przedstawić w sposób następujący.

Losowo generowane jest pewne rozwiązanie dopuszczalne. Następnie określane jest otoczenie tego rozwiązania składające się z punktów bliskich temu rozwiązaniu dopuszczalnemu. Wśród punktów tego otoczenia znajdowane jest rozwiązanie z ekstremalną wartością funkcji kryterium. Z kolei generowane jest losowo nowe rozwiązanie dopuszczalne i znowu znajdowane jest lokalne optimum w otoczeniu tego rozwiązania. Taki proces powtarzany jest wielokrotnie. Następnie spośród otrzymanych lokalnych optimów wybierane jest rozwiązanie najlepsze w sensie wartości funkcji kryterium. Otrzymane rozwiązanie będzie rozwiązaniem przybliżonym.
Mimo niewątpliwej prostoty, stosowanie takiego sposobu postępowania w celu znalezienia rozwiązania przybliżonego dla zadań o dużych rozmiarach napotyka na trudności. Zadania te zawierają dużą liczbę ograniczeń i warunków co powoduje, że sformułowanie algorytmu losowego generowania rozwiązania dopuszczalnego, a następnie realizowanie takiego algorytmu na maszynie cyfrowej nie jest sprawą prostą. Zadania o dużych rozmiarach charakteryzuje też duża liczba czynności, dla których należy ułożyć harmonogram. Trzeba więc brać pod uwagę koszty obliczeń na maszynie, która w tym przypadku będą znaczne.

Na obecnym etapie rozwoju teorii przedsięwzięć czasowych, dla zadań o dużych rozmiarach najlepsze efekty daje zastosowanie metod bazujących na programowaniu heurystycznym tzw. m e t o d   h e u r y s t y c z n y c h . Istnieje tutaj pełna zgodność poglądów wśród znanych autorytetów naukowych zajmujących się tą problematyką [41] , [95] , [132], [134] , [164] , [177] I tak np. znany matematyk radziecki, członek-korespondent AN ZSRR N.Moisiejew twierdzi, że metody heurystyczne to obecnie jedyne metody przy pomocy których można otrzymać rozwiązanie zadań z teorii przedsięwzięć czasowych spotykanych w praktyce [134 s.253].

D.Judin, znawca badań operacyjnych, zgadza się z twierdzeniem N.Moisiejewa dodając przy tym, że metody heurystyczn-ne w odróżnieniu od innych metod pozwalają rozwiązywać nie tylko zadania dużych rozmiarów ale i uwzględniać w rozwiązaniach różnorodne ograniczenia produkcyjne i technologiczne [125 s.580] .

Konieczność i celowość stosowania metod heurystycznych do rozwiązywania zadań o dużych rozmiarach rozpatrywanych w teorii przedsięwzięć czasowych można przejrzyście i przekonywująco zaprezentować, analizując sposób rozwiązania z a d a n i a o o p t y m a l n y m n a p e ł n i e n i u z b i o r n i k a, sformułowanego przez Z.Hellwiga. Zadanie to Z.Hellwig sformułował w sposób następujący: Niech będzie dany zbiornik o pojemności c. Zbiornik ten trzeba napełnić wodą za pomocą naczyń z pewnego zbioru o liczebności n. Naczynia ponumerujemy liczbami naturalnymi $1,\dots,k,\dots,n$. Dla k-tego naczynia zadana jest pojemność, którą będziemy oznaczali symbolem p_k, przy czym

$$\sum_{k=1}^{n} p_k \neq c \,.$$

Spośród wektorów $K = (k_1, k_2, \dots, k_s)$ dla $s = 1,2,\dots,z$ należy określić taki wektor, aby wyrażenie $c = \sum_{l=1}^{s} p_{k_l}$ osiągało minimum, przy ograniczeniu $(c - \sum_{l=1}^{s} p_{k_l}) \geqslant 0$.

Określimy na początku przestrzeń rozwiązań tego zadania. Będzie się ona składała z $C^1_{n+1-1} + C^2_{n+2-1} + \ldots + C^z_{n+z-1}$ elementów. Nie każdy z tych elementów będzie rozwiązaniem dopuszczalnym. Aby element z przestrzeni rozwiązań był rozwiązaniem dopuszczalnym, musi on spełniać warunek

$$\sum_{l=1}^{s} p_{k_l} \leqslant c.$$

Oczywiście w naszym zadaniu chodzi o znalezienie takiego rozwiązania wśród elementów zbioru rozwiązań dopuszczalnych, dla którego funkcja-kryterium osiąga minimum czyli r o z - w i ą z a n i a o p t y m a l n e g o .

Do znalezienia rozwiązania optymalnego może być zaproponowana metoda pełnego przeliczania wariantów. W rozważanym zadaniu sprowadziłoby się to do wygenerowania wszystkich elementów przestrzeni rozwiązań. Metoda ta daje zawsze gwarancję uzyskania rozwiązania optymalnego, jednak już przy małych wartościach n generowanie wszystkich wariantów jest nie do pokonania nawet dla najszybszych komputerów.

Rozważane zadanie można sformułować w kategoriach programowania zero-jedynkowego:

Zminimalizować funkcję $f = c - \sum_{k=1}^{n} \sum_{j=1}^{s} x_{kj} p_k$, gdzie

$$x_{kj} = \begin{cases} 1 & \text{gdy do napełnienia zbiornika użyte będzie } k\text{-te naczynie } j\text{-ty raz z kolei} \\ 0 & \text{w przeciwnym przypadku,} \end{cases}$$

$s = 1, 2, \ldots, z$

przy następujących ograniczeniach

$$\sum_{k=1}^{n} \sum_{j=1}^{s} x_{kj} p_k \leqslant c , \qquad \sum_{k=1}^{n} \sum_{j=1}^{s} x_{kj} = s .$$

Do znajdowania rozwiązań dopuszczalnych a następnie rozwiązania optymalnego można wygenerować wszystkie elementy przestrzeni rozwiązań.

O wiele bardziej efektywne są specjalne algorytmy dla rozwiązywania zadań programowania zero-jedynkowego /np. B-algorytm J.Finkelsztejna por. [104]/.

W porównaniu z pełnym przeliczeniem zastosowania tego typu algorytmów zmniejsza kilkunastokrotnie czas obliczeń na maszynie [104]. Mimo to przy dużych wartościach s czas obliczeń na maszynie jest znaczny.

W przypadku dużych n, s oraz c do rozwiązania zadania o optymalnym napełnieniu zbiornika trzeba stosować metody przybliżone. Przy pomocy metod przybliżonych generowany jest pewien podzbiór ze zbioru rozwiązań dopuszczalnych. Wśród elementów wygenerowanego podzbioru wybierane jest rozwiązanie najlepsze ze względu na zadane kryterium. O rozwiązaniu takim trudno najczęściej powiedzieć czy jest rozwiązaniem optymalnyc

Jedną z częściej stosowanych metod przybliżonych jest metoda Monte-Carlo. Metodę tę w zastosowaniu do rozważanego zadania można przedstawić następująco:

1. Znajdujemy liczbę losowań L, wektor $S = (s_1, s_2, \ldots, s_r)$ gdzie s_i oznacza liczbę składowych wektora K - kładziemy $i=1$.
2. Przy pomocy generatora liczb losowych generujemy L liczb naturalnych z przedziału $[1, C_{n+s_i-1}^{s_i}]$. Wygenerowane liczby naturalne stanowią numery elementów przestrzeni rozwiązań tj. numery s_i-elementowych wektorów K. Elementy poszczególnych wektorów K rozszyfrowujemy przy pomocy algorytmu denumerowania kombinacji z powtó-

rzeniami zaproponowanego przez W.Ostasiewicza [140].

3. Sprawdzamy czy poszczególne wektory K stanowią rozwiązania dopuszczalne. Wśród rozwiązań dopuszczalnych znajdujemy rozwiązania najlepsze ze względu na zadane kryterium. Oznaczymy je symbolem $\bar{K}(i)$ przy czym gdy $i=1$ rozwiązanie to będziemy oznaczać symbolem $K^{*}(1)$. Ogólnie symbolem $K^{*}(i)$ będziemy oznaczać najlepsze z dotychczas wygenerowanych rozwiązań.

4. Dla $i=2,3,\ldots,$ badamy czy rozwiązanie $\bar{K}(i)$ jest lepsze od rozwiązania $K^{*}(i-1)$. Gdy tak jest wtedy rozwiązaniem $K^{*}(i)$ będzie rozwiązanie $\bar{K}(i)$, w przeciwnym przypadku rozwiązanie $K^{*}(i-1)$ będzie także rozwiązaniem $K^{*}(i)$.

5. Wartość i zwiększamy o jeden. Gdy $i > r$ wtedy obliczenia przerywamy, w przeciwnym przypadku przechodzimy do punktu 2.

Należy zaznaczyć, że zamiast liczby L' może być zadany wektor, w którym poszczególne składowe oznaczają liczbę losowań.

Do metod przybliżonych należą metody heurystyczne. W przypadku rozważanego zadania - metodą heurystyczną można nazwać następujące postępowanie:

1. Porządkujemy malejąco ciąg $p_1, p_2, \ldots, p_i, \ldots, p_n$, kładziemy $j=1$, $i=1$.

2. Wybieramy m_i razy i-ty element uporządkowanego ciągu, przy czym wielkość m_i powinna być taka, aby:

$$\left(\sum_{j=1}^{m_1} p_1^{(j)} + \ldots + \sum_{j=1}^{m_i} p_i^{(j)} \right) - c < p_i$$

Odpowiadający i-temu elementowi uporządkowanego ciągu numer naczynia stanowi m_i kolejnych elementów wektora K będącego rozwiązaniem zadania.

3. Jeżeli $\sum_{j=1}^{m_1} p_1^{(j)} + \ldots + \sum_{j=1}^{m_i} p_i^{(j)} = c$ zadanie zostało rozwiązane, gdy natomiast $\sum_{j=1}^{m_1} p_1^{(j)} + \ldots + \sum_{j=1}^{m_i} p_i^{(j)} < c$ zwiększamy wskaźnik i o jeden i przechodzimy do punktu 2. Obliczenia przerywamy gdy $i > n$.

Oba opisane algorytmy zostały zaprogramowane w języku ALGOL. Na maszynie Odra 1204 przeliczono szereg przykładów. Wyniki obliczeń dla jednego spośród nich podamy poniżej.

Dane liczbowe

Pojemność zbiornika - 670

Liczba naczyń - 8

Pojemność poszczególnych naczyń - 38, 28, 19, 15, 12, 9, 6, 3.

Metoda Monte-Carlo

Przyjęto następujący wektor $S = (17, 18, 19, 20, 21, 22)$ oraz dokonano losowań dla L równego 20, 50, 70, 90, 110, 130, 150. Wyniki obliczeń zawarte są w tablicy 1.

Tablica 1

Liczba L	Rozwiązanie		Wartość funkcji kryterium
	Liczba naczyń	Numery naczyń	
20	22	1,1,1,1,1,1,1,1,1,1,2,2,2,2, 2,2,2,4,5,8,8,8,	43
50	21	1,1,1,1,1,1,1,1,1,2,2,2,2,2, 2,2,4,4,5,5,6	69
70	21	1,1,1,1,1,1,1,1,1,1,1,1,1,1, 2,3,4,5,5,6,6,	34
90	21	1,1,1,1,1,1,1,1,1,1,1,1,1,1, 1,1,4,4,8,8,8,	8
110	21	1,1,1,1,1,1,1,1,1,1,1,1,1,1, 1,2,4,5,5,6,7,	18
130	20	1,1,1,1,1,1,1,1,1,1,1,1,1,1, 2,2,3,3,3,3,	6
150	20	1,1,1,1,1,1,1,1,1,1,1,1,1,1, 2,2,3,3,3,3,	6

Jak widać najlepsze rozwiązanie uzyskano dla L = 130 oraz L = 150.

Metoda heurystyczna

Numery naczyń, którymi należy kolejno wlewać wodę do zbiornika, są następujące: 1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1, 3,8. Wartość funkcji kryterium wynosi 0.

Na podstawie przeprowadzonych eksperymentów komputerowych można sformułować następujące wnioski:

- brak podstaw do twierdzenia, że jedna z rozpatrywanych metod daje zdecydowanie lepsze wyniki,

- przy pomocy metody Monte-Carlo na komputerze Odra 1204 mogą być liczone niewielkie przykłady, podobne do zaprezentowanego. W przykładach, w których rozpatruje się kilkadziesiąt i więcej naczyń o niewielkiej pojemności w stosunku do pojemności zbiornika /tzw. zadanie o średnich rozmiarach /, rozpatrywanie wektorów K o niewielkiej liczbie składowych jest oczywiście niecelowe. Z kolei przy większej liczbie składowych, wyliczenie i zapisanie wartości $C_{n+s_i-1}^{s_i}$ w komórce pamięci komputera Odra 1204 nie jest możliwe /mamy wtedy do czynienia z liczbami składającymi się z 16 i więcej cyfr/. Ograniczenie to można pokonać pisząc program w FORTRANIE zrealizowany na komputerze Odra 1305. Jednak i w tym przypadku dalsze zwiększanie rozmiarów zadania spowoduje wystąpienie podobnych trudności jak przy obliczeniach na komputerze Odra 1204. Trudności tego typu nie występują przy stosowaniu metody heurystycznej. Za pomocą metody heurystycznej można rozwiązywać zadania dużych rozmiarów.

Tak więc w odniesieniu do zadania o optymalnym napełnianiu zbiornika można sformułować następującą tezę: rozwiązanie zadania średnich rozmiarów można otrzymać zarówno przy pomocy metody Monte-Carlo jak i metody heurystycznej, natomiast zadanie o dużych rozmiarach można rozwiązać jedynie przy pomocy metody heurystycznej.

1.4. W a ż n i e j s z e   z a g a d n i e n i a   p r a k t y c z n e

Znany matematyk radziecki W.Szkurba, twórca rosyjskiego terminu teoria raspisanji i jednocześnie ten, który przyczynił się w znacznym stopniu do rozwoju omawianej teorii stwierdza, że z zagadnieniami, którymi zajmuje się teoria przedsięwzięć czasowych można spotkać się praktycznie we wszystkich sferach działalności człowieka, ponieważ w istocie cała działalność człowieka tak lub inaczej związana jest z planowaniem w czasie [176] . Stwierdzenie to w sposób jednoznaczny podkreśla ważność tych zadań i jednocześnie wielką ich ilość.

W zadaniach teorii przedsięwzięć czasowych spotykamy się z koniecznością uporządkowania w czasie określonych prac zgrupowanych w pewne kompleksy i znajdujących się w pewnych z góry zadanych związkach. Podobnej problematyki tj. przebiegu dyskretnych procesów w czasie dotyczą zadania z teorii obsługi masowej.

W niniejszej pracy przyjmiemy za Szkurbą i Burdiukiem, że teoria przedsięwzięć czasowych bada zdeterminowane systemy obsługi, natomiast teoria obsługi masowej stochastyczne systemy obsługi [35] .

Wśród zadań z teorii przedsięwzięć czasowych czołowe miejsce zajmują zadania występujące w planowaniu produkcji głównie w tzw. o p e r a t y w n y m   p l a n o w a n i u   p r o d u k c j i   zwanym też kalendarzowym lub wykonawczym.

Oznacza ono planowanie na krótkie odcinki czasu i zaczyna się na szczeblu przedsiębiorstwa od ustalenia planów kwartalnych. Problematyka optymalizacji operatywnych planów produkcji to jedno z najbardziej złożonych zagadnień matematyczno-ekonomicznych. Od szeregu lat znajduje się ona w centrum zainteresowań matematyków i ekonomistów zarówno u nas w kraju jak i za granicą. Zainteresowanie to nie jest przypadkowe. Wynika ono z ogromnego zapotrzebowania przedsiębiorstw na algorytmy inne od dotychczasowych w związku z coraz szerszym stosowaniem komputerów.

Problemy występujące w operatywnym planowaniu produkcji mimo na ogół prostego sformułowania nie są łatwe do rozwiązania. W przedsiębiorstwach stosowane są przede wszystkim tzw. o b j ę t o ś c i o w e   m e t o d y   p l a n o w a n i a [144] zwane też statycznymi, bądź gdy chodzi o najniższy poziom planowania /tzw. planowanie wewnątrzwydziałowe lub planowanie operacji/ pozostawia się je w gestii mistrzów, którzy rozwiązują je na bieżąco w trakcie realizacji produkcji Powoduje to spiętrzenie prac w pewnych okresach, przestoje lub wykonywanie produkcji na tzw. magazyn.

Stosowanie objętościowych metod związane jest z nieuchronnym popełnieniem znacznych błędów dlatego, że metody te nie uwzględniają technologii wytwarzania. Otrzymane przy ich pomocy proporcje między potrzebnymi i dysponowanymi środkami nie odpowiada z reguły rzeczywistości, ponieważ w związku z realizacją produkcji w porządku technologicznym potrzeby na określone środki w różnych przedziałach czasowych będą niejednakowe. Można powiedzieć, że planowanie przy pomocy metod objętościowych pociąga za sobą nierównomierność obciążenia maszyn i urządzeń, a także stanowi jedną z przyczyn wszelkiego rodzaju napięć w procesie realizacji planu.

Trudności tych można uniknąć stosując bardziej racjonalne metody planowania. Metody te związane są z takim sformułowaniem zadań, w którym jest modelowane rozłożenie potrzebnych środków. Jak wiadomo takimi zadaniami zajmuje się teoria przedsięwzięć czasowych. Takich zadań sformułowanych w kategoriach teorii przedsięwzięć czasowych w procesie określania operatywnych planów produkcji może być wiele.
W literaturze a także w rozwiązaniach praktycznych rozpatruje się pewne grupy. I.Szubkina wyróżnia pięć następujących grup, które nazywa krótko zadaniami [178] .

1. Zadanie rozłożenia rocznego programu produkcji wyrobów według kwartałów.
2. Zadanie rozłożenia programu montażu wyrobów według miesięcy.
3. Zadanie rozłożenia programu produkcji detali na wydziałach obróbczych i przygotowawczych według miesięcy.
4. Zadanie określania harmonogramu produkcji dla wydziału montażowego.
5. Zadanie określenia harmonogramu produkcji wydziału obróbczego i przygotowawczego.

Podobny podział proponują też inni autorzy zajmujący się tą problematyką [14] , [47] , [164] .
W czterech pierwszych zadaniach w zbiorach czynności jest zadany porządek B, natomiast w ostatnim porządek A. Biorąc pod uwagę zaproponowaną wcześniej klasyfikację zadań, zadanie

pierwsze, drugie, trzecie i czwarte należeć będzie do zadań z szeregowo-równoległą kolejnością czynności /zadanie B/, a zadanie piąte do zadań z szeregową kolejnością czynności /zadanie A/.

Krąg zadań, którymi zajmuje się teoria przedsięwzięć czasowych nie ogranicza się oczywiście do operatywnego planowania produkcji.

Z zadaniami takimi spotykamy się np. w szkole wyższej. Wymienić tu można przykładowo takie problemy jak układanie harmonogramu zajęć, układanie harmonogranu egzaminów, urlopów itp.

Centralne miejsce w tej grupie problemów zajmuje zadanie układania harmonogramu zajęć. W zadaniu tym występuje porządek C. Będzie więc ono należało do zadań z częściową kolejnością czynności /zadanie C/.

W chwili obecnej układanie harmonogramu zajęć w naszych uczelniach odbywa się ręcznie. Taki sposób układania harmonogramuwjest bardzo pracochłonny, a otrzymane rozwiązanie jest z reguły krytykowane zarówno przez studentów jak i pracowników.

W wielu uczelniach /Akademia Górniczo-Hutnicza, Politechnika Gdańska, Politechnika Wrocławska, WAT, Uniwersytet Warszawski, oraz Akademia Ekonomiczna we Wrocławiu/ podjęto prace nad możliwością komputerowego układania harmonogramu zajęć.

Ważnym zagadnieniem jest też zagadnienie układania rozkładu jazdy pociągów czy autobusów. Zostało ono szczegółowo opisane /wraz z propozycją przybliżonej metody rozwiązania/ w pracy [45] .

Zadania odnoszące się do teorii przedsięwzięć czasowych można też spotkać w wojsku / [52] str. 167 i nast./, a także w procesie świadczenia usług [106] .

W dalszej części pracy zostanie omówione zastosowanie metody priorytetów do rozwiązywania zadań A, zadań B oraz zadań C.

Rozwiązywanie zadań A oraz B przy pomocy metody priorytetów zostanie zaprezentowane na przykładach z operatywnego planowania produkcji, przy czym zadanie B zostanie omówione na przykładzie zadania określania harmonogramu produkcji dla wydziału montażowego /zadanie Nr 4/, natomiast zadanie A na przykładzie zadania określania harmonogramu produkcji wydziału obróbczego i przygotowawczego /zadanie Nr 5/.

Z kolei rozwiązywanie zadań C przy pomocy metody priorytetów zostanie omówione na przykładzie układania harmonogramu zajęć w szkole wyższej.

Należy zaznaczyć, że wybrane zadania są najbardziej typowe w każdej z trzech grup zadań.

2. PROGRAMOWANIE HEURYSTYCZNE JAKO PODSTAWA METODY PRIORYTETÓW

2.1. W s t ę p

Należy zauważyć, że wiele zadań rozważanych w teorii przedsięwzięć czasowych jest rozwiązywanych "ręcznie" przez człowieka. O otrzymanym rozwiązaniu trudno powiedzieć czy jest to rozwiązanie optymalne, jednak jest to rozwiązanie całkowicie do przyjęcia w praktyce.

Powstaje pytanie dlaczego człowiek, który liczy tysiące razy wolniej niż komputer jest w stanie rozwiązać takie zadanie. Okazuje się, że człowiek stosuje wysoce selektywne poszukiwanie w zbiorze rozwiązań dopuszczalnych. W sposób mało precyzyjny takie cechy postępowania człowieka przy rozwiązywaniu zadań określa się jako "pomysłowość" lub "intuicję". W rzeczywistości jest to rezultat rozumnego stosowania pewnych zasad opartych na doświadczeniu, strategii, uproszczeń zwanych h e u r y s t y k a m i , które drastycznie ograniczają zakres poszukiwania rozwiązania i odpowiednio kierują tymi poszukiwaniami.

2.2. P o j ę c i a p o d s t a w o w e

Problemami tworzenia heurystyk zajmuje się tzw. p r o g r a m o w a n i e h e u r y s t y c z n e, którego podstawowym zadaniem jest uwolnienie się od konieczności stosowania metody wyczerpującego przeglądu wariantów. Jak stwierdza

B. Puszkin programowanie heurystyczne bazuje po pierwsze na odtworzeniu niektórych czynności umysłu ludzkiego oraz po drugie na analizie swoistych cech rozwiązywanego problemu [156].

Za początek prac badawczych w dziedzinie programowania heurystycznego należy uznać opracowanie w roku 1956 przez A. Newella, I. Shawa, H. Simona programu dla komputera pod nazwą "Teoretyk logiki". Program ten dowodził twierdzeń logiki matematycznej. Przy jego opracowywaniu zastosowano heurystyki przyjęte na podstawie analizy procesu rozwiązywania podobnych zadań przez człowieka. Było to pierwsze osiągnięcie w dziedzinie procesów intelektualnych wysokiej klasy.

W kilka lat później A. Newell, I. Shaw oraz H. Simon podjęli próbę zbudowania programu, który mógłby być zastosowany do rozwiązywania różnorodnych problemów. Programowi temu dali oni nazwę General Problem Solver - GPS /Ogólny Rozwiązywacz Problemów/. Badacze ci wyszli z założenia, że człowiek w procesie rozwiązywania różnorodnych zadań posługuje się pewnymi ogólnymi strategiami. Opracowany program nie był jednak na tyle ogólny jak założono początkowo. Stosowano go głównie do rozwiązywania pewnych zadań z dziedziny logiki, trygonometrii i algebry.

Popularnym poligonem badań dotyczących programowania heurystycznego było i jest w dalszym ciągu układanie programów grających w gry. Problemy występujące w grach są stosunkowo łatwo formułowane w sposób ścisły i prosty. Z kolei znalezienie rozwiązania jest na tyle skomplikowane, że decydującą rolę w procesie rozwiązywania odgrywa intelekt człowieka 1/. Konstruowanie a następnie realizacja na maszynie programów dotyczących różnego rodzaju gier w dużym stopniu przyczyniła się do wyjaśnienia niektórych aspektów twórczego myślenia przy rozwiązywaniu złożonych problemów. Badania w tym zakresie są prowadzone szczególnie w USA i ZSRR.

Do chwili obecnej opracowano szereg programów heurystycznych rozwiązujących zadania z różnych dziedzin życia. Można tu zwrócić uwagę na takie programy jak 2/: programy dowodzenia twierdzeń geometrycznych, programy wyznaczania całek nieoznaczonych, programy komponujące muzykę, programy określające strukturę chemiczną, programy układające programy dla komputera, programy rozpoznawania obrazów a także szereg programów z dziedziny sterowania ekonomiką.

Większość z tych programów znalazła zastosowanie w praktyce a ponadto przyczyniła się do ukształtowania teorii programowania heurystycznego i wypracowania metod rozwiązywania pewnych klas problemów. Omawiając problemy dotyczące teorii programowania heurystycznego trzeba zwrócić uwagę na metodologię programowania heurystycznego. Wg J. Aleksandrowa współczesna metodologia programowania heurystycznego składa się z następujących etapów [125]

a/ Badacz obserwując rozwiązywane przez badany obiekt za-

1/ Warto zaznaczyć, że np. przy grze w szachy liczba możliwych układów figur szachowych na szachownicy wynosi 10^{120}. Badania dotyczące gry w szachy przez maszynę nazywane są kamieniem probierczym różnych sposobów modelowania intelektu ludzkiego [156].

2/ Zasady działania niektórych z wymienionych tutaj programów a także szeregu innych zostały szczegółowo omówione w monografii [165].

dania testowe buduje algorytm rozwiązania.

b/ Algorytm jest realizowany na maszynie cyfrowej i proces konstruowania rozwiązania a także wyniki końcowe porównywane są z postępowaniem czyli wynikami pośrednimi oraz wynikami końcowymi uzyskanymi przez obiekt badany.

c/ Jeżeli są w tym względzie odchylenia, szuka się ich przyczyny i dokonuje się korekty algorytmu. W wypadku braku odchyleń algorytm sprawdza się na nowych zadaniach tej klasy.

d/ Proces kończy się, gdy dla większości zadań danej klasy nie występują odchylenia.

Należy zauważyć, że w konkretnych przypadkach mogą zachodzić pewne zmiany w przedstawionej wyżej metodologii. Czasem np. nie można sformułować zadania testowego. Wtedy badacz obserwuje i analizuje działania obiektu badanego.

Podstawowym pojęciem w teorii programowania heurystycznego jest pojęcie heurystyki.

Jak słusznie stwierdza P. Jędrzejowicz, w literaturze można spotkać szereg prób zdefiniowania tego pojęcia [92]. Heurystyka określana bywa jako proces rozwiązywania problemu lub też jako proces psychiczny, dzięki któremu znajduje się wyjście z sytuacji problemowych.

W pracy [123] pod pojęciem heurystyki rozumie się umiejętność wykrywania nowych faktów i związków między nimi, w szczególności formułowania hipotez umożliwiających odkrywanie nowych prawidłowości naukowych i wytwarzanie pomysłów nowych rozwiązań konstrukcyjnych.

W pracy niniejszej przyjmujemy pojęcie heurystyki podane na str. 46 . W ten sposób interpretowane jest pojęcie heurystyki w zdecydowanej większości pozycji literatury rozpatrujących metody wykorzystujące intelekt ludzki do rozwiązywania zadań przy pomocy komputera.
Trzeba jednak przyznać, że do chwili obecnej w teorii programowania heurystycznego brak jest ogólnego schematu tworzenia i stosowania heurystyk. Na podstawie dotychczasowych badań stwierdza się jedynie, że heurystyki można podzielić na dwa rodzaje na tzw. heurystyki uniwersalne i heurystyki specjalistyczne.
Heurystyki uniwersalne są stosowane w rozwiązywaniu różnorodnych zagadnień. Oto przykłady takich heurystyk.

1/ Rozumowanie przez analogię.
Jest to heurystyka bardzo często używana przez człowieka w procesie poszukiwania rozwiązania. Chodzi tu o to, że w celu znalezienia rozwiązania pewnego zadania można zastosować podejścia używane przy rozwiązywaniu zadań podobnej klasy / bądź węższej /[1].

2/ Podział problemu na mniejsze podproblemy/[2]. Zadania tzw. dużych rozmiarów dzieli się na kilka zadań bardziej prostych do rozwiązania. Tak np. zadanie układania harmonogramu w pewnym prze-

---

1/ I.Aleksandrow mówiąc o tej heurystyce interpretuje ją w sposób następujący. Niech będą dane dwa systemy, badany S' i znany S". Określone relacje między elementami $s' \in S'$ mają te same własności jak relacje między elementami $s'' \in S''$. Niech także między elementami s' i s" istnieje równoważność. Jeżeli teraz w S" znaleziono rozwiązanie s" to oznacza to, że znaleziono rozwiązanie s' [5] .

2/ W pracy [134] ta heurystyka jest określana jako osiąganie celów przejściowych.

dziale czasowym T zamienia się na kilka zadań układania harmonogramu na mniejsze odcinki czasu.

3/ A g r g o w a n i e. Rozpatrując zadanie łączy się w określony sposób pewne zmienne. Zmniejsza się w ten sposób rozmiar zadania oraz upraszcza się proces konstruowania rozwiązania. I tak np. układając harmonogram łączy się pewne czynności w wyniku czego otrzymuje się czynności zagregowane.

4/ R e g u ł y z w y c z a j o w e. Są to najczęściej reguły oparte na doświadczeniu praktycznym, pewne nawyki i zwyczaje.

5/ A n a l i z a k r o k o w a. Polega ona na przekształcaniu stanu początkowego rozpatrywanego zagadnienia w stan końcowy / docelowy / przy pomocy wybierania i stosowania pewnych reguł, operacji logicznych czy arytmetycznych, które w każdym kolejnym kroku redukują różnicę między tymi stanami.

6/ E k s p e r y m e n t o w a n i e i u c z e n i e s i ę. Opracowując metodę rozwiązania zadań o dużych rozmiarach postępuje się często w ten sposób, że przeprowadza się eksperyment w mniejszej skali.

Heurystyki specjalistyczne dotyczą wąskiej klasy zadań bądź nawet konkretnych zadań.

I tak np. w programach wyznaczania całek nieoznaczonych stosowana jest bardzo często następująca heurystyka: "jeśli w wyrażeniu podcałkowym można wyróżnić taką część składową, że jej pochodna jest dzielnikiem tego wyrażenia, to funkcję stanowiącą tę część składową przyjmujemy jako nową zmienną" [124]. W programach gry w szachy stosowane bywają takie heurystyki jak: " zawsze chroń króla, zawsze szachuj, może być mat" [92]. Przy dowodzeniu tożsamości trygonometrycznych jedną z podstawowych reguł jest redukowanie wszystkich wyrażeń do postaci sinusów i cosinusów.

Powstaje pytanie w jaki sposób należy szukać heurystyk specjalistycznych.

Badacz, który chce rozwiązać określony problem powinien przede wszystkim zapoznać się z literaturą gdzie może znaleźć propozycje pewnych heurystyk, powinien rozmawiać i obserwować jak rozwiązują takie lub podobne zadania specjaliści praktycy zapisując dokładnie ich "głośne myślenie", następnie powinien sam rozwiązać prostsze przykłady "ręcznie" i /lub/ przeprowadzić eksperymenty przy pomocy komputera.

Z pojęciem heurystyki związane jest ściśle pojęcie metody heurystycznej. Niekiedy o heurystykach uniwersalnych mówi się, że są to metody heurystyczne. Wydaje się, że takie podejście jest do przyjęcia w przypadku gdy w zadaniu zastosowano jedną heurystykę.

Ogólnie za J. Slagle można powiedzieć, że metoda heurystyczna jest to metoda pomagająca znaleźć rozwiązanie zadania przez podanie wiarygodnych, lecz nie zawsze słusznych przypuszczeń, dotyczących optymalnej decyzji na danym etapie rozwiązania zadania [165].

Dzięki stosowanym przez metody heurystyczne heurystykom upraszcza się znacznie proces konstruowania rozwiązania, ale odbywa się to kosztem rezygnacji z warunku aby rozwiązanie było optymalne.

Radziecki matematyk N. Moisiejew twierdzi, że rozwój myśli matematycznej prowadzi stopniowo do tego, że matematycy zaczynają rozumieć i akceptować miejsce metody heurystycznej w rozwiązywaniu określonych zagadnień. Takie tendencje są najbardziej zauważalne w teorii informacji i teorii rozpoznawania obrazów [134].

Z kolei T. Kasprzak podaje że "lawina informacyjna, kompleksowy charakter rozwiązań zmusza nas w pierwszym rzędzie do modelowania procesu decyzyjnego, następnie do jego aglorytmizacji i ogólnie do poznania zasad pozwalających na znajdowanie rozwiązań. Przy okazji mogą to być rozwiązania najlepsze z możliwych, ale to jest niejako dodatkowy warunek nałożony na modelowanie procesu decyzyjnego" [15. s. 15]

Dalej T. Kasprzak twierdzi, że ostatnio ulega modyfikacji taka teoretyczna kategoria matematyczna jaką jest istnienie rozwiązania. Według niego "przez rozwiązanie dopuszczalne rozumiemy dziś nie tylko rozwiązanie niesprzeczne, ale takie, które możemy uzyskać w określonym przedziale czasu i przy koszcie nie przewyższającym spodziewanego wzrostu efektu w wyniku zastosowania rozwiązania w praktyce gospodarczej [15 s. 22].

Powyższe rozważania pozwalają zrozumieć sens i potrzebę stosowania metod heurystycznych.

Wiele sporów i dyskusji wśród teoretyków programowania heurystycznego wywoływał i ciągle wywołuje problem wzajemnej relacji między metodami heurystycznymi i metodami algorytmicznymi. Przelewano / i ciągle jeszcze przelewa się / na tym polu, jak stwierdzono obrazowo w pracy [124] , wiele "intelektualnej krwi".

Przez metodę algorytmiczną rozumie się metodę, w której zastosowany zbiór reguł podany w określonej sekwencji jeśli się go dokładnie przestrzega z całą pewnością prowadzi do uzyskania pożądanego rozwiązania. Zastosowanie reguł w metodach heurystycznych nie daje natomiast takiej gwarancji [92] Pod metodami algorytmicznymi rozumie się często metody optymalizacyjne a więc metody, które zapewniają zawsze rozwiązanie optymalne [6] . Różnica między heurystyczną a nieheurystyczną metodą rozwiązania jest często nieuchwytna. Z tego względu zamiast określać zasadę, według której można podzielić wszystkie możliwe metody rozwiązywania zadań na dwie grupy, rozsądniej jest podać cechy charakterystyczne stosowanych obecnie metod heurystycznych. W pracy [124] wymienia się takie cechy jak:

- dekompozycja problemu na względnie niezależne podproblemy oraz hierarchizacja celów połączona z wyodrębnieniem celów globalnych i cząstkowych,
- wykorzystanie informacji zewnętrznych / z otoczenia /, które pomagają wybrać określone sposoby postępowania spośród wielu możliwych,
- nie istnienie gwarancji uzyskania rozwiązania zadowalającego.

Jak dotychczas osiągnięcia w dziedzinie teorii programowania heurystycznego nie są znaczne. Można powiedzieć, że programowanie heurystyczne znajduje się na początkowym odcinku krzywej logistycznej swego rozwoju tzn. przy pomocy metod programowania heurystycznego rozwiązywane są konkretne zadania i podejmowane są próby uogólnienia osiągniętych wyników. Nie ulega wątpliwości, że rozwój teorii programowania heurystycznego przyczyni się w znacznym stopniu do rozszerzenia sfery zastosowań komputerów.

W literaturze zwraca się uwagę na następujące problemy teoretyczne, które obecnie są przedmiotem szczególnego zainteresowania [165] :

- ogólność; chodzi tutaj o budowę takich programów heurystycznych, które mogłyby rozwiązywać różnorodne zadania.
- poszukiwanie; problem ten polega na tworzeniu takich algorytmów poszukiwania w przestrzeni dopuszczalnych rozwiązań, które zapewniałyby otrzymanie rozwiązania dopuszczalnego przy minimalnym zużyciu takich środków jak czas i pamięć maszyny.
- funkcje oceny; w każdym kroku konstruowania rozwiązania powinno się przeprowadzać oceny dla wyjaśnienia, która z rozpatrywanych sytuacji jest najlepsza. Problem polega na poszukiwaniu pewnych nowych własności, które stanowiłyby podstawę do określania takich ocen.
- uczenie się; chodzi tu o to, aby program heurystyczny w trakcie jego realizacji przez komputer mógł uwzględniać doświadczenie, korygując następnie dalsze postępowanie.

Wypada zwrócić uwagę na jeszcze jeden problem, który znajduje się w centrum uwagi teoretyków programowania maszyn cyf-

rowych. W związku z coraz szerszym realizowaniem programów heurystycznych na maszynie cyfrowej, prowadzone są intensywne studia nad opracowaniem języków dostosowanych do programowania zagadnień heurystycznych. Na zachodzie zanotowano już pewne sukcesy. Dzięki m.in. pracom prowadzonym w dziedzinie programowania heurystycznego powstały takie języki jak: LISP, IPL, FLPL. Języki te znacznie ułatwiają programowanie zapewniając dynamiczne przydzielanie miejsca w pamięci oraz przetwarzanie tzw. struktur listowych.
Rozległe prace nad opracowaniem bardziej uniwersalnych języków trwają w ZSRR. Kieruje tymi badaniami znany uczony A. Jerszow. Ostatnio na VII sympozjum cybernetycznym odbywającym się w Tbilisi /4-6 czerwca 1974/ A. Jerszow wygłosił referat p.t. "Programmnoje obespieczenie dla zadacz po iskusstwiennomu intielektu", w którym omówił problematykę tych badań.

2.3. Zagadnienie sztucznej inteligencji

Programowanie heurystyczne stanowi jeden ze sposobów podejścia do rozległego problemu badawczego w cybernetyce a mianowicie do tzw. sztucznej inteligencji.
Jak wiadomo terminu "inteligencja" używa się w różnym sensie. W tej pracy termin ten będzie rozumiany zgodnie z dwiema następującymi definicjami [165]:

a/ Inteligencja to umiejętność reagowania w dowolnej a szczególnie w nowej sytuacji drogą odpowiedniej korekty dotychczasowego postępowania.

b/ Inteligenoja to umiejętność rozumienia związków występujących w naszej rzeczywistości w celu wypracowania posunięć prowadzących do osiągnięcia postawionego celu.

Ogólnie można powiedzieć, że celem badań w dziedzinie sztucznej inteligencji jest budowanie takich programów komputerowych, które zachowują się w sposób inteligentny.
Prace badawcze w tej dziedzinie prowadzone są w dwóch zasadniczych kierunkach.
Pierwszy kierunek zajmuje się problemami wyjaśnienia istoty intelektu człowieka. Chodzi tu więc o takie programowanie komputerów aby rozwiązywały one zadania dokładnie tak samo jak to czynią ludzie. Ten kierunek nazywany jest w literaturze symulacją procesów myślenia [124] . Prace w tym zakresie prowadzone są głównie przez psychologów.
O drugim kierunku można powiedzieć, że zajmuje się konstruowaniem takich algorytmów rozwiązywania złożonych zadań, w których wykorzystuje się inteligencję człowieka. Problemy te są przedmiotem studiów badaczy z rozmaitych dyscyplin wiedzy.
Jak podaje J. Slagle do problematyki badawczej sztucznej inteligencji stosowane są obecnie trzy podejścia [165]

- budowa sztucznych sieci,
- sztuczne odtworzenie ewolucji,
- programowanie heurystyczne.

Dla pierwszego przypadku na maszynie cyfrowej modelowana jest sieć,której węzły stanowią odpowiedniki komórek nerwowych /niekiedy buduje się model fizyczny takiej sieci/. Zwolennicy tego podejścia w dziedzinie sztucznej inteligencji próbują rozwiązywać problemy przy pomocy tych sieci , ponieważ jak twierdzą intelekt człowieka jest formowany na bazie podobnych sieci. Jak dotychczas sztuczne sieci "nauczyły się" tylko rozpoznawać bardzo proste obrazy. Nie wydaje się też możliwe stworzenie takiej sieci, którą można by chociaż w przybliżeniu porównać z mózgiem człowieka składającym się z około 10^{10} komórek, tym bardziej że nauka jest jeszcze bardzo daleka od rozumienia mechanizmów pracy i wzajemnego oddziaływania komórek nerwowych.

Przy podejściu polegającym na sztucznym odtworzeniu ewolucji próbuje się przekształcać model systemu przedstawiony w pamięci maszyny, tworząc kolejno mutacje takiego systemu. W ten sposób w kilku przypadkach udało się przekształcić pewne modele tak, że rozwiązywały bardzo proste zadania. Zwolennicy tego podejścia twierdzą, że intelekt ludzki ukształtował się w takiej postaci jak obecnie dzięki ewolucji. Główną trudność w tym podejściu stanowi fakt, że mechanizmy naturalnej ewolucji są bardzo słabo poznane.

Trzecie i jak dotychczas najlepsze podejście do problematyki sztucznej inteligencji, to podejście z pozycji programowania heurystycznego. Przy pomocy heurystyk modelowane jest inteligentne zachowanie się człowieka w procesie rozwiązywania zadań. Na bazie programowania heurystycznego rozwiązano i w dalszym ciągu rozwiązuje się szereg problemów z dziedziny sztucznej inteligencji.

Ostatnio badania w dziedzinie sztucznej inteligencji i programowania heurystycznego rozwijają się na świecie bardzo inten-

sywnie[1/]. Swiadczy o tym chociaż by fakt, że średnio co dwa lata organizowane są światowe konferencje poświęcone sztucznej inteligencji.
Dla kontaktów i wymiany informacji w tym zakresie powołano międzynarodową radę /International Joint Council on Artificial Intelligence/. W USA przy Association for Computing Machinery /ACM/ - towarzystwie skupiającym większość uczonych pracujących w dziedzinie elektronicznej techniki obliczeniowej - powstała specjalna grupa skupiająca badaczy zajmujących się sztuczną inteligencją. Wydają oni pod auspicjami ACM interesujący i znany w świecie biuletyn. Problematyka sztucznej inteligencji i programowania heurystycznego jest bardzo często poruszana w takich wydawnictwach ACM jak: Computing Reviews, Computing Surveys, Journal of the ACM, Communications of the ACM. Na uniwersytetach amerykańskich odbywają się systematyczne wykłady i seminaria dotyczące sztucznej inteligencji.
Bardzo duży rozmach osiągnęły badania w omawianej dziedzinie w ZSRR. Znajduje to wyraz we wzroście ilości publikacji, zarówno monografii [6] , [150] jak i artykułów w znanych czasopismach naukowych: Kibiernetika, Ekonomika i matematiczeskije mietody, Awtomatika i tielemiechanika. Poza tym dwa kolejne ogólnozwiązkowe sympozja zorganizowane przez Radę Naukową przy Akademii Nauk ZSRR organizującą i koordynują-

---

1/ Znany amerykański teoretyk w dziedzinie programowania heurystycznego i sztucznej inteligencji M.Minsky twierdzi że "znajdujemy się na progu epoki, w której zaznaczy się silny a może nawet dominujący wpływ inteligentnych maszyn zdolnych do rozwiązywania problemów [124] .

cą badania w dziedzinie cybernetyki /Naucznyj sowiet po kompleksnoj problemie "kibiernetika" / były poświęcone w całości problemom sztucznej inteligencji - VI sypozjum odbywające się Tbilisi w dniach 1-3 listopada 1972 oraz wspomniane już VII sympozjum. Ponadto w niektórych uczelniach prowadzone są już wykłady i seminaria /np. Uniwersytet Moskiewski - doc. W. Brabrin, Moskiewski Instytut Ekonomiczno-Statystyczny - doc. J. Aleksandrow /.

W naszym kraju problematyką sztucznej inteligencji i programowania heurystycznego zajmuje się niewiele osób. Badania nie są koordynowane, mało jest też publikacji w języku polskim.

Ostatnio można zanotować pewien przełom. We wrześniu 1974 r. odbyło się w Warszawie I krajowe sympozjum p.n. "Metody heurezy" i dokładnie za rok we wrześniu 1975r II krajowe sympozjum. Referaty przedstawione na tych sympozjach świadczą, że osiągnięcia nasze są jeszcze skromne, [128] , [129] .

Prowadzone obecnie na świecie badania koncentrują się przede wszystkim na problemach rozwiązywania przy pomocy programowania heurystycznego złożonych problemów sterowania i planowania zarówno na szczeblu mikroekonomicznym jak i makroekonomicznym, rozumienia przez maszyny naturalnego języka, rozpoznawania obrazów. Trwają też badania nad kierowaniem robotów przy pomocy programów heurystycznych. Jeżeli chodzi o problemy teoretyczne, to dotyczą one szczególnie poznania istoty tworzenia heurystyk przez człowieka w procesie rozwiązywania zadania oraz rozszerzenia środków matematycznych służących do opisu działalności intelektualnej człowieka.

3. PREZENTACJA METODY PRIORYTETÓW

3.1. Uwagi wstępne

Na podstawie analizy szeregu pozycji literatury, badań nad sposobami rozwiązywania w praktyce pewnych konkretnych zadań z teorii przedsięwzięć czasowych oraz eksperymentów "ręcznych" a także z wykorzystaniem komputerów Odra 1204 i Odra 1305 okazało się, że dla rozwiązywania szerokiej klasy zadań z teorii przedsięwzięć czasowych może być zaproponowana pewna metoda heurystyczna a właściwie pewien s c h e m a t p o s t ę p o w a n i a nazwany w tej pracy m e t o d ą p r i o r y t e t ó w. Zastosowanie tego schematu postępowania pozwala znajdować w realnym przedziale czasu rozwiązanie dopuszczalne dla zadań o dużych rozmiarach a więc dla zadań spotykanych w praktyce.

Poniżej zostanie omówiona idea metody priorytetów a w następnych rozdziałach zostaną zaprezentowane zastosowania tej metody do rozwiązywania konkretnych zadań.

3.2. Problem konfliktu

Rozwiązanie zadania w teorii przedsięwzięć czasowych sprowadza się jak wiadomo, do wyznaczania harmonogramu czyli znalezienia macierzy T^o /por. wzór (5) /. Przy pomocy proponowanego w tej pracy schematu postępowania, wartości poszczególnych elementów macierzy T^o będą określane sekwencyjnie, przy czym w każdym kroku określa się wartości jednego lub więcej elementów macierzy T^0. Element macierzy T^0 czyli wielkość t^0_{ij} oznacza jak wiadomo czas rozpoczęcia czynności d_{ij}.

Przyjmijmy następujące definicje.

Powiemy, że czynność $d_{ij} \in D_1$ / $i = 1,2,\ldots, m_1$ / jest gotowa do wykonania w r-tym kroku, jeżeli dla czynności $d'_{ij} \in D_1$ / $i = 1,2,\ldots, m_1$ / takiej, że $d'_{ij} R d_{ij}$ /$d'_{ij} \neq d_{ij}$ / został określony moment jej rozpoczęcia w kroku o numerze mniejszym od r. Przy czym, jeżeli dla czynności d_{ij} nie istnieje taka czynność d'_{ij}, wówczas czynność d_{ij} jest gotowa do wykonania w dowolnym kroku / $r = 1,2,3, \ldots$ /.

Jeżeli w r-tym kroku dla pewnej czynności d_{ij} gotowej do wykonania, została określona wielkość t^0_{ij} będziemy mówili, że w r-tym kroku czynność d_{ij} została wprowadzona do harmonogramu.

W r-tym kroku / $r = 1,2,3,\ldots$ / może wystąpić tzw. konflikt wtedy, gdy spośród elementów pewnego zbioru $U(r)$ należy wybrać pewną ilość elementów mniejszą od liczebności tego zbioru. Elementy te będą tworzyły podzbiór $\bar{U}(r)$.

Algorytm w wyniku realizacji którego będzie określony podzbiór $\bar{U}(r)$, a tym samym zostanie zlikwidowany konflikt, będziemy nazywali sposobem likwidacji konfliktu.

W dalszym ciągu rozważane będą trzy typy konfliktów. Konflikt pierwszego i drugiego typu wystąpi w tzw. *postępowaniu I* natomiast konflikt trzeciego typu w *postępowaniu II*. Zbiór $U(r)$ w konflikcie pierwszego typu będziemy oznaczać symbolem $U_1^1(r)$, w konflikcie drugiego typu symbolem $U_2^1(r)$ oraz w konflikcie trzeciego typu symbolem $U_3^2(r)$, a podzbiór $\bar{U}(r)$ odpowiednio: $\bar{U}_1^1(r)$, $\bar{U}_2^1(r)$, $\bar{U}_3^2(r)$.

W celu określenia poszczególnych elementów macierzy T^0 może być zastosowane postępowanie I bądź postępowanie II. Ogólnie powiemy, że z postępowaniem I mamy do czynienia wtedy, gdy w r-tym kroku / $r = 1,2,3,\ldots$ / do harmonogramu wprowadzana jest tylko jedna czynność, natomiast z postępowaniem II wtedy, gdy do harmonogramu wprowadzana jest jedna lub więcej czynności i dodatkowo każdemu r-temu krokowi odpowiada pewien moment czasu.

Rozpatrzmy postępowanie I.

W każdym r-tym kroku / $r = 1,2,3,\ldots$ / należy utworzyć zbiór czynności gotowych do wykonania. Spośród czynności należących do utworzonego zbioru należy wybrać jedną czynność.

Jeżeli liczebność tego zbioru jest większa od jedności powiemy wtedy, że w r-tym kroku występuje *konflikt pierwszego typu*, a zbiór ten jest zbiorem $U_1^1(r)$. Likwidacja tego konfliktu będzie więc polegała na wybraniu ze zbioru $U_1^1(r)$ jednego elementu 1/.

1/ Aby nie komplikować dalszych rozważań przyjmujemy, że liczba rozpatrywanych zasobów równa się jedności / $k=1$ /. W przeciwnym razie zbiór $U_1^1(r)$ oraz konflikt pierwszego typu w każdym kroku należałoby także rozpatrywać w odniesieniu do poszczególnych rodzajów zasobów. Uwaga ta dotyczy również konfliktu drugiego i trzeciego typu.

Dla wybranej, w wyniku likwidacji konfliktu pierwszego typu, czynności należy teraz określić moment jej rozpoczęcia. Aby to wykonać trzeba utworzyć zbiór możliwych momentów rozpoczęcia tej czynności. Do tego zbioru będą należały wszystkie takie momenty, które spełniają ograniczenie bilansowe oraz ograniczenia dodatkowe.

Gdy liczebność zbioru możliwych momentów rozpoczęcia jest większa od jedności wtedy mamy do czynienia z k o n f l i k t e m d r u g i e g o t y p u a zbiór ten jest zbiorem $U_2^1(r)$. Likwidując konflikt drugiego typu według zadanego sposobu likwidacji tzn. wybierając w zbiorze $U_2^1(r)$ jeden element, określamy moment t_{ij}^0 dla czynności wybranej podczas likwidacji konfliktu pierwszego typu. W ten sposób w r-tym kroku czynność ta zostanie wprowadzona do harmonogramu i należy przejść do kroku $r+1$.
Postępowanie w kroku $r+1$-szym jest identyczne z postępowaniem opisanym dla r-tego kroku przy czym gdy w $r+1$-szym kroku zbiór czynności gotowych do wykonania jest zbiorem pustym, wtedy zadanie zostało rozwiązane.

Drugi sposób wprowadzania czynności do harmonogramu czyli postępowanie II jest następujące:

Każdemu krokowi odpowiada tu pewien moment czasu, który oznaczymy symbolem[1/] h_r. W r-tym kroku / $r = 2,3,\ldots$ /

1/ Można też przyjąć, że będzie to przedział $[h_r^0, h_r^1]$ taki, że $h_r^0 = h_{r-1}^1$

wartość h_r jest liczona według następującego wzoru $h_r = h_{r-1} + \Delta h_r$ przy czym wzory dla wartości Δh_r / dla $r = 2,3,\ldots$ / oraz h_1 zależą od ograniczeń w każdym konkretnym zadaniu i podaje się je w dalszej części przy omawianiu tych zadań.

W r-tym kroku ustala się zbiór czynności gotowych do wykonania. W tym przypadku ulegnie jednak modyfikacji definicja czynności gotowej do wykonania.

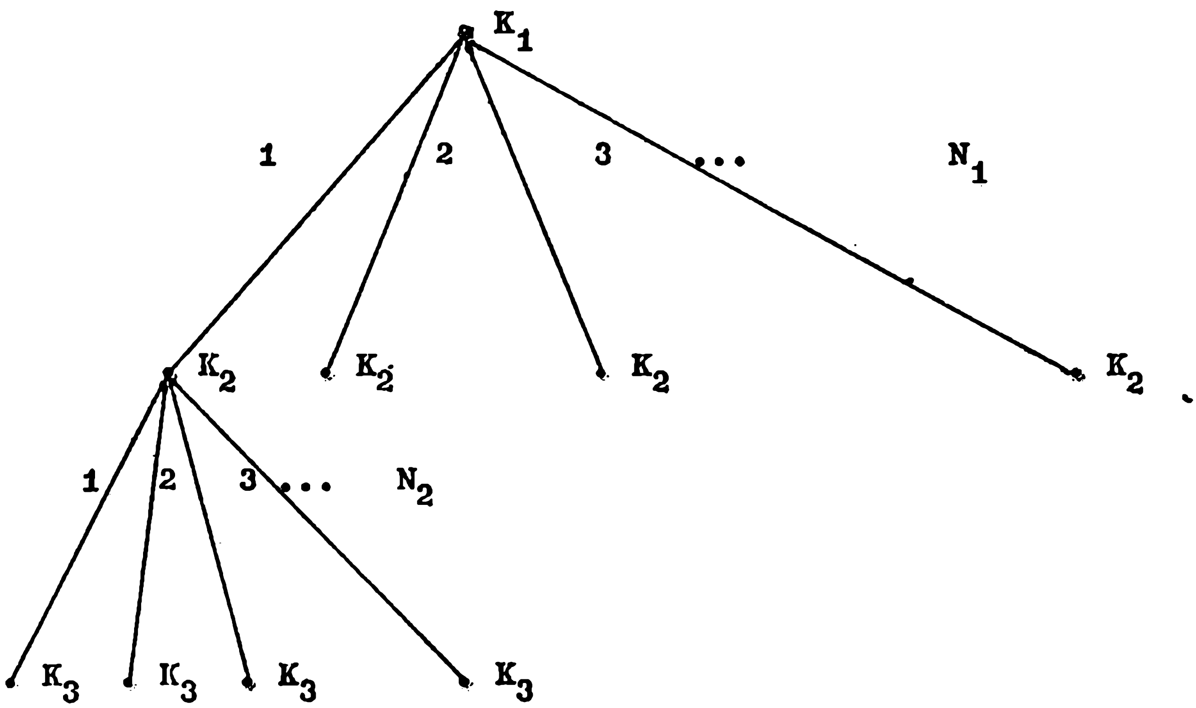
Powiemy, że czynność jest gotowa do wykonania w r-tym kroku, jeżeli czynności które ją poprzedzają / tzn. są z nią w relacji R / zostały wprowadzone do harmonogramu w kroku mniejszym od r oraz moment zakończenia tych czynności jest mniejszy lub równy wartości h_r. Gdy natomiast rozpatrywanej czynności nie poprzedzają żadne inne czynności, to jest ona gotowa do wykonania w dowolnym kroku / $r = 1,2,3,\ldots$ /.

Wszystkim czynnościom należącym w r-tym kroku do zbioru czynności gotowych do wykonania można przyporządkować momenty rozpoczęcia równe wartości h_r wtedy, gdy spełnione jest ograniczenie bilansowe. W przeciwnym przypadku mamy do czynienia z konfliktem trzeciego typu, a zbiór czynności gotowych do wykonania będzie zbiorem $U_3^2(r)$. Likwidacja tego konfliktu polega na określeniu w zbiorze $U_3^2(r)$ takiego podzbioru czynności $\bar{U}_3^2(r)$, dla którego jest spełnione ograniczenie bilansowe, a więc moment rozpoczęcia dla czynności należących do podzbioru $\bar{U}_3^2(r)$ stanowi wtedy wielkość h_r. Ta[illegible]c określając podzbiór $\bar{U}_3^2(r)$ wprowadzamy jednocześnie do harmonogramu czynności należące do tego podzbioru. Tym samym można wtedy przejść do kroku

r + 1 . Postępowanie w kroku r + 1 jest identyczne jak w r-tym kroku. Zadanie jest rozwiązane gdy zbiór czynności gotowych do wykonania jest zbiorem pustym.
Dla określenia podzbioru $\bar{U}_3^2(r)$ musi być zadany pewien algorytm postępowania czyli sposób likwidacji konfliktu typu trzeciego. Podstawową wielkością wykorzystywaną do likwidacji omawianego konfliktu a także konfliktu typu pierwszego oraz typu drugiego jest tzw. p r i o r y t e t.
Zagadnienie stosowania priorytetu do likwidacji konfliktów stanowi przedmiot rozważań następnego paragrafu.

O zaprezentowanych dwóch postępowaniach przy wprowadzaniu czynności do harmonogramu nie można powiedzieć w sposób jednoznaczny, które z nich daje lepsze wyniki. Wyższość jednego postępowania nad drugim można byłoby wykazać na podstawie licznych eksperymentów. Trzeba jednak wyraźnie podkreślić, że przeprowadzenie dużej ilości eksperymentów, w wypadku realnych zadań rozpatrywanych w teorii przedsięwzięć czasowych, jest procesem bardzo pracochłonnym i wymaga zaangażowania dużego zespołu ludzi, środków itp.

Proces konstruowania rozwiązania zarówno w postępowaniu I jak i w postępowaniu II można przedstawić przy pomocy tzw. d r z e w a k o n f l i k t ó w / rys. 4 / . Drzewo konfliktów ilustruje wszystkie konflikty z jakimi mamy do czynienia generując rozwiązanie oraz wszystkie możliwe sposoby likwidacji tych konfliktów. Oznaczenia przy krawędziach grafu określają numery sposobów likwidacji konfliktu, natomiast oznaczenia wierzchołków grafu przedstawiają konflik-



Rys. 4 . Fragment drzewa konfliktów

ty spotykane w procesie konstruowania rozwiązania. Krawędzie łączą dwa kolejne konflikty.

Przyjmijmy, że rys. 4 przedstawia fragment drzewa konfliktów przy stosowaniu postępowania I. Jak widać na rysunku, likwidując konflikt K_1 typu pierwszego na N_1 sposobów otrzymuje się N_1 konfliktów K_2 typu drugiego różniących się od siebie sposobem likwidacji poprzedniego konfliktu. W procesie konstruowania rozwiązania likwidujemy jak wiadomo kolejno powstające konflikty. Na drzewie konfliktów będziemy się wtedy posuwali wzdłuż odpowiednich krawędzi grafu. Każdy taki łańcuch opisuje proces konstruowania jednego z możliwych rozwiązań. Wiszące węzły grafu prezentują możliwe rozwiązania zagadnienia.

3.3. F u n k c j a p r i o r y t e t u i s p o s o b y l i k w i d a c j i k o n f l i k t u

Każdemu elementowi ze zbiorów $U_1^1(r)$, $U_2^1(r)$ przyporządkujemy pewną wielkość zwaną priorytetem, którą będziemy oznaczać symbolem w. Sposób wyliczania wielkości w zadany jest w postaci funkcji. Funkcję tę będziemy nazywali f u n k c j ą p r i o r y t e t u i oznaczali symbolem G.

$$G : R^n \longrightarrow R \tag{6}$$

Wartości funkcji G określone są w przestrzeni $(x_1,\ldots,x_n)$ tzn.

$$w = G(X) \quad ; \quad X \in R^n .$$

Poszczególne składowe wektora X oznaczają charakterystyki liczbowe danego elementu należącego do zbioru $U_1^1(r)$, $U_2^1(r)$ lub $U_3^2(r)$. Ponieważ w dalszym ciągu rozpatrywać będziemy różne warianty funkcji G, więc oznaczać je będziemy symbolami $G_1, G_2, \ldots$ a ich wartości odpowiednio $w_1, w_2, \ldots$

Poniżej zostaną omówione sposoby likwidacji konfliktów. Wyróżnimy tutaj sposób d e t e r m i n i s t y c z n y oraz sposób l o s o w y. Jako pierwszy omówimy sposób deterministyczny, który ze względu na prostotę i nieskomplikowane obliczenia znajduje szczególne zastosowanie przy rozwiązywaniu tzw. zadań o dużych rozmiarach.

Sposób deterministyczny opiszemy następująco.
Dla każdego elementu zbioru $U_1^1(r)$; $U_2^1(r)$ lub $U_3^2(r)$ według zadanej funkcji priorytetu należy wyliczyć wielkość w. Następnie zbiór trzeba uporządkować rosnąco lub malejąco ze względu na tę wielkość. W przypadku gdy rozpatrujemy zbiór $U_1^1(r)$ lub $U_2^1(r)$ / czyli mamy do czynienia z konfliktem typu pierwszego lub typu drugiego / wówczas w uporządkowanym zbiorze wybierany jest pierwszy element. W ten sposób odpowiedni konflikt należy uważać za zlikwidowany. Gdy natomiast mamy do czynienia z konfliktem trzeciego typu czyli rozpatrujemy zbiór $U_3^2(r)$ wtedy z uporządkowanego zbioru $U_3^2(r)$ wybierane są kolejne elementy, które będą tworzyły podzbiór $\bar{U}_3^2(r)$ przy czym kolejny element uporządkowanego zbioru $U_3^2(r)$ nie będzie należał do podzbioru $\bar{U}_3^2(r)$, jeżeli włącznie go spowoduje naruszenie ograniczenia bilansowego.

Opisane postępowanie, zarówno w przypadku konfliktu typu pierwszego, konfliktu typu drugiego oraz konfliktu typu trzeciego jest możliwe wówczas, gdy wszystkie wartości w odpowiadające poszczególnym elementom zbioru $U_1^1(r)$, $U_2^1(r)$ lub $U_3^2(r)$ są różne. W przeciwnym przypadku będzie to zbiór nie w pełni uporządkowany tj. w zbiorze tym występują przynajmniej dwa takie elementy, o których nie można powiedzieć, który z nich jest elementem wcześniejszym, wtedy likwidacja konfliktu w sposób jednoznaczny jest niemożliwa. Aby temu zapobiec zaproponujemy użycie kolejno kilku funkcji priorytetu. Sens takiego postępowania jest następujący.

Załóżmy, że dany jest ciąg funkcji priorytetu $G_1, G_2, \ldots, G_l$. Przyjmijmy, że zbiór $U_1^1(r)$, $U_2^1(r)$ lub $U_3^2(r)$ został uporządkowany według priorytetu wyliczonego na podstawie funkcji G_1 czyli wielkości w_1. Jeżeli okaże się, że likwidacja konfliktu w sposób jednoznaczny jest niemożliwa wtedy dla tych podzbiorów, którym odpowiadają jednakowe wartości w_1 należy zastosować funkcję G_2 tj. wyliczyć dla elementów takich podzbiorów wartości w_2 a następnie elementy w poszczególnych podzbiorach uporządkować według tych wartości. Gdy w wyniku pewne podzbiory nie zostaną w pełni uporządkowane wtedy dla elementów tych podzbiorów, dla których wartości w_2 są jednakowe należy zastosować funkcję G_3 itd. aż do zlikwidowania niejednoznaczności.

O losowym sposobie likwidacji konfliktu powiemy wtedy, gdy elementy ze zbioru $U_1^1(r)$, $U_2^1(r)$ lub $U_3^2(r)$ wybierane są losowo z jednakowym prawdopodobieństwem bądź z częstotliwością proporcjonalną do wartości funkcji priorytetu wyliczonej dla każdego elementu rozpatrywanego zbioru.
Losowy wybór z częstotliwością proporcjonalną do wartości funkcji priorytetu przedstawimy na przykładzie konfliktu typu pierwszego.
Niech

$$U_1^1(r) = \left\{d_{i_1 j_1}, d_{i_2 j_2}, \ldots, d_{i_k j_k}, \ldots, d_{i_d j_d}\right\} \quad (7)$$

Dla k-tego elementu / $k = 1,2,\ldots,d$ / obliczamy wartość priorytetu $w_{i_k j_k}$ przy czym funkcja priorytetu musi być taka aby wartości $w_{i_k j_k}$ były większe od zera.

Następnie obliczamy wielkość $\bar{w}(k)$ w sposób następujący:

$$\bar{w}(k) = \frac{w_{i_k j_k}}{\sum w_{i_k j_k}} \qquad (\bar{8})$$

gdzie sumowanie odbywa się po wszystkich wskaźnikach elementów zbioru $U_1^1(r)$.

Ponieważ zgodnie z założeniem $w_{i_k j_k} > 0$ więc $0 < \bar{w}(k) \leq 1$.

Każdemu $d_{i_k j_k} \in U_1^1(r)$ określamy przedział $[z_k, y_k)$ taki, że $z_1 = 0$, $z_k = y_{k-1}$ i $y_k = \bar{w}(k) + z_k$.

Aby określić, który element zbioru $U_1^1(r)$ będzie wybrany należy wygenerować liczbę losową z przedziału $[0,1]$. Jeżeli z generatora liczb losowych otrzymujemy liczbę β taką, że

$$z_k < \beta \leq y_k ,$$

wtedy wybrany zostanie element $d_{i_k j_k}$.

Likwidacja konfliktu typu drugiego przebiega analogicznie. W przypadku konfliktu typu trzeciego w zbiorze $U_3^2(r)$ wybiera się jak wiadomo pewien podzbiór $\bar{U}_3^2(r)$. Poszczególne elementy zbioru $\bar{U}_3^2(r)$ określa się w wyniku opisanego wyżej postępowania z tym, że kolejny element $d_{i_k j_k}$ będzie należał do podzbioru U_3^2 (r) wtedy gdy spełnione jest ograniczenie bilansowe. W przeciwnym wypadku element $d_{i_k j_k}$ nie będzie należał do podzbioru $\bar{U}_3^2$ (r).

Przy zastosowaniu losowego sposobu likwidacji konfliktu można wybierać nie tylko elementy ze zbiorów U_1^1 (r) , U_2^1(r), U_3^2 (r) , ale także samą postać funkcji priorytetu. Przyjmijmy, że w kolejnych krokach konstruowania rozwiązania do harmonogramu można wprowadzać czynności wg jednej z następujących funkcji priorytetu:

$$G_1(X),\ G_2(X)\ ,\ldots,\ G_q(X),\ldots,\ G_k(X)\ .$$

W każdym kroku funkcję G_q (X) wybiera się losowo z jednakowym prawdopodobieństwem równym 1/k bądź też można każdej funkcji G_q(X) przyporządkować pewną wielkość $0 < p_q < 1$ taką, że

$$\sum_{q=1}^{k} p_q = 1\ .$$

Następnie każdej funkcji G_q(X) należy przyporządkować pewien przedział $[z_q, y_q)$ taki, że

$$z_1 = 0\ ,\ z_q = y_{q-1} \quad \text{oraz} \quad y_q = z_q + p_q$$

i wygenerować liczbę losową β z przedziału $[0,1]$. Wybieramy tę funkcję, dla której

$$z_q \leqslant \beta < y_q\ .$$

Wielkość p_q może być zadana bądź też ustala się ją na podstawie analizy poprzednio wygenerowanych rozwiązań. O programie, w którym dokonuje się analizy poprzednio wygenerowanych rozwiązań będziemy mówili, że jest to p r o g r a m s a m o u c z ą c y s i ę .

Losowy sposób likwidacji konfliktu może być też realizowany przy pomocy tzw. p r o g r a m u s a m o o r g a n i z u j ą c e g o s i ę. Wtedy w kolejnych krokach generowania rozwiązania analizowane są poszczególne wielkości p_q i w zależności od wyników analizy ustala się nowe wielkości.

Dodajmy, że przy pomocy programów samouczących się czy też samoorganizujących się może być również realizowany sposób deterministyczny. Wówczas w trakcie generowania rozwiązania bądź po jego wygenerowaniu ustalony jest nowy ciąg funkcji priorytetu.

Trzeba jednak zaznaczyć, że przy dzisiejszym stanie elektronicznej techniki obliczeniowej generowanie rozwiązań przy pomocy programów samouczących się bądź samoorganizujących się dla realnych zadań rozważanych w teorii przedsięwzięć czasowych nie wydaje się możliwe.

Należy też podkreślić, że stosowanie losowego sposobu likwidacji konfliktów przy rozwiązywaniu realnych zadań jest o wiele bardziej uciążliwe aniżeli stosowanie sposobu deterministycznego. Bardziej pracochłonny i skomplikowany jest wtedy proces przygotowania programów na maszynę cyfrową, a także wydłuża się czas obliczeń na maszynie.

Wydaje się, że w praktyce należy stosować mieszany sposób likwidacji konfliktów. Sens takiego postępowania jest następujący. Jeżeli zadany ciąg funkcji priorytetu nie zapewnia jednoznacznej likwidacji konfliktu wtedy stosujemy dodatkowo losowy sposób likwidacji.

Ważnym zagadnieniem jest ustalenie postaci funkcji priorytetu.
Jak już zaznaczono, duża część zadań rozważanych w teorii przedsięwzięć czasowych jest rozwiązywana "ręcznie" przez człowieka. W celu ustalenia postaci funkcji priorytetu trzeba wykorzystać, przede wszystkim, heurystyki stosowane przez człowieka w procesie "ręcznego" rozwiązywania. Uwzględni się w ten sposób doświadczenie zgromadzone w rozwiązywaniu "ręcznym". Pomocnymi przy ustalaniu postaci funkcji priorytetu są też niektóre heurystyki uniwersalne takie np. jak eksperymentowanie i uczenie się, rozumowanie przez analogię, reguły zwyczajowe itp. Analiza "ręcznego" rozwiązywania wykazuje, że człowiek posługuje się stosunkowo prostymi heurystykami. Funkcje priorytetu bazujące na tych heurystykach mają także prostą postać. Są to funkcje typu $G_1(X) = x_1$, $G_2(X) = x_1 + 2x_2$ itp.
Autor pracy, na podstawie własnych doświadczeń uzyskanych w toku eksperymentów komputerowych uważa, że właśnie takie funkcje można z powodzeniem stosować do rozwiązywania realnych zadań. Zastosowanie takich funkcji powoduje, że algorytm nie jest złożony, obliczenia na maszynie trwają krótko, a wygenerowane rozwiązania są całkowicie do przyjęcia w praktyce.

W dalszej części niniejszej pracy przy omawianiu konkretnych zadań proponowane są pewne proste wzory funkcji priorytetu. Postępowanie prowadzące do ustalenia postaci tych funkcji było następujące.
1/ Analizowano szczegółowo opisane w literaturze sposoby roz-

wiązywania podobnych zadań.

2/ W przypadku zadań występujących w operatywnym planowaniu produkcji /rozdz. 4 i rozdz. 5 / zapoznano się w Jelczańskich Zakładach Samochodowych w jaki sposób podobne zagadnienia rozwiązuje się "ręcznie". Prowadzono przy tym rozmowy z doświadczonymi pracownikami z komórek planowania i kierownictwa wydziałów dla uchwycenia toku ich postępowania. Następnie rozwiązano "ręcznie" proste i nieduże przykłady.

3/ W przypadku zadania będącego przedmiotem rozważań rozdziału 6 zapoznano się ze sposobem układania harmonogramu zajęć w Akademii Ekonomicznej we Wrocławiu. Funkcje priorytetu weryfikowano układając "ręcznie" realny harmonogram zajęć, dokonując eksperymentów w małej skali na maszynie Odra 1204 oraz eksperymentując na realnym przykładzie przy wykorzystaniu komputera Odra 1305.

Na podstawie dotychczasowych rozważań można stwierdzić, że w opisanym wyżej schemacie postępowania podstawową wielkością, według której wyznaczany jest harmonogram jest priorytet. W dalszej części pracy przedstawiony schemat postępowania będziemy nazywali m e t o d ą p r i o r y t e t ó w. Łatwo też zauważyć, że strategia stosowana w metodzie priorytetów jest strategią heurystyczną czyli metoda priorytetów jest metodą heurystyczną.

Stosując metodę priorytetów w praktyce trzeba wygenerować kilka lub kilkanaście harmonogramów używając do wyznaczenia każdego harmonogramu innego ciągu funkcji priorytetów.

Wiele wariantów harmonogramu można też uzyskać stosując jeden ciąg funkcji priorytetu, jednak wtedy musi być zastosowany mieszany bądź losowy sposób likwidacji konfliktów. Wśród wygenerowanych harmonogramów należy wybrać hrmonogram n a j l e p s z y ze względu na przyjęte kryterium.

Kryterium, według którego spośród zbioru wygenerowanych rozwiązań wybieramy rozwiązanie najlepsze będziemy nazywali w dalszym ciągu k r y t e r i u m o c e n y. Jest rzeczą oczywistą, że nie istnieje uniwersalne kryterium oceny, które można zastosować w każdym konkretnym zadaniu. Dla każdego zadania można natomiast ustalić pewien zbiór kryteriów oceny. O tym jakie kryterium oceny zostanie przyjęte, w każdym przypadku musi zadecydować człowiek. W dalszym ciągu pracy, przy omawianiu konkretnych zadań, zostanie zaproponowany szereg kryteriów oceny wygenerowanych rozwiązań.

4. WYZNACZANIE HARMONOGRAMU DLA ZAGADNIEŃ Z SZEREGOWĄ KOLEJNOŚCIĄ CZYNNOSCI

4.1. Sformułowanie problemu

4.1.1. Uwagi wstępne

Problem wyznaczania harmonogramu dla zagadnień z szeregową kolejnością czynności zostanie omówiony na przykładzie wyznaczania harmonogramu produkcji dla wydziału obróbczego. Określane ono bywa również wyznaczaniem harmonogramu produkcji detali bądź zadaniem o detalach i stanowiskach.
Na początku sformułujemy zadanie, z którym można się często spotkać w realnych warunkach. Zadanie to nie ujmuje oczywiście wszystkich ograniczeń, które można spotkać w praktyce lecz stanowi pewnego rodzaju syntezę a więc zawiera ograniczenia najbardziej istotne. Następnie zostaną rozpatrzone pewne sformułowania rozważane w literaturze.

Zanim przedstawimy wymienione zadania podamy definicje ważniejszych pojęć.
W y r o b e m będzie pewien przedmiot złożony z jakichś innych przedmiotów, zwanych składnikami, te ostatnie złożone są z innych składników itd.
Pod pojęciem d e t a l u będziemy rozumieli taki składnik wyrobu, który nie zawiera w sobie żadnych innych składników. Aby detal lub wyrób został wykonany musi być zrealizowany szereg operacji.
Przez o p e r a c j ę będziemy rozumieli zespół czynności, których wykonanie powoduje zmianę co najmniej jednej z ta-

kich charakterystyk detalu /wyrobu/ jak: skład, kształt, wymiar.

Każda operacja może być wykonana na określonym s t a n o w i s k u r o b o c z y m , pod którym należy rozumieć powierzchnię produkcyjną wraz z niezbędnymi narzędziami do pracy.

O d c i n k i e m p r o d u k c y j n y m będziemy nazywali zbiór stanowisk roboczych, który zapewnia realizację przewidzianego do wykonania zbioru detali /wyrobów/.

Najczęściej detale wykonywane są w tzw. p a r t i a c h. Pod tym pojęciem określa się pewną ilość jednakowych detali, które poddawane są kolejnym operacjom w takiej samej liczbie sztuk.

4.1.2. Zadanie podstawowe

Zadanie to zostanie sformułowane w sposób następujący. Na pewnym odcinku produkcyjnym należy wykonać m_1 detali / albo partii detali /. Poszczególne detale będą oznaczone symbolem b_i / $i = 1,2,\ldots,m_1$ /. Rozpatrywany odcinek produkcyjny składa się z K stanowisk roboczych. Stanowiska robocze są ponumerowane kolejnymi liczbami naturalnymi $1,2,\ldots,k,\ldots,K$. Przy czym stanowiska o takim samym zakresie wykonywanych prac łączone są w grupy, które ponumerowane są liczbami naturalnymi $1, 2,\ldots,l,\ldots,m_2$. Każda l-ta grupa stanowisk składa się z q_l stanowisk, gdzie $1 \leq q_l \leq K$. Można więc zapisać, że

$$\acute{K} = \sum_{l=1}^{m_2} q_l \ . \qquad (9)$$

Wykonanie i-tego detalu wymaga wykonania na odpowiednich stanowiskach szeregu operacji. Operacje te będą oznaczone symbolem d_{ij} / $j = 1,2,\ldots,n_i$ /. Każda operacja może być jednoznacznie scharakteryzowana za pomocą pary liczb

$$d_{ij} = (l_{ij}, t_{ij}) \qquad (10)$$

gdzie:

l_{ij} - numer grupy stanowisk, na której może być wykonana operacja d_{ij},
$l = 1,2,\ldots,m_2$

t_{ij} - czas trwania operacji d_{ij} na dowolnym stanowisku roboczym z grupy l_{ij}.

Aby detal b_i został wykonany, poszczególne operacje d_{ij} / $j = 1,2,\ldots,n_i$ / muszą być wykonane w ściśle określonej kolejności. Dla każdego detalu b_i zadany jest wtedy wektor $(d_{i1},d_{i2},\ldots,d_{in_i})$. Wektor ten nazywa się m a r s z r u t ą t e c h n o l o g i c z n ą. Pod marszrutą technologiczną dla i-tego detalu będziemy też rozumieli wektor charakteryzujący kolejność w jakiej są wykonywane operacje na poszczególnych grupach stanowisk czyli kolejność w jakiej i-ty detal w procesie obróbki przechodzi przez poszczególne grupy stanowisk. Marszrutę technologiczną oznaczymy symbolem L_i

$$L_i = (l_{i1}, l_{i2}, \ldots, l_{in_i}) \ . \qquad (11)$$

Marszruty technologiczne dla wszystkich detali b_i wykonywanych w rozpatrywanym odcinku produkcyjnym mogą być zadane w postaci macierzy L:

$$L = [l_{ij}] \quad i = 1,2,\ldots,m \quad ; \quad j = 1,2,\ldots,m_3 \qquad (12)$$

gdzie $m_3 = \max\limits_i n_i$, przy czym jeśli $n_i < m_3$, to $l_{ij} = 0$ dla $j > n_i$.

Analogicznie, czasy trwania operacji wykonywanych na poszczególnych m_1 detalach można zapisać w postaci macierzy T:

$$T = [t_{ij}] \quad i = 1,2,\ldots,m \quad ; \quad j = 1,2,\ldots,m_3 \qquad (13)$$

gdzie $t_{ij} = 0$ gdy $l_{ij} = 0$.

Wykonując operacje na poszczególnych stanowiskach, oprócz wymienionych powyżej warunków należy uwzględnić następujące ograniczenia:

1° Żadna operacja po rozpoczęciu nie może być przerwana czyli

$$t^1_{ij} = t^o_{ij} + t_{ij} \quad ,$$

gdzie t^o_{ij} oznacza moment rozpoczęcia operacji d_{ij}, natomiast t^1_{ij} oznacza moment zakończenia operacji d_{ij}.

2° Każda następna operacja wykonywana na i-tym detalu może być zaczęta po zupełnym zakończeniu poprzedniej tzn. dla dwóch operacji d_{ij_1} oraz d_{ij_2} , jeżeli $t^o_{ij_1} > t^o_{ij_2}$

to $t^1_{ij_1} \geqslant t^o_{ij_2}$.

3° Na jednym stanowisku roboczym nie może być jednocześnie wykonywana więcej jak jedna operacja tzn. dla dwóch operacji $d_{i_1j_1}$ oraz $d_{i_2j_2}$ wykonywanych na tym samym stanowisku jeżeli $t^o_{i_1j_1} < t^o_{i_2j_2}$, to $t^1_{i_1j_1} \leqslant t^o_{i_2j_2}$.

Problem będzie polegał na tym aby określić dla wszystkich operacji d_{ij} / $i = 1,2,\ldots,n$; $j = 1,2,\ldots,n_i$ / stanowiska robocze, na których będą wykonywane operacje oraz momenty rozpoczęcia t^o_{ij} na tych stanowiskach, w ten sposób, aby były uwzględnione wszystkie podane wyżej ograniczenia.

W wyniku otrzymamy d o p u s z c z a l n y h a r m o n o g r a m p r o d u k c j i d e t a l i.

Dopuszczalny harmonogram można przedstawić 1/ w postaci macierzy T^o

$$T^o = \left[(t^o_{ij},\ k_{ij})\right] \qquad i = 1,2,\ldots,m \ ; \ j=1,2,\ldots,m_3 \qquad (14)$$

gdzie (t^o_{ij}, k_{ij}) - uporządkowana para zapisana na przecięciu i-tego wiersza i j-tej kolumny, wartości której oznaczają, że j-ta operacja i-tego detalu rozpoczyna się w momencie t^o na k-tym stanowisku.

Spośród zbioru rozwiązań dopuszczalnych należy wybrać takie, dla którego zadana funkcja kryterium osiąga ekstremum. Postacie funkcji kryterium rozpatrywane są w § 4.2.

W literaturze można spotkać się z innymi sformułowaniami tego problemu. W dalszym ciągu niniejszego rozdziału podamy przykłady innych sformułowań a ponadto opiszemy ważniejsze modyfikacje zadania postawowego.

1/ Należy zaznaczyć, że istnieją różne formy przedstawiania harmonogramu /np. wykresy Gantta, chronomogramy, tzw. wektorowe funkcje czasu/. Szeroko zagadnienie to zostało opisane w pracach [164], [177], [179]. Wybór określonej formy zależy od celów jakim ta forma ma służyć.

4.1.3. Inne sformułowania

Do klasycznych już sformułowań należy zadanie Johnsona. Opiszemy je w następujący sposób:

Pewien odcinek produkcyjny składa się z n stanowisk roboczych. Na odcinku tym należy wykonać m detali przy czym każdy detal b_i / $i = 1,2,\ldots,m$ / powinien być poddawany obróbce kolejno na $1,2,\ldots,j,\ldots,n$ stanowiskach. Dla każdego i-tego detalu / $i = 1,2,\ldots,m$ / podany jest czas obróbki $t_{ij} > 0$ / $j = 1,2,\ldots,n$ / na każdym stanowisku. Ponadto pozostają w mocy ograniczenia 1,2,3 z punktu 4.1.2.

Zadanie Johnsona będzie polegało na znalezieniu takiej sekwencji detali $(b_{i_1}, b_{i_2}, \ldots, b_{i_m})$, przy której czas obróbki wszystkich detali jest najmniejszy.

Jak widać w zadaniu tym wszystkie detale mają jednakowe marszruty technologiczne oraz kolejność obróbki detali na każdym stanowisku jest taką sama. Liczba dopuszczalnych rozwiązań, spośród których należy wybrać sekwencję optymalną wynosi $m!$. Johnson zaproponował bardzo prosty algorytm rozwiązania tego zadania ale dla przypadku gdy $n = 2$.
Próby uogólnienia algorytmu Johnsona nie przyniosły jak dotychczas pozytywnych rezultatów.

Bardzo często rozpatrywany jest uproszczony wariant zadania Johnsona tzw. zadanie jednostanowiskowe [32], [35], [177]. W zadaniu tym przyjmuje się, że $n = 1$. Ponieważ czas obróbki wszystkich detali w tym wypadku jest zawsze taki sam niezależnie od tego w jakiej sekwencji detale będą poddawane obróbce, jako kryterium optymalności przyjmuje się tutaj np. minimalizację kosztów obróbki detali, czy też minimalizację czasu oczekiwania detali na obróbkę.

Można się też spotkać z bardziej złożonym sformułowaniem zadania Johnsona, w którym przyjmuje się, że kolejność obróbki detali na każdym stanowisku jest różna. W tym przypadku liczba dopuszczalnych rozwiązań jest równa $(m!)^n$.
Dla zilustrowania jak pracochłonne jest poszukiwanie dla takiego przypadku wszystkich rozwiązań dopuszczalnych, przytoczymy następujący przykład [171] .
Niech $n = 5$ oraz $m = 5$. Wtedy liczba dopuszczalnych rozwiązań wynosi 25 mld. Jeżeli wygenerowanie i ocena jednego rozwiązania zajmowałaby na maszynie cyfrowej 1 sek., to wszystkie rozwiązania zostałyby wygenerowane w ciągu ośmiu stuleci nieprzerwanej pracy maszyny.

Zadanie Johnsona należy do najprostszych sformułowań problemu rozpatrywanego w tym rozdziale oraz najczęściej nie znajduje odzwierciedlenia w praktyce. Większość matematyków-numeryków podejmując badania nad konstrukcją optymalnych algorytmów, pozwalających znaleźć rozwiązania w krótkim okresie czasu bez przeliczania wszystkich wariantów, bierze za podstawę zadanie Johnsona. Zadanie to stanowi dla nich swego rodzaju poligon doświadczalny.

Oprócz zadania Johnsona w literaturze rozpatrywane jest często inne zadanie zwane zadaniem Gifflera-Thompsona. Jest ono przedmiotem rozważań min. w pracach [95] , [171] . Zadanie to formułuje się w sposób następujący.

Na odcinku produkcyjnym składającym się z n różnych stanowisk roboczych trzeba wykonać m detali. Podana jest macierz kolejności obróbki każdego detalu na odpowiednich stanowiskach

$$P = [p_{ij}] \qquad i = 1,2,\ldots,m \quad ; \quad j = 1,2,\ldots,n \quad ,$$

gdzie p_{ij} oznacza kolejność w jakiej będzie poddawany obróbce i-ty detal na j-tym stanowisku. Każdy i-ty wiersz macierzy P określa nam marszrutę technologiczną i-tego detalu. Jeżeli $p_{ij_1} = 1$ wówczas i-ty detal rozpoczyna obróbkę na j_1 stanowisku, następnie przechodzi na stanowisko dla którego $p_{ij_2} = 2$.

Gdy $p_{ij} = 0$ oznacza to, że detal nie jest poddawany obróbce na j-tym stanowisku. Dana jest także macierz T

$$T = [t_{ij}] \qquad i = 1,2,\ldots,m \; ; \quad j = 1,2,\ldots,n$$

gdzie t_{ij} oznacza czas trwania obróbki i-tego detalu na j-tym stanowisku, przy czym jeżeli $p_{ij} = 0$ wtedy również $t_{ij} = 0$.

Rozwiązanie tego zadania będzie polegało na określeniu, dla każdego i-tego detalu poddawanego obróbce na j-tym stanowisku, momentu rozpoczęcia tej obróbki w ten sposób aby otrzymany w wyniku harmonogram był optymalnym ze względu na minimalny czas obróbki wszystkich detali.

Jak łatwo zauważyć zadanie Gifflera-Thompsona nie zawiera już szeregu uproszczeń przyjętych zadaniu Johnsona. Jed
ważne warunki jak obecność na rozpatrywanym odcinku produkcyj-

nym stanowisk zamiennych /równoważnych/ czyli tzw. grup stanowisk oraz możliwość powrotu detalu w kolejnej operacji na to samo stanowisko, w zadaniu tym nie zostały uwzględnione.

4.1.4. Modyfikacje zadania podstawowego .

W niniejszym punkcie podamy kilka ważniejszych naszym zadaniem ograniczeń nie ujętych w zadaniu podstawowym rozpatrywanym w punkcie 4.1.2. Będziemy je numerować liczbami rzymskimi .

I. Dla każdego k-tego stanowiska podany jest m o m e n t g o t o w o ś c i s t a n o w i s k a. Jest to moment, od którego począwszy można przydzielić na tym stanowisku operacje do wykonania. Oznaczmy go symbolem o_k $(o_k \geqslant 0)$.

II. Dla niektórych detali zadane są momenty gotowości oraz momenty zakończenia. Moment gotowości i-tego detalu / oznaczać go będziemy symbolem R_i / jest to taki moment, począwszy od którego można rozpocząć wykonywanie operacji d_{i1}.
Wtedy wielkość t^o_{i1} musi być większa lub równa wielkości R_i $(R_i > 0)$.
Z kolei moment zakończenia i-tego detalu /oznaczymy go symbolem R'_i / jest to moment, do którego powinna być zakończona operacja d_{in_i}. W tym wypadku pożądane jest aby wielkość $t^1_{in_i}$ była mniejsza lub równa wielkości R'_i $(R'_i > 0)$.

III. Wielkości R_i oraz R_i' zadane są dla każdego detalu b_i / $i = 1,2,\ldots,m_1$ /.

IV . Dla każdego k-tego stanowiska podany jest w s p ó ł - c z y n n i k w y k o n a n i a n o r m y na tym stanowisku; oznaczmy go symbolem q_k.

Wtedy

$$t_{ij} = t_{ij|k} q_k \tag{15}$$

gdzie t_{ij} oznacza rzeczywisty czas trwania operacji d_{ij} a zapis $t_{ij|k}$ wskazuje, że operacja d_{ij} jest wykonywana na k-tym stanowisku.

V. Dla każdego k-tego stanowiska podany jest czas przezbrojenia stanowiska. Zależy on od operacji $p_{i_2 j_2}$, która była wykonana przed przezbrojeniem i od operacji $p_{i_1 j_1}$, która będzie wykonana po przezbrojeniu. Oznaczać go będziemy symbolem $g_{i_1 j_1 | i_2 j_2}$.

W tym przypadku czas trwania operacji będzie wynosił:

$$g_{i_1 j_1 | i_2 j_2} + t_{i_1 j_1}.$$

VI. Niektóre partie detali b_i mogą być wykonane przy szeregowo-równoległym przebiegu operacji. Wtedy czas rozpoczęcia operacji d_{ij} przy $j > 1$ liczony jest według następującego wzoru [177]

$$t^o_{ij} \geqslant \begin{cases} t^o_{i,j-1} + V_{i,j-1} & \text{gdy} \quad t_{ij} \geqslant t_{i,j-1} \\ t^1_{i,j-1} + V_{i,j-1} - t_{ij} & \text{gdy} \quad t_{ij} < t_{i,j-1} \end{cases} \tag{16}$$

gdzie $V_{i,j-1}$ - czas wykonania partii transportowej przy realizacji operacji $d_{i,j-1}$.

Modyfikacje zadania podstawowego otrzymujemy poprzez wprowadzenie jednego lub kilku z powyższych ograniczeń.

4.2. Kryteria oceny

Zanim podamy pewne kryteria oceny wprowadzimy następujące określenia:

Czasem oczekiwania operacji d_{ij} będziemy nazywali wielkość τ_{ij} wyliczoną wg wzoru

$$\tau_{ij} = t^o_{ij} - t^1_{i,j-1} \qquad \text{dla} \quad j > 2 \tag{17}$$

przy czym $\tau_{i1} = t^o_{i1} - R_i$ gdy dana jest wielkość R_i, w pozostałych przypadkach $\tau_{i1} = t^o_{i1}$.

Natomiast wielkość

$$\tau_i = \sum_{j=1}^{n_i} \tau_{ij} \tag{18}$$

będziemy nazywali ogólnym czasem oczekiw a n i a detali.

Przyjmijmy dalej, że wielkość $\bar{t}_i = \sum_{j=1}^{n_i} t_{ij}$, oznacza czas wykonania wszystkich operacji i-tego detalu przy założeniu że $\tau_i = 0$.

Symbolem A_i będziemy oznaczali czas przebywania i-tego detalu w procesie obróbki. Wielkość A_i jest liczona następująco:

$$A_i = t^o_{in_i} - R_i \tag{19}$$

przy czym, gdy dla i-tego detalu nie została zadana wielkość R_i, wtedy $A_i = t^o_{in_i}$.

Symbolem $\bar{A}_i$ oznaczymy z kolei czas, w ciągu którego jest wykonywany i-ty detal. Wielkość $\bar{A}_i$ będzie liczona według następującego wzoru:

$$\bar{A}_i = t^1_{in_i} - t^o_{i1} \tag{20}$$

W przypadku, gdy dla i-tego detalu zadana jest wielkość R'_i, wtedy dla tego detalu może być określone p r z e k r o c z e n i e t e r m i n u w y k o n a n i a d e t a l u oraz p r z y s p i e s z e n i e t e r m i n u w y k o n a n i a d e t a l u. Pierwszą z tych wielkości oznaczymy symbolem $\bar{R}_i$ oraz będziemy liczyć według następującego wzoru:

$$\bar{R}_i = \max \left\{0, t^o_{in_i} - R'_i\right\}. \tag{21}$$

Drugą natomiast oznaczymy symbolem $\bar{\bar{R}}_i$ i wyliczamy następująco:

$$\bar{\bar{R}}_i = \max \left\{ 0, R_i' - t^1_{in_i} \right\} \tag{22}$$

Niech D_k oznacza zbiór tych operacji d_{ij}, które zgodnie z wyznaczonym harmonogramem będą wykonywane na k-tym stanowisku.

O b c i ą ż e n i e m k - t e g o s t a n o w i s k a od momentu a do momentu b nazywamy wielkość

$$\bar{D}_k(a,b) = \sum_{d_{ij} \in D_k} \bar{d}_{ij} \tag{23}$$

gdzie

$$\bar{d}_{ij} = \begin{cases} 0 & \text{gdy } t^1_{ij} \leqslant a \text{ lub } t^o_{ij} \geqslant b \\ t_{ij} & \text{gdy } t^o_{ij} \geqslant a \text{ i } t^1_{ij} \leqslant b \\ b - t^o_{ij} & \text{gdy } t^o_{ij} \geqslant a \text{ i } t^1_{ij} > b \\ t^1_{ij} - a & \text{gdy } t^o_{ij} < a \text{ i } t^1_{ij} \leqslant b \end{cases}$$

W przypadku gdy $R_i = 0$ oraz $a = 0$ natomiast $b = \max t^1_{in_i}$ wtedy

$$\bar{D}_k(a,b) = \sum_{d_{ij} \in D_k} t_{ij} \tag{24}$$

Wykorzystując wprowadzone wyżej określenia podamy następujące kryteria oceny:

a/ Kryterium minimalizacji czasu trwania obróbki wszystkich detali czyli minimalizacji wielkości F_1

$$F_1 = \max_i t^i_{in_i}$$

b/ Kryterium minimalizacji średnich czasów /momentów/ zakończenia obróbki detali tj. wielkość F_2

$$F_2 = \frac{1}{m_1} \sum_{i=1}^{m_1} t^i_{in_i}$$

c/ Kryterium minimalizacji maksymalnego czasu oczekiwania detalu tj. minimalizacji wielkości F_3

$$F_3 = \max_i (\tau_i)$$

d/ Kryterium minimalizacji czasu oczekiwania wszystkich detali

$$F_4 = \sum_{i=1}^{m_1} \tau_i$$

e/ Kryterium minimalizacji średniego czasu oczekiwania detalu

$$F_5 = \frac{1}{m_1} \sum_{i=1}^{m_1} \tau_i$$

f/ Kryterium maksymalizacji obciążenia stanowisk od momentu a do momentu b

$$F_6 = \sum_{k=1}^{K} \bar{D}_k (a,b)$$

g/ Kryterium maksymalizacji współczynnika wykorzystania stanowisk od momentu a do momentu b

$$F_7 = \frac{K(a-b)}{\sum_{k=1}^{K} \bar{D}_k(a,b)} , \quad F_7 \leqslant 1$$

h/ W przypadku gdy $R_i = 0$ oraz $c_k = 0$ można wprowadzić kryterium maksymalizacji współczynnika wykorzystania stanowisk od momentu $a = 0$ do momentu zakończenia wykonywania ostatniej operacji przez poszczególne stanowiska.

Wtedy

$$F_8^- = \frac{\sum_{i=1}^{m_1} t_i}{\sum_{k=1}^{K} \max\limits_{d_{ij} \in D_k} t_{ij}^1} , \quad F_8^- \leqslant 1$$

i/ Kryterium minimalizacji maksymalnego przekroczenia terminu wykonania detalu czyli minimalizacji wielkości F_9

$$F_9 = \max_i (\bar{R}_i)$$

j/ Kryterium minimalizacji średniego przekroczenia terminu wykonania detalu

$$F_{10} = \frac{\sum_{i=1}^{m_1} \bar{R}_i}{m_1}$$

k/ Kryterium minimalizacji przekroczenia terminu wykonania wszystkich detali

$$F_{11} = \sum_{i=1}^{m_1} R_i$$

l/ Kryterium maksymalizacji średniego przyspieszenia terminu wykonania detalu

$$F_{12} = \frac{1}{m_1} \sum_{i=1}^{m_1} \bar{\bar{R}}_i$$

m/ Kryterium maksymalizacji przyspieszenia terminu wykonania wszystkich detali

$$F_{13} = \sum_{i=1}^{m_1} \bar{\bar{R}}_i$$

4.3. Analiza metod rozwiązania

Istniejące metody rozwiązania problemu wyznaczania harmonogramu produkcji detali dzieli się w literaturze na dwie grupy [18]

- metody, za pomocą których znajdujemy rozwiązanie dokładne tzn. optymalne ze względu na określone kryterium oraz
- metody, za pomocą których otrzymujemy rozwiązanie przybliżone.

Metody pierwszej grupy stosuje się do rozwiązywania zadania Johnsona. Do metod tych zaliczyć należy oryginalną metodę Johnsona [87] , metodę pełnego przeliczania wariantów [167] , metody programowania matematycznego /liniowego, nieliniowego i dynamicznego / [95] , [177] , metodę analizy i eliminacji wariantów [177] , [191] , oraz metody rozpatrujące przypadki szczególne [28], [91].

Stosowanie pełnego przeliczania wariantów oznacza konieczność badania wszystkich możliwych wariantów rozwiązania. Dla każdego wariantu wyliczana jest wielkość funkcji-kryterium i jako rozwiązanie optymalne wybierany jest ten wariant, dla którego funkcja kryterium osiąga ekstremum. Daje to zawsze gwarancję, że otrzymamy rozwiązanie optymalne, jednak taki sposób postępowania w praktyce jest niewykonalny. Świadczy o tym chociażby prosty przykład. Przyjmijmy, że dysponujemy tylko jednym stanowiskiem, na którym powinno być wykonanych 10 różnych detali. Optymalny harmonogram dla tego stanowiska należy wybrać spośród wszystkich możliwych harmonogramów. Jak łatwo zauważyć takich harmonogramów będzie 10! czyli 3 628 000.

Możliwości zastosowania metod programowania liniowego i nieliniowego do wyznaczania harmonogramu produkcji detali były często rozpatrywane w literaturze. Jednak metody te prowadzą do otrzymania modeli o wielu zmiennych i ograniczeniach. Najbardziej "ekonomiczny" model podany przez Manne'a [95] wymaga wprowadzenia $mn\left(1+\frac{m-1}{2}\right)$ zmiennych, gdzie m -ilość detali, n-ilość stanowisk. Daje to możliwość rozwiązania tylko takich żadań, dla których wielkości m oraz n są liczbami małymi.

W pracy [164] pokazano ponadto, że te proste zadania, które udało się rozwiązać przy pomocy metod programowania liniowego lub nieliniowego mogą być rozwiązane na maszynie cyfrowej w krótszym czasie przy pomocy pełnego przeliczania.

Znaczne skrócenie obliczeń w porównaniu z pełnym przeliczeniem można uzyskać przy pomocy metody programowania dynamicznego. I tak np. jeżeli przy pomocy pełnego przeliczania w podanym wcześniej przykładzie trzeba sprawdzić m! wariantów, to stosując metodę programowania dynamicznego liczba wariantów zmniejszy się do 2^m. Jest to jednak jeszcze duża liczba.

Bardziej efektywną /biorąc pod uwagę czas obliczeń na maszynie/ jest metoda analizy i eliminacji wariantów. Stanowi ona pewnego rodzaju modyfikację metody programowania dynamicznego.
Zmniejszenie obliczeń osiąga się przez to, że początkowo analizowane są wariancje o niewielkiej liczbie elementów z m-elementowego zbioru detali. Przy pomocy wprowadzanych specjalnych ocen eliminowania jest duża ilość tzw. n i e p e r s p e k t y w i c z n y c h   w a r i a c j i. Eliminując każdą taką wariację nie rozpatrujemy w dalszych krokach tych wariacji, których początkowe elementy stanowiłyby wariacje nieperspektywiczne.
Metoda analizy i eliminacji wariantów pozwala jednak rozwiązywać takie zadania, dla których wielkości m oraz n są liczbami małymi. Zwiększenie tych wielkości powoduje bardzo szybki wzrost czasu obliczeń na maszynie.

Efektywne metody można uzyskać tylko dla przypadków szczególnych. Przykład takiej metody można znaleźć w pracy [91] .
W sformułowanym zadaniu wprowadzono mianowicie dodatkowe ograniczenie / w odniesieniu do zadania Johnsona/ polegające na tym, że detale mają być obrabiane w sposób ciągły. Inaczej mówiąc, chodzi tu o to, że dla każdego $i = 1,2,\ldots,m$ oraz $j=1,2,\ldots,n-1$ musi zachodzić równość

$$t^{o}_{i,j+1} = t^{o}_{ij} + t_{ij} \quad ,$$

gdzie t^{o}_{ij} oznacza jak zwykle moment rozpoczęcia obróbki i-tego detalu na j-tym stanowisku.

Na podstawie powyższych rozważań można powiedzieć, że metody należące do pierwszej grupy mają zasadniczą wadę. Przy ich pomocy mogą być rozwiązywane niewielkie zadania Johnsona, które jak wiadomo najczęściej nie znajdują odzwierciedlenia w praktyce. Mimo to nie należy negować znaczenia tych metod, ponieważ stanowią one swego rodzaju poligon doświadczalny w poszukiwaniach metod rozwiązania zadań spotykanych w realnych warunkach.

Jak dotychczas nawet dla zadania Johnsona brak jest ogólnej metody pozwalającej przy stosunkowo niewielkiej liczbie obliczeń, uzyskać rozwiązanie optymalne. Stąd pojawiło się wiele prób wyznaczania harmonogramu produkcji detali przy pomocy metod przybliżonych. Główna idea tych metod, to skrócenie ilości przeliczeń.
Szereg metod przybliżonych, przy pomocy których można rozwią-

zać zadanie Johnsona zostało opisanych w pracach [91] , [139]. Metody rozwiązania zadania Glifferta-Thompsona są podane w pracy [95] . W obu przypadkach są to metody bazujące na znanej metodzie Monte-Carlo bądź są to metody heurystyczne.

Jeżeli chodzi o metody rozwiązywania zadań spotykanych w realnych warunkach / zadanie podstawowe i jego modyfikacje/, to tutaj należy wymienić przede wszystkim metody heurystyczne [177] , [183] .

Jak już zaznaczono, otrzymując rozwiązanie zadania przy pomocy heurystycznej metody nie możemy być pewni, czy jest to rozwiązanie optymalne i często nie możemy ocenić na ile różni się ono od optymalnego. Jednak na korzyść tych metod przemawia prostota obliczeń i fakt że są to w chwili obecnej metody, za pomocą których można wyznaczyć harmonogram produkcji detali przy pomocy komputera dla zagadnień spotykanych w praktyce. Do metod tych należy jak wiadomo proponowana w tej pracy metoda priorytetów. W dalszym ciągu podamy na czym polega rozwiązanie zadania podstawowego przy pomocy metody priorytetów.

4.4. Sposób rozwiązania przy pomocy metody priorytetów

4.4.1. Wstęp

Zanim zostanie zaprezentowany algorytm rozwiązania tzw. zadania podstawowego /patrz punkt 4.1.2./, wyjaśnimy jak będą interpretowane w omawianym zadaniu ważniejsze pojęcia dotyczące metody priorytetów.

W algorytmie zastosujemy postępowanie II. Oznacza to, jak wiadomo, że w każdym kroku do harmonogramu będą wprowadzane jedna lub więcej czynności / w tym przypadku operacji/ i każdemu krokowi odpowiada moment czasu.

Moment czasu odpowiadający r-temu krokowi / $r = 1,2,3,\ldots$/ będziemy oznaczali jak poprzednio /por. § 3.2./ symbolem h_r.

O operacji d_{ij} / $i = 1,2,\ldots,m_1$, $j = 1,2,\ldots,n_i$ / powiemy, że z o s t a ł a w p r o w a d z o n a d o h a r m o n o g r a m u w r - t y m k r o k u, jeżeli dla tej operacji został określony moment rozpoczęcia równy h_r oraz stanowisko robocze.

O stanowisku tym powiemy wówczas, że z o s t a ł a m u p r z y d z i e l o n a o p e r a c j a d o w y k o n a n i a. Dla takiego stanowiska można wyliczyć moment, od którego począwszy stanowisko to będzie wolne. Oznaczymy go symbolem ω_k. Wielkość ω_k równa się momentowi zakończenia operacji przydzielonej do wykonania.

Symbolem Z(r) oznaczymy zbiór tych operacji d_{ij} / $i = 1,2,\ldots,m_1$, $j = 1,2,\ldots,n_i$ /, które zostały wprowadzone do harmonogramu w dowolnym kroku mniejszym lub równym r / $r = 1, 2, 3,\ldots$ / oraz moment zakończenia tych operacji t^1_{ij} jest większy od h_r.

Dla $r = 2,3,\ldots$ wielkość h_r będzie liczona następująco:

$$h_r = \min_{d_{ij} \in Z(r-1)} t^1_{ij} \qquad (25)$$

Gdy $r = 1$ wtedy $h_1 = 0$. Zauważmy jednak, że gdyby dla k-tego stanowiska została zadana wielkość c_k wtedy $h_1 = \min_k c_k$.

Będziemy mówili, że operacja d_{ij} / $i = 1,2,\ldots,m_1$, $j = 2,3,\ldots,n_i$ / jest **gotowa do wykonania na stanowiskach roboczych l-tej grupy w r-tym kroku** / $r = 2,3,\ldots,$ /, jeżeli operacja $d_{i,j-1}$ została wprowadzona w dowolnym kroku z przedziału $[1,r-1]$ a moment zakończenia tej operacji $t^1_{i,j-1}$ jest mniejszy lub równy wielkości h_r oraz $l_{ij} = l$. Operacja d_{i1} / $i = 1, 2,\ldots,m_1$ / jest gotowa do wykonania w pierwszym kroku.

4.4.2. Algorytm rozwiązania

Przebieg konstruowania rozwiązania można przedstawić w następujący sposób.

Pierwszy krok / $r = 1$ /

1. Dla każdej l-tej grupy stanowisk określić należy zbiór Q_l składający się z tych operacji d_{ij} / $i = 1,2,\ldots,m_1$ / $j = 1,2,\ldots,n_i$ / dla których $l_{ij} = l$.

2. Dla każdej l-tej grupy należy określić zbiór operacji gotowych do wykonania w pierwszym kroku. Oznaczymy go symbolem $\bar{Z}_l(1)$ - w ogólnym przypadku $\bar{Z}_l(r)$.
   Do zbioru $\bar{Z}_l(1)$ będą należały operacje oznaczone symbolem d_{i1} znajdujące się w zbiorze Q_l. Po utworzeniu zbioru

$\bar{Z}_l(1)$ operacje należące do tego zbioru "wykreślamy" ze zbioru Q_1.

3. Dla każdej l-tej grupy dla której $\bar{Z}_l(1) \neq \emptyset$ należy zbadać czy jest spełniona nierówność $\|\bar{Z}_l(1)\| \leqslant q_l$, gdzie $\|\bar{Z}_l(1)\|$ - moc zbioru, zaś q_l - liczba stanowisk w l-tej grupie. Jeżeli nierówność jest spełniona przechodzimy do punktu 5, w przeciwnym wypadku do punktu następnego.

4. Nie spełnienie nierówności świadczy o tym, że występuje konflikt. Ze zbioru $\bar{Z}_l(1)$ należy wtedy wybrać pewien podzbiór o liczebności q_l stosując w tym celu /zarówno w kroku pierwszym jak i w każdym następnym / zadany sposób likwidacji konfliktu.

Propozycje sposobów likwidacji konfliktów zostaną podane na str. 102.

5. Każdej operacji należącej do podzbioru wybranego w punkcie 4, bądź gdy konflikt nie występuje każdej operacji ze zbioru $\bar{Z}_l(1)$, przyporządkowujemy moment rozpoczęcia równy h_1 oraz dowolne stanowisko robocze z l-tej grupy, przy czym dla stanowiska obliczamy moment ω_k. Operacje te "wykreślamy" następnie ze zbioru $\bar{Z}_l(1)$. Będą one tworzyły zbiór $Z_l(1)$. Wszystkie operacje wprowadzone do harmonogramu w pierwszym kroku będą tworzyły z kolei zbiór $Z(1)$ - ogólnie $Z(r)$, gdzie $Z(1) = Z_1(1) \cup Z_2(1) \cup \ldots$ $\ldots \cup Z_{m_2}(1)$. Po wykonaniu działań opisanych w punktach 1 - 5 należy przejść do kroku drugiego.

O g ó l n y k r o k / r = 2,3,... /

1. Gdy wszystkie zbiory Q_l są puste, zadanie zostało rozwiązane, w przeciwnym wypadku należy wyliczyć wielkość h_r wg wzoru (25).
2. Dla każdej l-tej grupy należy określić zbiór $\bar{Z}_l(r)$. Do zbioru tego będą należały, oprócz elementów zbioru $\bar{Z}_l(r-1)$, pewne elementy zbioru Q_l. Elementy takie są określane następująco. W zbiorach $Z_l(r-1)$ należy zidentyfikować elementy / inaczej operacje / dla których moment zakończenia równa się wielkości h_r. Przyjmijmy, że taką operacją w pewnym zbiorze $Z_l(r-1)$ jest operacja d_{ij} / $i = 1,2,\ldots,m_l$, $j = 1,2,\ldots,n_i-1$ /, wtedy operacja $d_{i,j+1}$ ze zbioru Q_l będzie należała do zbioru $\bar{Z}_l(r)$ przy czym wskaźnik l przyjmuje jednakową wartość dla zbiorów Q_l oraz $\bar{Z}_l(r)$.
Operację d_{ij} należy wówczas "wykreślić" ze zbioru $Z_l(r-1)$ oraz odpowiednio operację $d_{i,j+1}$ ze zbioru Q_l.
3. W każdej l-tej grupie, dla której $\bar{Z}_l(r) \neq \emptyset$ należy zidentyfikować wolne stanowiska robocze. Będą to takie stanowiska, dla których wielkość ω_k jest równa lub mniejsza od wielkości h_r. Następnie należy obliczyć ilość takich stanowisk, czyli wielkość $q_l(r)$.
4. Jeżeli w l-tej grupie stanowisk $q_l(r) > 0$ należy zbadać czy jest spełniona nierówność $\|\bar{Z}_l(r)\| \leq q_l(r)$. Gdy nierówność jest spełniona wtedy należy przejść do punktu 5. W przeciwnym wypadku następuje konflikt. Konflikt będzie polegał na wybraniu w zbiorze $\bar{Z}_l(r)$ pewnego podzbioru o liczebności $q_l(r)$.

Gdyby we wszystkich l-tych grupach, dla których $\bar{Z}_l(r) \neq \emptyset$ wielkości $q_l = 0$, wtedy zwiększamy krok o jeden i przechodzimy do punktu pierwszego.

5. Każdej operacji należącej do podzbioru wybranego w punkcie 4, bądź w przypadku gdy konflikt nie występuje, każdej operacji ze zbioru $\bar{Z}_l(r)$, należy przyporządkować moment rozpoczęcia równy h_r oraz dowolne stanowisko wśród wolnych stanowisk l-tej grupy. Dla stanowisk należy następnie obliczyć wielkość ω_k. Operacje wprowadzone do harmonogramu "wykreślamy" ze zbioru $\bar{Z}_l(r)$ a następnie dodajemy do zbioru $Z_l(r-1)$. W wyniku otrzymamy zbiór $Z_l(r)$, przy czym $Z(r) = Z_1(r) \cup Z_2(r) \cup \ldots \cup Z_{m_2}(r)$.

Po wykonaniu powyższych działań należy zwiększyć krok o jeden i przejść do punktu pierwszego.

4.4.3. Funkcje priorytetu oraz sposoby likwidacji konfliktów

Zanim podamy wzory funkcji priorytetu, określimy z jakich składowych składa się wektor X zadawany dla każdej operacji ze zbioru $\bar{Z}_l(r)$

$$X = (x_1, x_2, x_3, x_4, x_5, x_6, x_7)$$

gdzie:

x_1 - czas trwania operacji

x_2 - numer operacji w marszrucie technologicznej

x_3 - czas trwania operacji bezpośrednio następującej w marszrucie technologicznej po rozpatrywanej operacji

x_4 - ilość wszystkich operacji w marszrucie technologicznej

x_5 - czas trwania wszystkich operacji w marszrucie technologicznej czyli wielkość - $\sum_{j=1}^{n_i} t_{ij}$

x_6 - koszt wykonania detalu na którym wykonywana jest rozpatrywana operacja

x_7 - termin wykonania detalu na którym wykonywana jest rozpatrywana operacja.

Oto przykłady funkcji priorytetu, które mogą być stosowane w różnych sposobach likwidacji konfliktów.

1^o $w_1 = G_1(X) = x_1$

2^o $w_2 = G_2(X) = x_4$

3^o $w_3 = G_3(X) = x_4 - x_2$

4^o $w_4 = G_4(X) = x_1 + x_3$

5^o $w_5 = G_5(X) = x_5$

6^o $w_6 = G_6(X) = x_6$ przy czym $w_6 > 0$

7^o $w_7 = G_7(X) = x_7$ przy czym $w_7 > 0$.

Rozpatrzmy kilka sposobów likwidacji konfliktów.

Sposób A

Likwidacja konfliktu odbywa się w sposób następujący:

- wśród zbioru operacji $\bar{Z}_r(1)$ wybieramy kolejno operacje według minimalnych wartości funckji G_1,
- w przypadku niejednoznaczności wyboru, dla podzbioru operacji, których wartości funkcji G_1 są jednakowe, stosujemy

funkcję G_5 i wybieramy według minimalnych wartości tej funkcji,

- przy dalszej niejednoznaczności wybieramy losowo z jednakowym prawdopodobieństwem.

Sposób B

Sposób B tym się różni od sposobu A, że operacje wybieramy według maksymalnych wartości zarówno funkcji G_1 jak i funkcji G_5.

Sposób C

Ze zbioru operacji $\bar{Z}_1(r)$ wybieramy operacje według minimalnych wartości funkcji G_4. Przy niejednoznaczności wyboru wybieramy według minimalnych wartości funkcji G_3 a przy dalszej niejednoznaczności losowo z jednakowym prawdopodobieństwem.

Sposób D

Operacje wybieramy według maksymalnych wartości funkcji G_1. W przypadku niejednoznaczności wybieramy według maksymalnych wartości funckji G_2 a następnie według maksymalnych wartości funkcji G_5 oraz losowo z jednakowym prawdopodobieństwem.

Sposób E

Operacje wybieramy według minimalnych wartości funkcji G_1. Następnie przy niejednoznaczności wybieramy według minimalnych wartości funkcji G_6 i dalej według minimalnych wartości funkcji G_7 oraz losowo z jednakowym prawdopodobieństwem.

4.4.4. Przykład

Działanie algorytmu prześledzimy na przykładzie zaczerpniętym z pracy [164].
Pewien odcinek produkcyjny składa się z pięciu stanowisk roboczych K = 5. Stanowiska 1 i 2 należą do pierwszej grupy stanowisk /l = 1/, stanowisko 3 do drugiej grupy stanowisk /l = 2/, stanowisko 4 do trzeciej grupy stanowisk /l = 3/ oraz stanowisko 5 do czwartej grupy stanowisk /l = 4/.
Na odcinku tym trzeba wykonać cztery detale. Marszruty technologiczne detali oraz czas trwania poszczególnych operacji podane są w tablicy 1a.

Tablica 1a

Nr detalu (i)	Nr operacji (j)	Nr grupy stanowisk (l)	Czas trwania operacji (t_{ij})
1	1	1	3
	2	2	2
	3	3	2
	4	4	1
2	1	1	4
	2	3	1
	3	2	2
3	1	2	3
	2	1	3
	3	4	2
	4	3	2
4	1	1	3
	2	2	1
	3	1	2
	4	3	1

Należy określić harmonogram wykonywania operacji na tym odcinku produkcyjnym. W tym celu wygenerujemy dwa rozwiązania dopuszczalne stosując kolejno dwa pierwsze sposoby likwidacji konfliktów / sposób A oraz B /. Tok obliczeń zaprezentujemy na przykładzie generowania rozwiązania pierwszego.

Pierwszy krok / $h_1 = 0$ /.

1. Określamy zbiory Q_1, Q_2, Q_3, Q_4 / por. punkt 1 algorytmu/.

Dla oznaczenia elementów tych zbiorów zastosujemy przyjętą wcześniej symbolikę, tj. symbol d_{ij} oznaczać będzie j-tą operację wykonywaną na i-tym detalu.

$$Q_1 = \{d_{11}, d_{21}, d_{32}, d_{41}, d_{43}\}$$

$$Q_2 = \{d_{12}, d_{23}, d_{31}, d_{42}\}$$

$$Q_3 = \{d_{13}, d_{23}, d_{34}, d_{44}\}$$

$$Q_4 = \{d_{14}, d_{33}\}$$

2. Określamy zbiory $\bar{Z}_1(1)$, $\bar{Z}_2(1)$, $\bar{Z}_3(1)$, $\bar{Z}_4(1)$ - patrz punkt 2 algorytmu

$$\bar{Z}_1(1) = \{d_{11}, d_{21}, d_{41}\}$$

$$\bar{Z}_2(1)' = \{d_{31}\}$$

$$\bar{Z}_3(1) = \emptyset$$

$$\bar{Z}_4(1) = \emptyset$$

3. Konflikt występuje w pierwszej grupie stanowisk, ponieważ $\|\bar{Z}_1(1)\| \geqslant 2$ /por. punkt 3 algorytmu /. Stosując pierwszy sposób likwidacji konfliktu /sposób / ze zbioru $\bar{Z}_1(1)$ wybierzemy operacje d_{11} oraz d_{41}.

4. Operacjom d_{11}, d_{31}, d_{41} przyporządkowujemy moment rozpoczęcia równy h_1 oraz stanowiska robocze / patrz punkt 5 algorytmu / Dla operacji d_{31} będzie to oczywiście stanowisko trzecie natomiast dla dwóch pozostałych wybieramy jak wiadomo w sposób dowolny wśród pierwszej grupy stanowisk. Operacji d_{11} przyporządkowujemy stanowisko pierwsze, operacji d_{41} stanowisko drugie . Powyższe dane umieszczamy w tablicy 2. / str.108 /.

5. Wyliczamy wielkości $\omega_1, \omega_2, \omega_3, \omega_4, \omega_5$ / por.def. na str. 97 / oraz określamy zbiory $Z_1(1)$; $Z_2(1)$, $Z_3(1)$, $Z_4(1)$ /por. punkt 5 algorytmu/.

$$\omega_1 = 3 , \quad \omega_2 = 3 , \quad \omega_3 = 3 , \quad \omega_4 = 0 , \quad \omega_5 = 0$$

$$Z_1(1) = \{d_{11}, d_{41}\}$$

$$Z_2(1) = \{d_{31}\}$$

$$Z_3(1) = \emptyset$$

$$Z_4(1) = \emptyset$$

z kolei $Z(1) = \{d_{11}, d_{31}, d_{41}\}$ /por. punkt 5 algorytmu /

Na tym działania w pierwszym kroku zostały zakończone i należy przejść do kroku drugiego.

Krok drugi / r = 2 /

1. Zbiory Q_1, Q_2, Q_3, Q_4 określone w poprzednim kroku nie są puste wobec tego obliczamy $h_2 = 3$ / patrz wzór (25) oraz punkt 1 algorytmu /

2. Określamy zbiory /por. punkt 2 algorytmu /:

$Q_1 = \{d_{43}\}$

$Q_2 = \{d_{23}\}$

$Q_3 = \{d_{13}, d_{22}, d_{34}, d_{44}\}$

$Q_4 = \{d_{14}, d_{33}\}$

$\bar{Z}_1\ (2) = \{d_{21}, d_{32}\}$

$\bar{Z}_2\ (2) = \{d_{12}, d_{42}\}$

$\bar{Z}_3\ (2) = \emptyset$

$\bar{Z}_4\ (2) = \emptyset$

3. Wyliczamy wielkości $q_1\ (2)$ oraz $q_2\ (2)$ /por. punkt 3 algorytmu /

$q_1(2) = 2$

$q_2(2) = 1$

4. Konflikt występuje w drugiej grupie stanowisk, ponieważ $\| Z_2\ (2) \| > 1$ /por. punkt 4 algorytmu / Według sposobu A wybrana będzie operacja d_{42}.

5. Operacji d_{21} przyporządkowujemy stanowisko pierwsze, operacji d_{32} stanowisko drugie natomiast operacji d_{42} stanowisko trzecie. Moment rozpoczęcia trzech operacji równa się h_2 / patrz punkt 5 algorytmu /. Operacje te zostały w ten sposób wprowadzone do harmonogramu.

6. Zgodnie z punktem 5 algorytmu wyliczamy

$\omega_1 = 7$, $\omega_2 = 6$, $\omega_3 = 4$; $\omega_4 = 0$, $\omega_5 = 0$,

określamy

$Z_1(2) = \{d_{21}, d_{32}\}$

$Z_2(2) = \{d_{42}\}$

$Z_3(2) = \emptyset$

$Z_4(2) = \emptyset$

$Z(2) = \{d_{21}, d_{32}, d_{42}\}$,

oraz przechodzimy do kroku następnego / $r = 3$ /.

Działania w następnych krokach przebiegają analogicznie jak w kroku drugim.

W siedmiu krokach otrzymujemy harmonogram, który przedstawia tablica 2 / str. 108 /.

Tablica 3 / str. 109/ zawiera harmonogram, przy wyznaczaniu którego konflikty były likwidowane według sposobu B.

Tablica 2

Nr stanowiska	Nr detalu	Nr operacji	Moment rozpoczęcia operacji	Moment zakończenia operacji
1	1	1	0	3
	2	1	3	7
2	4	1	0	3
	3	2	3	6
	4	3	6	8
3	3	1	0	3
	4	2	3	4
	1	2	4	6
	2	3	9	11
4	1	3	6	8
	2	2	8	9
	4	4	9	10
	3	4	10	12
5	3	3	6	8
	1	4	8	9

Tablica 3

Nr stanowiska	Nr detalu	Nr operacji	Moment rozpoczęcia operacji	Moment zakończenia operacji
1	1	1	0	3
	3	2	3	6
2	2	1	0	4
	4	1	5	8
	4	3	9	11
3	3	1	0	3
	1	2	3	5
	2	3	5	8
	4	2	8	9
4	2	2	4	5
	1	3	5	7
	3	4	8	10
	4	4	11	12
5	3	3	6	8
	1	4	8	9

W obu tablicach dla każdego stanowiska podane są operacje, które będą wykonywane na tym stanowisku oraz momenty rozpoczęcia i zakończenia operacji.

O tak przestawionym harmonogramie w języku profesionalnym mówi się, że jest h a r m o n o g r a m e m d l a s t a n o w i s k.

Podaje się też h a r m o o g r a m y d l a d e t a l i. Wtedy dla poszczególnego detalu podawane są momenty rozpoczęcia i zakończenia każdej operacji wykonywanej na tym detalu oraz stanowisko, na którym ta operacja jest wykonywana. Zauważmy, że na podstawie harmonogramu dla stanowisk łatwo sporządzić harmonogram dla detali. Biorąc pod uwagę kryterium minimalizacji średnich momentów zakończenia obróbki detali[1/] /por. § 4.2. / lepszym harmonogramem będzie harmonogram zamieszczony w tablicy 3.

1/ Kryterium to bywa często stosowane w literaturze do oceny rozwiązań żadań o małych rozmiarach.

5. WYZNACZANIE HARMONOGRAMU DLA ZAGADNIEŃ Z SZEREGOWO-RÓWNOLEGŁĄ KOLEJNOŚCIĄ CZYNNOŚCI

5.1. W p r o w a d z e n i e

Wśród trzech grup zagadnień będących przedmiotem rozważań teorii przedsięwzięć czasowych, zagadnienia omawiane w tym rozdziale są najczęściej poruszane w literaturze.
Dla rozwiązywania zagadnień z szeregowo-równoległą kolejnością czynności wypracowanych zastało szereg metod przy czym niektóre z nich są stosowane do rozwiązywania zagadnień spotykanych w praktyce. Jak wcześniej wspomniano ogół tych metod nazywa się w teorii badań operacyjnych metodami sieciowymi.
Za początek rozwoju metod sieciowych można uważać opracowanie w roku 1957 w Stanach Zjednoczonych metody pod nazwą Critical Path Method /CPM/.
W podejściu do rozwiązywania zagadnień z szeregowo-równoległą kolejnością czynności / będziemy je też w skrócie nazywać modelami sieciowymi / wyróżnia się w zasadzie trzy kierunki badań.

W pierwszym kierunku badań, historycznie najwcześniejszym, przy budowie harmonogramu bierze się pod uwagę tylko czasy trwania czynności. Wyznaczenie harmonogramu polega na określeniu najwcześniejszego terminu rozpoczęcia i zakończenia poszczególnych czynności, obliczeniu rezerw czasowych oraz określeniu tzw. ś c i e ż k i k r y t y c z n e j.
Rozpatrywane przez ten kierunek badań modele sieciowe nazy-

wane są w literaturze m o d e l a m i m i n i m a l n o-c z a s o w y m i [82]
Wydaje się, że w aspekcie teoretycznym ten kierunek stanowi już całkowicie zamknięty rozdział badań.

Drugi kierunek badań rozpatruje modele sieciowe z punktu widzenia minimalizacji ponoszonych nakładów. Zakłada się tutaj, że czas trwania czynności zależy od nakładów jakie przeznacza się na jej wykonanie.
W zasadzie rozpatruje się tutaj dwa rodzaje zagadnień

- wyznaczenie najkrótszego harmonogramu przy założonych z góry kosztach realizacji poszczególnych czynności,
- wyznaczenie minimalnych kosztów realizacji czynności przy zadanym terminie zakończenia wszystkich czynności.

Zagadnienia te rozwiązuje się przede wszystkim przy pomocy metod programowania liniowego oraz całkowito-liczbowego [10], [195]. Pewne nowe propozycje oparte na teorii programowania matematycznego zawiera ostatnio wydana obszerna monografia O. Giedymina [55]
W przypadku dużych i skomplikowanych sieci rozwiązanie zadania przy pomocy podanych wyżej metod jest bardzo skomplikowane. Główną jednak trudność stanowi wyznaczenie funkcji " czas - koszt " tj. funkcji wyrażającej zależność czasu od kosztu jaki ponoszony jest przy wykonywaniu czynności. Bardzo często określenie tej zależności przed przystąpieniem do realizacji czynności jest niemożliwe.
W obu kierunkach zakłada się ponadto, że nie istnieją ograniczenia dostępności zasobów koniecznych do realizacji

wyliczonych harmonogramów. Taka sytuacja w realnej rzeczywistości występuje bardzo rzadko.
Ilość zasobów / maszyn i urządzeń, siły roboczej, powierzchni itp. / dysponowanych przez komórkę realizującą dany kompleks czynności jest wielkością skończoną i najczęściej w okresie wykonywania czynności czyli w okresie trwania danego przedsiębiorstwa nie ulega zmianie.
W związku z tym wyodrębnił się nowy kierunek badań, przedmiotem których są zagadnienia racjonalnego rozdziału ograniczonych zasobów.
Do chwili obecnej opracowano szereg metod przydatnych do optymalizacji rozdziału zasobów m.in. z uwzględnieniem minimalizacji czasu trwania przedsięwzięcia, równomiernego obciążenia poszczególnych jednostek wykonawczych przy zadanym terminie zakończenia przedsięwzięcia itp.

Pierwsza grupa metod bazuje na programowaniu matematycznym. Zastosowaniu tych metod do rozdziału zasobów w typowych modelach sieciowych poświęcona jest w całości praca [82] .
Warunkiem wstępnym do sformułowania zadania ekstremalnego dla każdego modelu sieciowego jest pogrupowanie czynności w zbiory czynności, które można wykonywać równocześnie /tzw. fronty/ i uporządkowania tych zbiorów leksykograficznie. Ta oryginalna propozycja autora pracy [82] ułatwia zdefiniowanie zadania ekstremalnego a także formułowanie algorytmów obliczeniowych w porównaniu do sposobów przedstawionych np. w pracy [195].
Jak podkreślono w zakończeniu pracy [82] zastosowanie w modelach sieciowych do rozdziału dostępnych zasobów metod progra-

mowania matematycznego znajduje się jeszcze w stadium prób. Zadania,które udaje się rozwiązać przy pomocy tych metod są niewielkich rozmiarów i opierają się często na tzw. mocnych założeniach, które ograniczają możliwość stosowania otrzymanych rozwiązań w praktyce.1/

W literaturze można spotkać próby zastosowania metod kombinatorycznych do omawianych tutaj problemów. I tak w pracy [164] opisano zastosowanie metody analizy i eliminacji wariantów stanowiącej w istocie mutację metody podziału i ograniczeń. Dla znalezienia optymalnego rozwiązania niewielkiego zadania / 11 czynności i dwa rodzaje zasobów / trzeba było wygenerować i ocenić aż 83 wariantów rozwiązań.

Wobec braku możliwości otrzymania przy pomocy metod ścisłych rozwiązania optymalnego zadań realnych zaczęto rozwijać metody przybliżone. Rozwiązania uzyskiwane przy pomocy tych metod mają charakter suboptymalny tzn. są to rozwiązania dopuszczalne ale jedne z wielu i niekoniecznie optymalne. W pracy [27] do rozwiązania omawianych zagadnień zaproponowano zastosowanie metody Monte-Carlo przedstawiając szczegółowo algorytm rozwiązania. W literaturze brak jednoznacznych stwierdzeń na temat efektywności algorytmów wykorzystujących metodę Monte-Carlo do rozwiązywania dużych modeli sieciowych. Autorzy cytowanej wyżej pracy stwierdzają jedynie, że liczba kroków

---

1/ Proponowane w pracy [82] zadania ekstremalne na dostępnych w Polsce komputerach można rozwiązać jeżeli liczba frontów sieci nie przekracza 160 .

jaką należy wykonać poszukując rozwiązania przy pomocy metody Monte-Carlo jest mniejsza od liczby kroków przy stosowaniu pełnego przeliczania wariantów.
Wydaje się, że wykorzystanie metod programowania matematycznego, metod kombinatorycznych oraz metody Monte-Carlo do rozwiązywania sieciowych modeli rozdziału zasobów istniejących, w realnych warunkach pozostaje jeszcze w sferze teoretycznych propozycji.
W chwili obecnej do rozwiązywania realnych zadań mogą być zastosowane jedynie metody heurystyczne [50] , [81] , [138] , [164] , [195] .
Rozwiązywanie prostszych i mniejszych zagadnień przy pomocy metod heurystycznych polega na sporządzeniu tzw. wykresów obciążenia zasobów [81] i rachunków bilansujących zasoby z potrzebami. Jednak większość problemów rozwiązywanych przy pomocy metod heurystycznych wymaga zastosowania komputera.
Jak już podkreślono metoda priorytetów stanowi pewną syntezę metod heurystycznych. Może być ona zastosowana do większości zagadnień dotyczących racjonalnego rozdziału ograniczonych zasobów.

Ostatnnio można mówić o czwartym kierunku badań, który wynika ze wzrostu wymagań ze strony konkretnych użytkowników w stosunku do otrzymywanych harmonogramów. Chodzi tu przede wszystkim o budowę takich harmonogramów, w których uwzględniono by zarówno parametr kosztów jak i ograniczoność zasobów. W pracy [27] sformułowano takie zadanie i zaprezentowano możliwość jego rozwiązania przy pomocy metody Monte-Carlo.

Zadania te pozostają w sferze teoretycznych rozważań. Wyniki jakie uzyskuje się obecnie są dalekie od praktycznego ich zastosowania.

Przedmiotem rozważań niniejszego rozdziału będzie zastosowanie metody priorytetów do rozwiązywania modeli sieciowych rozdziału zasobów występujących w operatywnym planowaniu prac wydziału montażowego. Poniżej zostanie podane sformułowanie takiego problemu. Trzeba podkreślić, że modelami sieciowymi zainteresowano się po raz pierwszy niejako na zamówienie przedsiębiorstw przemysłu maszynowego, w których operatywnym planowaniu produkcji występuje szereg takich zadań /por. § 4.1 /. Są to jak się wydaje, jedne z najtrudniejszych zadań w operatywnym planowaniu produkcji, przy tym rzadko rozpatrywane w literaturze i dalekie od zadowalającego rozwiązania w praktyce.

5.2. S f o r m u ł o w a n i e p r o b l e m u

Niech będzie trzeba wykonać m_1 wyrobów. Poszczególne wyroby ponumerujemy liczbami naturalnymi od 1 do m_1 i będziemy oznaczać symbolem b_i. Odcinek produkcyjny na którym wyroby będą wykonywane składa się z m_2 grup stanowisk. Grupy stanowisk ponumerujemy liczbami naturalnymi $1,\ldots,k,\ldots,m_2$. Dla każdej k-tej grupy stanowisk zadany jest d y s p o n o w a n y f u n d u s z m o c y p r o d u k c y j n e j. Oznaczmy go symbolem c_k / $k = 1,2,\ldots,m_2$ /. Aby wykonać wyrób b_i / $i = 1,2,\ldots,m_1$ / należy zrealizować

n_i operacji. Poszczególne operacje będziemy oznaczać symbolem d_{ij} / $i = 1,2,\dots,m_1$, $j = 1,2,\dots,n_i$ / . Operacje wykonywane na i-tym wyrobie muszą być realizowane w określonej kolejności. Kolejność ta dla każdego i-tego wyrobu zadawana jest przy pomocy macierzy kwadratowej [1/] A_i.

$$A_i = \left[a^i_{jw}\right] \qquad \begin{array}{l} i = 1,2,\dots,m_1 \\ j = 1,2,\dots,n_i \\ w = 1,2,\dots,n_i \end{array} \tag{26}$$

gdzie:

$$a^i_{jw} = \begin{cases} 1 & \text{gdy operacja } d_{ij} \text{ bezpośrednio poprzedza wykonanie operacji } d_{iw} \\ 0 & \text{w przeciwnym przypadku .} \end{cases}$$

Dla każdej operacji d_{ij} zadane są:

- czas trwania , który oznaczymy symbolem t_{ij},
- grupa stanowisk na której wykonywana jest operacja, będziemy ją oznaczać symbolem k_{ij} / $k = 1,2,\dots,m_2$ /,
- fundusz mocy produkcyjnej k-tej grupy stanowisk wymagany do wykonania operacji w każdym momencie jej realizacji [2/], oznaczymy go symbolem f^k_{ij} .

Przy realizacji opisanego kompleksu operacji powinny być przestrzegane następujące ograniczenia:

1/ Macierz kwadratowa A_i zwana jest w literaturze przedmiotu [8], [82] macierzą łuków grafu. W opisywanym przypadku będzie to zdefiniowany wcześniej graf zwany siecią.

2/ W sformułowaniach bardziej ogólnych wielkość ta nazywana bywa intensywnością zużycia zasobu [13] , [195] .

1° Wielkość f_{ij}^k będzie stała w każdym momencie realizacji operacji d_{ij}

2° Wielkość c_k będzie stała w każdym momencie t

3° Żadana operacja z chwilą rozpoczęcia nie może być przerwana. Tak więc pracochłonność wykonania operacji można obliczyć korzystając ze wzoru:

$$t_{ij} = f_{ij}^k \tag{27}$$

4° W każdym momencie czasu zapotrzebowanie na moc produkcyjną w poszczególnych grupach stanowisk nie może przekraczać dysponowanego funduszu c_k czyli

$$\sum_{i,j} f_{ij}^k \leqslant c_k \qquad \text{dla } k = 1,2,\ldots,m_2 \tag{28}$$

Problem polega na tym aby określić dla każdej operacji d_{ij} / $i = 1,2,\ldots,m_1$, $j = 1,2,\ldots,n_i$ / moment jej rozpoczęcia t_{ij}^o przestrzegając przy tym wszystkich sformułowanych wyżej warunków i ograniczeń. W ten sposób otrzyma się dopuszczalny harmonogram przebiegu przedsięwzięcia. Oczywiście należy szukać harmonogramu optymalnego tzn. takiego, dla którego zadana funkcja-kryterium osiąga wartość ekstremalną. Jednak znalezienie takiego harmonogramu dla zdecydowanej większości zadań występujących w praktyce jest jak wiadomo obecnie niemożliwe.

Formułując problem starano się zwrócić uwagę na najbardziej typowe ograniczenia występujące w zadaniach praktycznych. W literaturze przedmiotu [27] , [164], [195] występuje wielka różnorodność sformułowań opisanego wyżej problemu i praktycznie trudno znaleźć takie, które byłoby przedmiotem zainteresowania w wielu pracach.

W rozpatrywanych w literaturze zadaniach uwzględnia się szereg innych ograniczeń. Zwróćmy tutaj uwagę na niektóre z nich, numerując je kolejno liczbami rzymskimi.

I Dla każdego wyrobu podany jest dyrektywny termin rozpoczęcia montażu τ_1 oraz dyrektywny termin zakończenia montażu τ'_1.

II Wielkość f^k_{ij} dla niektórych lub wszystkich operacji jest wielkością stałą w trakcie wykonywania operacji, ale jej konkretna wartość jest wybierana w trakcie szukania rozwiązania w przedziale między wartością minimalnie możliwą i maksymalnie dopuszczalną.

III Niektóre lub wszystkie operacje wykonywane są jednocześnie na kilku grupach stanowisk tzn. dla operacji d_{ij} zadawany jest ciąg $k_{1ij}, k_{2ij}, \ldots, k_{lij}$. Poszczególne wyrazy tego ciągu bywają też interpretowane jako różne rodzaje zasobów / materiały, siła robocza itp. /.

5.3. K r y t e r i a   o c e n y

Jak już wcześniej podkreślano, nie istnieje kryterium uniwersalne, które można zastosować w każdym konkretnym zadaniu. Przyjęte kryterium oceny w problemie rozpatrywanym zarówno w tym rozdziale jak i w poprzednim zależy przede wszystkim od warunków panujących aktualnie w przedsiębiorstwie tzn. od tego, które z ograniczeń takich np. jak przeciążenie maszyn i urządzeń, odstępstwa od terminów dyrektywnych, są dla przedsiębiorstwa mniej istotne.

Poniższe wzory funkcji-kryteriów zostały podane w ten sposób, aby w miarę szeroko zaprezentować występującą tu różnorodność.

a/ Kryterium minimalizacji czasu trwania przedsięwzięcia tj. minimalizacji wielkości F_1

$$F_1 = \max_{i,j} t^1_{ij}$$

gdzie t^1_{ij} - moment zakończenia operacji d_{ij}.

Kryterium to jest powszechnie stosowane w rozważaniach teoretycznych przy omawianiu zarówno zadań A jak i zadań B [82], [95], [132], [164], [195]. E. Ignasiak motywując przyczynę przyjęcia takiego kryterium w rozważanych zadaniach stwierdza, że "jest to usprawiedliwione doniosłością postulatu czasu realizacji przedsięwzięć. Postulat taki jest całkowicie uzasadniony zarówno z punktu widzenia teorii jak i praktyki gospodarowania" [82 s. 15]. Wydaje się jednak, że zakres stosowania tego kryterium w zadaniach praktycznych jest ograniczony z powodów o których wspomniano na wstępie.

b/ Kryterium minimalizacji odchyleń od dyrektywnych terminów zakończenia wyrobów τ'_i, czyli minimalizacji wielkości F_2

$$F_2 = \max_{i} (\bar{\tau}'_i - \tau'_i)$$

gdzie $\bar{\tau}'_i$ - faktyczny termin wykonania wyrobu, gdy wyrób został wykonany wcześniej niż wynosi termin dyrektywny przyjmuje się wtedy, że $\bar{\tau}'_i = \tau'_i$.

Można też użyć zmodyfikowanego wzoru tego kryterium

$$F_2 = \max_i \left[a_i \left(\bar{\tau}'_i - \tau'_i\right)\right]$$

gdzie a_i współczynnik ważności i-tego wyrobu.

c/ Kryterium minimalizacji sumy odchyleń od dyrektywnych terminów zakończenia wyrobów

$$F_3 = \sum_{i=1}^{m_1} \left(\bar{\tau}'_i - \tau'_i\right)$$

bądź

$$F_3 = \sum_{i=1}^{m_1} a_i \left(\bar{\tau}'_i - \tau'_i\right)$$

d/ Kryterium minimalizacji kosztów zwiększenia mocy produkcyjnej

$$F_4 = \sum_{k=1}^{m_2} \left(c'_k - c_k\right) g_k$$

gdzie

c'_k - wymagany fundusz czasu pracy w k-tej grupie stanowisk

g_k - cena zwiększenia funduszu czasu pracy w k-tej grupie stanowisk; zwiększenie funduszu pracy może się odbywać nie tylko przy pomocy zakupów nowych maszyn i urządzeń ale też poprzez zwiększenie zmianowości.

Kryterium to stosowane jest przy rozwiązywaniu modeli minimalno-czasowych.

e/ Kryterium minimalizacji sumy kwadratowego odchylenia wymaganego funduszu czasu od dysponowanego

$$F_5 = \sum_{p=1}^{s} \sum_{k=1}^{m_1} A_{pk}^2$$

gdzie wskaźnik p oznacza p-tą jednostkę wielkości T czyli czasu realizacji harmonogramu[1/], $1 \leqslant p \leqslant s$. W kryterium tym znajduje odzwierciedlenie postulat rytmiczności wykorzystania dysponowanego funduszu czasu przy ograniczonym czasie realizacji harmonogramu. Kryterium to bywa stosowane przy wykorzystaniu metody Monte-Carlo do układania harmonogramu.

Wielkość A_{pk} jest liczona następująco [27]

$$A_{pk} = \begin{cases} 0 & \text{gdy} \quad c_k'(p) \leqslant c_k(p) \\ \dfrac{c_k'(p) - c_k(p)}{c_k(p)} & \text{gdy} \quad c_k'(p) > c_k(p) \end{cases}$$

lub

$$A_{pk} = 1 - \operatorname{sign}\left[c_k(p) - c_k'(p)\right]$$

gdzie

1/ Czas realizacji harmonogramu T może być wyliczony przy pomocy następującego wzoru: $T = \max\left\{\max_i T_i, \max_k \frac{1}{c_k} \sum_{(i;j)} f_{ij}^{k} t_{ij}\right\}$

gdzie T_i najwcześniejszy czas zrealizowania wszystkich operacji i-tego wyrobu wyliczony bez uwzględniania dostępności zasobów.

$$\operatorname{sign} x = \begin{cases} +1 & \text{gdy} \quad x > 0 \\ -1 & \text{gdy} \quad x < 0 \\ 0 & \text{gdy} \quad x = 0 \end{cases}$$

5.4. Konstruowanie rozwiązania przy pomocy metody priorytetów

Zadanie sformułowane w § 5.2 należy do zadań, których rozwiązanie polega na rozdziale m_2 różnych zasobów /przy czym ilość zasobów w każdym momencie jest z góry ustalona / pomiędzy czynności należące do m_1 sieci i określeniu momentu rozpoczęcia każdej z tych czynności.

W literaturze dla rozwiązania takiego problemu przyjmuje się szereg uproszczeń [82], [164], [195]. Zakłada się mianowicie, że dla wykonania wszystkich czynności potrzebny jest jeden rodzaj zasobu a czynności należą do jednej sieci. W realnych zadaniach takie założenia występują bardzo rzadko. W zadaniach, które można rozwiązać przy pomocy rozważanej w tej pracy metody priorytetów takich uproszczeń nie trzeba stosować.

Jak wiadomo konstruowanie rozwiązania przy pomocy metody priorytetów odbywa się w n kolejnych krokach. Dla zagadnienia omawianego w niniejszym rozdziale w każdym kolejnym kroku momenty rozpoczęcia operacji będziemy określali przy pomocy postępowania II. W postępowaniu tym, każdemu krokowi odpowiada pewien przedział czasu $[h_r^o, h_r^1]$ taki, że $h_r^o = h_{r-1}^1$ oraz z reguły w każdym kroku ma się do czynienia z konfliktem trzeciego typu /por. § 3.1. /.

W procesie konstruowania rozwiązania będą wykorzystywane niektóre z tzw. p a r a m e t r ó w m o d e l u s i e c i o w e g o, do których zalicza się

- najwcześniejszy moment rozpoczęcia operacji,
- najwcześniejszy moment zakończenia operacji,
- najpóźniejszy moment rozpoczęcia operacji,
- najpóźniejszy moment zakończenia operacji,
- całkowity zapas czasu operacji,
- wolny / swobodny / zapas czasu operacji,
- niezależny zapas czasu operacji,
- najwcześniejszy czas zrealizowania wszystkich operacji wykonywanych na wyrobie.

Dla określenia przedziału $[h_r^o, h_r^1]$ wykorzystane będą: najwcześniejszy moment rozpoczęcia operacji, oznaczać go będziemy symbolem y_{ij}, oraz najwcześniejszy moment zakończenia operacji, oznaczać go będziemy symbolem $\bar{y}_{ij}$. Ponadto podczas likwidacji konfliktu będą wykorzystywane takie parametry modelu sieciowego jak:

- zapas całkowity operacji, oznaczymy go przez z_{ij},
- najwcześniejszy czas zrealizowania wszystkich operacji wykonywanych na i-tym wyrobie lub inaczej należących do i-tej sieci; oznaczymy go symbolem T_i.

W każdym kroku konstruowania rozwiązania należy znać aktualne wartości trzech wymienionych parametrów dla wszystkich operacji nie wprowadzonych do harmonogramu. Znalezienie wartości tych parametrów sprowadza się do wyznaczenia harmonogramu w tzw. sieciowym modelu minimalno-czasowym czyli harmonogramu, w którym nie uwzględnia się ograniczoności zasobów/I kierunek badań/[1/].

Dla tak wyliczonego harmonogramu za pomocą metody priorytetów będzie dokonywana korekta.

Na temat algorytmów wyznaczania harmonogramów, w których nie uwzględnia się ograniczoności zasobów istnieje bogata literatura m.in. podręczniki akademickie /por. np. [15] , [19], [20] , [81] , [195] . Ponadto w oprogramowaniu prawie każdego komputera znajdują się programy standardowe służące do wyznaczania tego rodzaju harmonogramów. Wobec powyższego nie będziemy podawać w tym miejscu algorytmów do wyliczenia parametrów y_{ij}, $\bar{y}_{ij}$, Z_{ij} oraz T_i .

Poniżej podamy interpretację niektórych pojęć dotyczących metody priorytetów takich jak: operacja gotowa do wykonania, konflikt oraz oraz określimy sposób wyznaczania w r-tym kroku przedziału $[h_r^o, h_r^1]$.

O operacji d_{ij} powiemy, że **jest gotowa do wykonania w r-tym kroku**, jeżeli najwcześniejszy moment rozpoczęcia tej operacji czyli wielkość y_{ij} równa się wielkości h_r^o.

Dla określenia przedziału $[h_1^o , h_1^1]$ należy uporządkować rosnąco najwcześniejsze momenty rozpoczęcia y_{ij} oraz najwcześniejsze momenty zakończenia $\bar{y}_{ij}$ wszystkich operacji.

1/ O takim zadaniu mówi się, że sprowadza się ono do obliczenia podstawowych parametrów modelu sieciowego [13].

Najmniejsza z tych wartości wyznacza początek przedziału czyli h^o_1 a następująca po niej wartość, koniec przedziału, czyli h^1_1.

Przy $r = 2,3,\ldots,$ dla określenia przedziału $[h^o_r, h^1_r]$ trzeba uporządkować rosnąco wielkości y_{ij} tych operacji, które nie zostały wprowadzone do harmonogramu[1/] oraz wielkości t^1_{ij} tych operacji, które zostały wprowadzone do harmonogramu w poprzednich krokach przy tym $t^1_{ij} \geqslant h^1_{r-1}$, gdzie t^1_{ij} - moment zakończenia operacji d_{ij}. Podobnie przy $r = 1$ najmniejsza z tych wartości wyznacza h^o_r a następująca po niej wartość h^1_r.

W każdym r-tym kroku / $r > 1$ / w każdej k -tej grupie stanowisk może wystąpić następująca sytuacja:

- w przedziale $[h^o_r, h^1_r]$ wykonywany jest pewien zbiór operacji oznaczony symbolem Q^k_1 (r) , zbiór ten zawiera operacje wprowadzone do harmonogramu w poprzednich krokach i nie zakończone do momentu h^o_r,
- w przedziale $[h^o_r , h^1_r]$ jest gotowa do wykonania pewna ilość operacji, będą one tworzyły zbiór Q^k_2 (r) .

W kroku pierwszym zbiór Q^k_1 (1) jest oczywiście zbiorem pustym, a w odpowiadającym temu krokowi przedziale $[h^o_1 , h^1_1]$

1/ Zauważmy, że gdy w r-tym kroku pewna ilość operacji gotowych do wykonania nie została wprowadzona do harmonogramu wówczas wielkość y_{ij} dla tych operacji w r + 1-szym kroku będzie się równała h^1_r .

będzie natomiast pewna liczba operacji gotowych do wykonania tworzących zbiór Q_2^k (1).

Podamy teraz na czym polega w r-tym kroku konflikt trzeciego typu /por. § 3.2 /.

W r-tym kroku / $r = 1, 2, 3, \ldots$ / konflikt trzeciego typu w odniesieniu do k-tej grupy stanowisk występuje wtedy gdy

$$\sum_{d_{ij} \in Q_3^k(r)} f_{ij}^k > c_k \tag{29}$$

gdzie $Q_3^k(r) = Q_1^k(r) \cup Q_2^k(r)$.

Likwidacja tego konfliktu będzie polegała na wybraniu ze zbioru $Q_2^k(r)$ pewnego podzbioru $\bar{Q}_2^k(r)$, takiego przy którym

$$\sum_{d_{ij} \in Q_3^k(r)} f_{ij}^k \leqslant c_k \,, \tag{30}$$

gdzie $Q_3^k(r) = Q_1^k(r) \cup \bar{Q}_2^k(r)$.

Przebieg konstruowania rozwiązania można przedstawić w sposób następujący:

K r o k p i e r w s z y / r = 1 /

1. Kładziemy $k = 1$ i obliczamy parametry modelu sieciowego y_{ij}, $\bar{y}_{ij}$, Z_{ij} oraz określamy przedział $[h^0_1, h^1_1]$.

Niech D oznacza zbiór operacji wykonywanych na m_1 wyrobach,

$D = \{d_{ij}\}$ $i = 1,2,\dots,m_1$, $j = 1,2,\dots,n_1$.

2. Wśród operacji należących do zbioru D wybieramy operacje wykonywane na k-tej grupie stanowisk. Wybrane operacje będą tworzyły zbiór oznaczony symbolem D_k (1) / w przypadku r -tego kroku ($r = 2,3,\dots$) symbolem D_k (r) /.
3. Określamy elementy zbioru oznaczonego symbolem Q_2^k (1) . Do tego zbioru będzie należała każda taka operacja ze zbioru D_k (1) , dla której wielkość y_{ij} równa się wielkości h_1^o .

   Gdy $Q_2^k (1) = \emptyset$ przechodzimy do punktu 6 w przeciwnym przypadku do punktu następnego.
4. Sprwadzamy czy mamy do czynienia z konfliktem. Jeżeli konflikt nie występuje wtedy dla każdej operacji należącej do zbioru Q_2^k (1) określamy jej moment rozpoczęcia $t_{ij}^o = h_1^o$.

   W ten sposób operacje należące do zbioru Q_2^k (1) wprowadzone zostały do harmonogramu. Operacje te "wykreślamy" ze zbioru D_k (1) otrzymując w ten sposób zbiór $\bar{D}_k$ (2) i przechodzimy do punktu 6.

   W przeciwnym przypadku tj. gdy występuje konflikt przechodzimy do punktu następnego.
5. Konflikt likwidujemy według jednego z opisanych w dalszej części sposobów A,B,C. Dla wybranych w wyniku likwidacji konfliktu operacji określamy moment ich rozpoczęcia $t_{ij}^o = h_1^o$.

   Po "wykreśleniu" tych operacji ze zbioru D_k (1) otrzymujemy zbiór $\bar{D}_k$(2). Następnie dla operacji nie wprowadzonych do harmonogramu obliczamy aktualne parametry modelu sieciowego tj. wielkości y_{ij} oraz z_{ij} .

6. Zwiększamy k o jeden. Gdy $k > m_1$ przechodzimy do kroku drugiego / $r = 2$ /, w przeciwnym wypadku do punktu 2.

K r o k i n a s t ę p n e / $r = 2,3,\ldots$ /

1. Określamy przedział $[h_r^o, h_r^1]$. Kładziemy $k = 1$,
2. Sprawdzamy czy zbiór $D_k(r) = \emptyset$. Jeżeli tak, to przechodzimy do punktu 6, w przeciwnym przypadku do punktu następnego.
3. Określamy elementy zbiorów $Q_1^k(r)$ oraz $Q_2^k(r)$. Do zbioru $Q_1^k(r)$ będą należały operacje wprowadzone do harmonogramu w krokach mniejszych od r, dla których wielkość $t_{ij}^1 > h_r^o$. Natomiast do zbioru $Q_2^k(r)$ będzie należała każda taka operacja ze zbioru $D_k(r)$, dla której wielkość $y_{ij} \leq h_r^o$.
4. Punkt ten jest analogiczny jak punkt 4 w kroku pierwszym
5. Punkt ten jest analogiczny jak punkt 5 w kroku pierwszym
6. Zwiększamy k o jeden. Gdy $k \leq m_1$ przechodzimy do punktu 2. Gdy natomiast $k > m_1$ oraz zbiory $D_k(r)$ dla $k = 1,2,\ldots,m_1$ były puste zadanie zostało rozwiązane, w przeciwnym przypadku tj. gdy zbiory $D_k(r) \neq \emptyset$, zwiększamy r o jeden i przechodzimy do punktu 1.

Poniżej podamy wzory funkcji priorytetu a następnie niektóre sposoby likwidacji konfliktu.

W przypadku konfliktu wartości funckji priorytetu są wyliczane dla każdego elementu zbioru $Q_2^k(r)$. Dla każdego elementu tego zbioru zadawany jest wtedy wektor X

$$X = (x_1, x_2, x_3, x_4)$$

gdzie

x_1 - całkowity zapas czasu operacji

x_2 - czas trwania operacji

x_3 - fundusz mocy produkcyjnej wymagany do wykonania operacji w każdym momencie jej trwania

x_4 - najwcześniejszy czas zrealizowania wszystkich operacji wykonywanych na i-tym wyrobie.

W różnych sposobach likwidacji konfliktu mogą być zastosowane następujące funkcje priorytetu:

1° $w_1 = G_1(X) = x_1$

O całkowitym zapasie czasu można powiedzieć, że jest to rezerwa czasu będąca w całości do dyspozycji danej operacji pod warunkiem, że wszystkie operacje poprzedzające ją zakończą się w najwcześniejszym momencie a operacje następne rozpoczną się w najpóźniejszym dopuszczalnym dla nich momencie tzn. w takim, że wyliczony wcześniej czas zrealizowania wszystkich operacji sieci - T_1 nie ulegnie zmianie. Operacje tzw. k r y t y c z n e mają zapas całkowity równy zeru i one właśnie w przypadku konfliktu będą wybierane przed innymi operacjami. Funkcja ta powinna występować jako pierwsza w każdym zastosowanym sposobie likwidacji konfliktu ponieważ użycie jej zapobiega nadmiernemu przekroczeniu czasu T_1.

2° $w_2 = G_2(X) = x_2$

3° $w_3 = G_3(X) = x_3$

4° $w_4 = G_4(X) = x_4$

5° $w_5 = G_4(X) = \frac{x_2}{x_4}$

A oto przykładowe sposoby likwidacji konfliktów spotykanych w procesie konstruowania rozwiązania.

Sposób A.

Wśród operacji uczestniczących w konflikcie - elementów zbioru $Q_2^k(r)$, należy wybierać kolejno operacje według minimalnych wartości funkcji $G_1(X)$ tworząc w ten sposób zbiór $\bar{Q}_2^k(r)$. Gdy dla pewnych operacji, spośród których część będzie zakwalifikowana do podzbioru $\bar{Q}_2^k(r)$, wartości funkcji $G_1(X)$ są równe, wtedy dla jednoznacznego wyboru należy zastosować funkcję $G_2(X)$, wybierając do podzbioru $\bar{Q}_2^k(r)$ operacje kolejno według maksymalnych wartości tej funkcji.
W przypadku gdy i ta funkcja nie zapewnia dla niektórych operacji jednoznacznego wyboru stosować należy funkcję $G_3(X)$.

Sposób B.

Aby określić, które operacje będą tworzyły podzbiór $\bar{Q}_2^k(r)$ należy kolejno stosować funkcję $G_1(X)$, funkcję $G_3(X)$ a następnie funkcję $G_2(X)$ oraz $G_4(X)$ wybierając według minimalnych wartości tych funkcji.

Sposób C.

Dla określenia operacji należących do zbioru $\bar{Q}_2^k(r)$ należy zastosować następujący ciąg funkcji priorytetu:
$G_1(X)$, $G_5(X)$, $G_3(X)$.

Dodajmy, że w każdym z podanych sposobów, jeżeli zadany ciąg funkcji nie zapewnia jednoznacznego wyboru, należy zastosować wybór loso[illegible] wybierając operację z jednakowym pr[illegible]dobieństwem.

5.5 P r z y k ł a d [1/]

Na pewnym odcinku produkcyjnym należy wykonać montaż wyrobu / $i = 1$ /. Odcinek produkcyjny składa się z dwóch stanowisk / $k = 1, 2$ /. Dysponowany fundusz mocy produkcyjnej / c_k / dla każdego stanowiska wynosi jedną jednostkę.
Aby wyrób został zamontowany, należy wykonać 11 operacji, / $j = 1, \ldots, 11$ /.
Kolejność w jakiej operacje te muszą być wykonywane zadana jest przy pomocy macierzy A / por. wzór (26) /:

$$A = \begin{bmatrix} 0 & 0 & 1 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 1 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \end{bmatrix}$$

Czasy t_{ij} trwania operacji oraz stanowiska, na których musi być wykonywana każda operacja, podane są w tablicy 4.
W każdym momencie, na każdym stanowisku wykonywana jest jedna operacja, która z chwilą rozpoczęcia nie może być przerwana. Ponieważ $c_k = 1$ oznacza to, że fundusz mocy produkcyjnej wymagany do wykonania operacji w każdym momencie jej realizacji równa się jedności / $r^k_{ij} = 1$ /.
Dla każdej operacji należy określić moment jej rozpoczęcia t_{ij} *

1/ Dane liczbowe do przykładu zaczerpnięto z pracy [177]

Przy pomocy metody priorytetów zostaną wyliczone dwa warianty rozwiązania przy czym w przypadku generowania pierwszego wariantu będzie stosowany pierwszy sposób likwidacji konfliktów / sposób A / natomiast w przypadku drugiego rozwiązania sposób B.

Tablica 4.

Nr operacji (j)	Czas trwania (t_{ij})	Nr stanowiska (k)
1	3	1
2	3	1
3	2	1
4	4	2
5	2	1
6	2	2
7	5	1
8	4	1
9	2	2
10	5	2
11	4	2

Krok pierwszy

1. $k = 1$

$$D = \{d_1, d_2, d_3, d_4, d_5, d_6, d_7, d_8, d_9, d_{10}, d_{11}\}$$

Parametry modelu sieciowego y_{ij}, $\bar{y}_{ij}$ oraz z_{ij} są następujące:[1/]

1/ Parametry modelu wyliczono korzystając z programu bibliotecznego dla maszyny Odra 1204 .
Ponieważ w zadaniu rozpatrywany jest jeden wyrób, wobec tego parametry modelu sieciowego będziemy oznaczać symbolami y_j, $\bar{y}_j$, z_j / $j = 1,2,\ldots,11$ / a operacje symbolem d_j / $j = 1,2,\ldots,11$ /.

$$
\begin{array}{llll}
y_1 = 0 \,, & \bar{y}_1 = 3 \,, & z_1 = 2 \\
y_2 = 0 \,, & \bar{y}_2 = 3 \,, & z_2 = 0 \\
y_3 = 3 \,, & \bar{y}_3 = 5 \,, & z_3 = 2 \\
y_4 = 3 \,, & \bar{y}_4 = 7 \,, & z_4 = 0 \\
y_5 = 3 \,, & \bar{y}_5 = 5 \,, & z_5 = 4 \\
y_6 = 7 \,, & \bar{y}_6 = 9 \,, & z_6 = 0 \\
y_7 = 7 \,, & \bar{y}_7 = 12 \,, & z_7 = 0 \\
y_8 = 9 \,, & \bar{y}_8 = 13 \,, & z_8 = 0 \\
y_9 = 12 \,, & \bar{y}_9 = 14 \,, & z_9 = 0 \\
y_{10} = 13 \,, & \bar{y}_{10} = 18 \,, & z_{10} = 0 \\
y_{11} = 14 \,, & \bar{y}_{11} = 18 \,, & z_{11} = 0
\end{array}
$$

Porządkujemy rosnąco wielkości y_j oraz $\bar{y}_j$. Najmniejsza wartość w tak uporządkowanym ciągu wynosi 0, następująca po niej wartość wynosi 3. Wobec tego $h_1^0 = 0$, $h_1^1 = 3$.

2. Określamy zbiory D_1 (1) oraz Q_2^1 (1) / por. punkt 2 oraz 3 algorytmu /

$$D_1\ (1) = \{d_1, d_2, d_3, d_5, d_7, d_8\}$$

$$Q_2^1\ (1) = \{d_1, d_2\}$$

3. Sprawdzamy czy występuje konflikt /patrz wzór (29) /. Ponieważ

$$\sum_{d_j \in Q_2^1 (1)} f_j^1 = 2$$

oraz $c_1 = 1$, wobec tego występuje konflikt. Likwidując konflikt według sposobu A wybiera się operację d_2. Moment rozpoczęcia tej operacji równa się wielkości h_1^0. W ten spo-

sób operacja d_2 została wprowadzona do harmonogramu.
Fakt ten zapisujemy w tablicy 5.
Obliczamy parametry modelu modelu sieciowego dla operacji nie wprowadzonych do harmonogramu tj. wielkości y_j oraz z_j / por. punkt 5 algorytmu /.

$y_1 = 3$, $z_1 = 0$
$y_3 = 6$, $z_3 = 0$
$y_4 = 3$, $z_4 = 1$
$y_5 = 6$, $z_5 = 2$
$y_6 = 8$, $z_6 = 0$
$y_7 = 8$, $z_7 = 0$
$y_8 = 10$, $z_8 = 0$
$y_9 = 13$, $z_9 = 0$
$y_{10} = 14$, $z_{10} = 0$
$y_{11} = 15$, $z_{11} = 0$

4. $k = 2$

Określamy zbiory D_2 (1) oraz Q_2^2 (1) / punkt 2 oraz 3 algorytmu /

$D_2 (1) = \{d_4, d_6, d_9, d_{10}, d_{11}\}$

$Q_2^2 (1) = \emptyset$

Ponieważ zbiór Q_2^2 (1) jest zbiorem pustym oraz $k = 2$, przechodzimy do kroku następnego / patrz punkt 3 algorytmu /

Krok drugi

1. $h_2^0 = 3$, $h_2^1 = 6$ / por. def. na str. 125 /

2. $k = 1$

Określamy zbiory $D_1(2)$, $Q_1^1(2)$, $Q_2^1(2)$

$D_1(2) = \{d_1, d_3, d_5, d_7, d_8\}$ /por. punkt 5 algorytmu - pierwszy krok /.

$Q_1^1(2) = \emptyset$ / patrz punkt 3 algorytmu - następne kroki /

$Q_2^1(2) = \{d_1\}$

3. Konflikt nie występuje ponieważ $\sum_{d_j \in Q_2^1(2)} f_j^1 = 1.$

Parametry modelu sieciowego nie ulegają zmianie. Moment rozpoczęcia operacji d_1 równa się wielkości h_2^0 , którą zapisujemy w tablicy 5.

4. k = 2

$D_2(2) = \{d_4, d_6, d_9, d_{10}, d_{11}\}$ / por. punkt 5 algorytmu- pierwszy krok /

$Q_1^2(2) = \emptyset$ / por. punkt 3 algorytmu - następne kroki /

$Q_2^2(2) = \{d_4\}$.

5. Podobnie jak w punkcie 3 , konflikt nie występuje. Moment rozpoczęcia operacji d_4 równa się wielkości h_2^0 , którą zapisujemy w tablicy 5. Ponieważ k = 2 przechodzimy do kroku następnego. Konstruowanie rozwiązania w następnych krokach odbywa się analogicznie jak w kroku drugim. Wyliczony harmonogram zawarty jest w tablicy 5 . W tablicy 6 podany jest harmonogram, który wyliczono likwidując konflikty według sposobu B.

Tablica 5

Nr operacji	Moment rozpoczęcia operacji	Moment zakończenia operacji	Nr stanowiska
1	3	6	1
2	0	3	1
3	6	8	1
4	3	7	2
5	13	15	1
6	8	10	2
7	8	13	1
8	15	19	1
9	13	15	2
10	19	24	2
11	24	28	2

Tablica 6

Nr operacji	Moment rozpoczęcia operacji	Moment zakończenia operacji	Nr Stanowiska
1	3	6	1
2	0	3	1
3	6	8	1
4	3	7	2
5	8	10	1
6	8	10	2
7	10	15	1
8	15	19	1
9	15	17	2
10	19	24	2
11	24	28	2

W obu tablicach dla każdej operacji podany jest moment jej rozpoczęcia i zakończenia oraz stanowisko, na którym będzie wykonywana operacja. Na podstawie tych danych ła-

two sporządzić harmonogram dla stanowiska.
Ze względu na kryterium minimalizacji czasu trwania przedsięwzięcia / por. § 5.3 / harmonogramy zawarte w tablicy 5 oraz w tablicy 6 są jednakowe.

6. WYZNACZANIE HARMONOGRAMU DLA ZAGADNIEŃ Z CZĘŚCIOWĄ KOLEJNOŚCIĄ CZYNNOŚCI

6.1. S f o r m u ł o w a n i e   p r o b l e m u

Problem ten zostanie omówiony na przykładzie układania harmonogramu zajęć w szkole wyższej.

Na początku podamy listę podstawowych pojęć.

G r u p a - pewna liczba studentów połączona decyzją dziekana w wyodrębnioną całość. W przypadku zajęć laboratoryjnych bądź zajęć z języków obcych, grupa może się dzielić na kilka podgrup.

W y d z i a ł - wyodrębniona w obrębie uczelni jednostka organizacyjna składająca się z pewnej ilości grup studenckich W ramach wydziału ściśle określone grupy łączone są w tzw. lata studiów, a z kolei w ramach poszczególnych lat studiów grupy mogą tworzyć kierunki studiów.

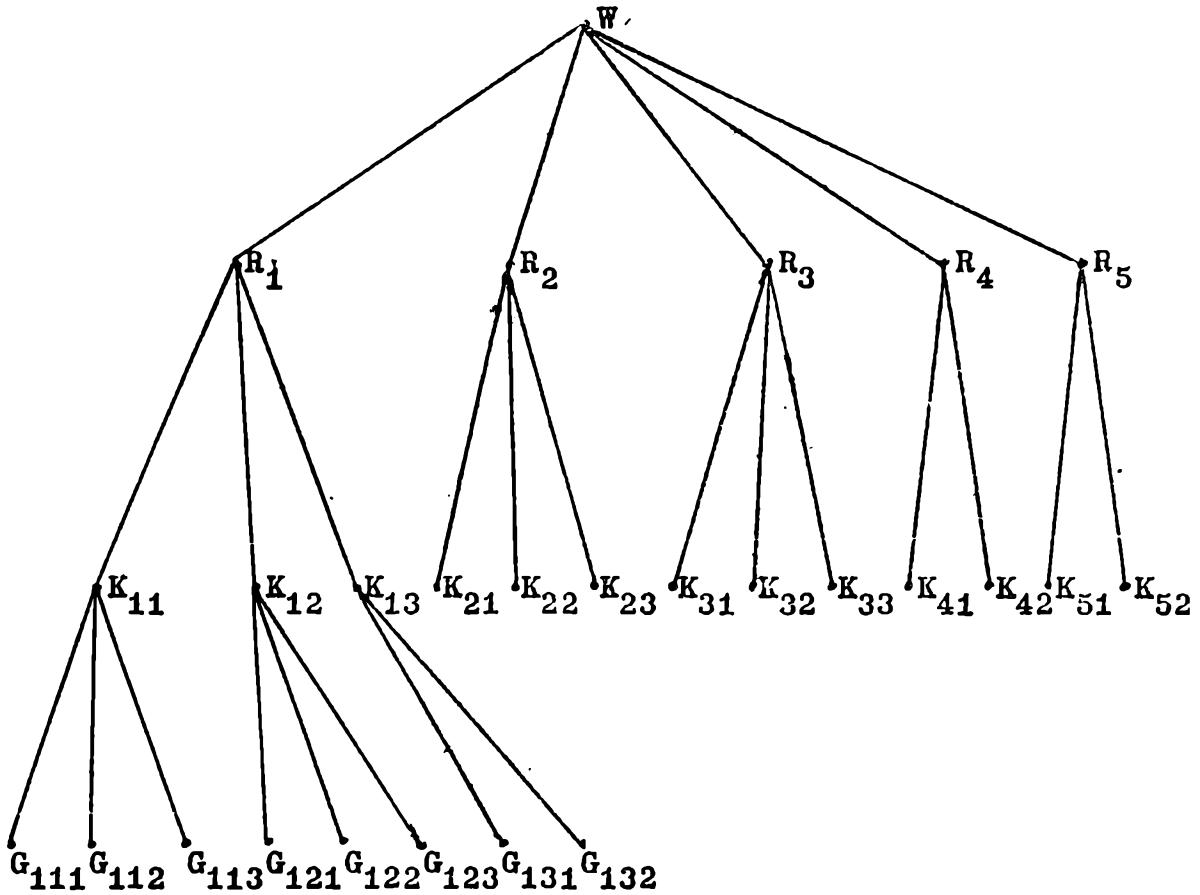
Hierarchiczną strukturę wydziału można przedstawić za pomocą grafu zwanego drzewem /patrz rys.5 str. 140 /.

P r z e d m i o t - wykład, ćwiczenie, seminarium /ogólnie zajęcie/ z pewnej dyscypliny naukowej objętej programem studiów wyższych.

W y k ł a d o w c a - ten, kto wykłada, prowadzi seminaria czy ćwiczenia, a więc ogół pracowników prowadzących zajęcia.

S a l a - pomieszczenie przeznaczone na odbywanie zajęć.

Niech $B = \{b_i\}$ $i=1,2,\ldots,m_1$ będzie zbiorem grup na pewnym wydziale. W zbiorze B wyróżnia się pewne rozłączne



W - wydział

R - rok

K - kierunek

G - grupa

Rys. 5 Struktura wydziału wyższej uczelni podzbiory B_w / $w = 1,2,\ldots,m_4$/ zwane latami. Z kolei w każdym zbiorze B_w wyróżnia się pewne rozłączne podzbiory B_w^j /$j=1,2,\ldots,p_w$/ zwane kierunkami. Symbole b_i, B_w, B_w^j będziemy też nazywali w dalszym ciągu j e d n o s t k a m i o r g a n i z a c y j n y m i w y d z i a ł u.

Wszystkie przedmioty na tym wydziale określa zbiór $D = \{d_l\}$ $l = 1,2,\ldots,m_3$. Przedmioty $d_l \in D$ prowadzone są przez wykładowców, którzy tworzą zbiór $W = \{w_r\}$ $r=1,2,\ldots,m_2$. Przedmioty odbywają się w salach tworzących zbiór $S = \{s_p\}$ $p = 1,2,\ldots,m_3$. W zbiorze S można wydzielić pewne typy sal, które tworzą podzbiory S_k /$k=1,2,\ldots,m_5$/ takie, że $S = \bigcup S_k$. Każdy przedmiot $d_l \in D$ może być scharakteryzowany w sposób jednoznaczny za pomocą trójki liczb

$$d_l = (J, w_r, s_k) \tag{31}$$

gdzie J – jednostka organizacyjna wydziału.

Tak więc każdemu elementowi $b_i \in B$ przyporządkowany jest pewien podzbiór zbioru D. Podzbiór ten oznaczymy symbolem D_i, a jego liczebność symbolem n_i. Zbiory D_i /$i=1,2,\ldots,m_1$ nie są oczywiście rozłączne, ale musi być spełniony warunek taki, że

$$D = D_1 \cup D_2 \cup \ldots \cup D_{m_1}$$

Dla pewnych przedmiotów przyporządkowanych i-tej grupie może być zadana k o l e j n o ś ć i c h r e a l i z a c j i. Kolejność ta dla i-tej grupy zadana jest wtedy przy pomocy macierzy kwadratowej A_i.

$$A_i = [a^i_{jw}] \qquad \begin{array}{l} i=1,2,\dots,m_1 \\ j=1,2,\dots,n_i \\ w=1,2,\dots,n_i \end{array} \tag{32}$$

gdzie:

$$a^i_{jw} = \begin{cases} 1 & \text{gdy } j\text{-ty przedmiot należący do podzbioru } D_i \text{ bezpośrednio poprzedza w czasie realizację } w\text{-tego przedmiotu należącego do podzbioru } D_i \\ 0 & \text{w przeciwnym przypadku.} \end{cases}$$

Przedmioty $d_l \in D$ muszą się odbywać w zadanym przedziale czasu $[0,H]$ /np. tygodnia/. Przedział $[0,H]$ dzielimy na równe odcinki przyjmując za jednostkę podziału godzinę. Utworzone w ten sposób odcinki będą tworzyły ciąg wielkości $t_1,t_2,\dots,t_j,\dots,t_H$. W tym ciągu kolejne 24 elementowe podciągi będziemy nazywali dniami tygodnia i oznaczali symbolem H_g /g=1,2,...,6/. Dla przedmiotu $d_l \in D$ podany jest czas trwania przedmiotu /w godzinach/, który oznaczać będziemy symbolem τ_l gdzie $\tau_l = 1,2,3\dots$.
Dla każdej grupy $b_i \in G$ oraz wykładowcy $w_r \in W$ podane są wektory $\bar{b}_i$ i $\bar{w}_r$, w których będą zawarte życzenia grup i wykładowców;

$$\bar{g}_i = (\bar{g}_{i1},\bar{g}_{i2},\dots,\bar{g}_{ij},\dots,\bar{g}_{iH}) \tag{33}$$

$$i=1,2,\dots,m_1$$
$$j=1,2,\dots,H$$

gdzie:

$$\bar{g}_{ij} = \begin{cases} 1 & \text{jeżeli grupa } b_i \text{ chciałaby mieć zajęcia w odcinku czasowym } t_j \\ 0 & \text{w przeciwnym przypadku,} \end{cases}$$

$$\bar{w}_r = (\bar{w}_{r1}, \bar{w}_{r2}, \ldots, \bar{w}_{rj}, \ldots, \bar{w}_{rH}) \qquad (34)$$

$$r=1,2,\ldots,m_2$$
$$j=1,2,\ldots,H$$

gdzie:

$$\bar{w}_{rj} = \begin{cases} 1 & \text{jeżeli wykładowca } w_r \text{ chciałby mieć zajęcia} \\ & \text{w odcinku czasowym } t_j \\ 0 & \text{w przeciwnym przypadku.} \end{cases}$$

Zadane są też następujące ograniczenia:

1° Dla każdego $b_i \in B$ w każdym t_j /$j=1,2,\ldots,H$/ może odbywać się jeden przedmiot $d_k \in D$ z jednym wykładowcą $w_r \in W$ oraz w jednej sali $s_p \in S$.

2° Każdy $w_r \in W$ w każdym t_j /$j=1,2,\ldots,H$/ może prowadzić tylko jeden przedmiot $d_l \in D$ z jedną grupą $b_i \in B$ oraz w jednej sali $s_p \in S$.

3° Dla każdego $b_i \in B$ w przedziale $[0,H]$ powinny być umieszczone wszystkie przedmioty $d_l \in D$.

4° Liczba sal wykorzystanych w każdym odcinku t_j /$j=1,2,\ldots T$ nie może przekraczać liczby sal dysponowanych.

Wymienione cztery warunki będziemy nazywali o g r a n i c z e n i a m i p o d s t a w o w y m i . Oprócz nich istnieje jeszcze grupa tzw. o g r a n i c z e ń d o d a t k o w y c h .

Oto lista niektórych ograniczeń dodatkowych:

1° Dla każdego $b_i \in B$ należy uwzględnić życzenia dotyczące dni wolnych i zajętych.

2^o Dla każdego $w_r \in W$ należy uwzględnić życzenia o najbardziej dogodnym dla nich czasie prowadzenia zajęć.

3^o Dla każdego $b_i \in B$ liczba odcinków t_j w każdym z poszczególnych H_g /g=1,2,...,6/, w czasie których będą się odbywać przedmioty powinny być w miarę jednakowe.

Problem ułożenia harmonogramu zajęć polega na tym aby każdemu przedmiotowi $d_l \in D$ przyporządkować pewien przedział $[t_j, t_{j+\tau_l-1}]$ gdzie j=1,2,...,H przy czym musi to być taki przedział, aby były spełnione wszystkie ograniczenia podstawowe i możliwie najwięcej ograniczeń dodatkowych. Otrzymamy w ten sposób rozwiązanie dopuszczalne lub dopuszczalny harmonogram zajęć.
Przy zadanych ograniczeniach może.oczywiście nie istnieć żadne rozwiązanie dopuszczalne. Jeżeli natomiast rozwiązanie dopuszczalne istnieje, wtedy wśród rozwiązań dopuszczalnych należy znaleźć takie rozwiązanie, które maksymalizowałoby zadaną funkcję-kryterium.

6.2. Przegląd metod

Problem układania harmonogramu zajęć, podobnie jak wcześniej omówione dwa zagadnienia należy do klasy zadań kombinatorycznych traktujących o porządkowaniu w czasie zbiorów skończonych. Istniejące metody rozwiązania tego problemu można podzielić na dwie grupy:

a/ metody, które pozwalają uzyskać rozwiązanie o którym z reguły trudno powiedzieć czy jest optymalne,

b/ metody, które pozwalają uzyskać wszystkie rozwiązania bądź jedno rozwiązanie, które jest rozwiązaniem optymalnym.

Do grupy pierwszej należy metoda priorytetów omówiona szczegółowo w niniejszej pracy oraz metody probabilistyczne [110] Natomiast do grupy drugiej należy zaliczyć metodę kolorowania grafu [108] , metodę szukania skojarzeń w grafie Königa [37] oraz metody programowania matematycznego.

Ideę metody kolorowania grafu można przedstawić następująco. Niech symbol H^{l}_{ir} oznacza zbiór tych odcinków t_j, w których zarówno grupa b_i jak i wykładowca w_r mogą mieć przedmiot d_l. Przyjmijmy, że $\tau_l = 1$ oraz że poszczególne przedmioty d_l prowadzone są dla jednej grupy b_i. Ponieważ zbiorów H^{l}_{ir} będzie tyle ile wynosi wartość m_3, to każdy taki zbiór może być oznaczony w sposób jednoznaczny przy pomocy symbolu H^{l}.

Zdefiniujmy pewną relację β określoną na zbiorach $H^{l_1}_{i_1 r_1}$ i $H^{l_2}_{i_2 r_2}$:

$$H^{l_1} \beta H^{l_2} = (l_1 \neq l_2 \wedge ((i_1 = i_2) \vee (r_1 = r_2))) .$$

Tak więc $H^{l_1} \beta H^{l_2}$ wtedy i tylko wtedy, gdy przedmioty d_{l_1} oraz d_{l_2} są prowadzone w tej samej grupie lub przez tego samego wykładowcę, a takie przedmioty muszą, jak wiadomo, odbywać się w różnych t_j.

Niech $\bar{H} = (t_{j_1}, t_{j_2}, \dots, t_{j_l}, \dots, t_{j_{m_3}})$

będzie ciągiem składającym się z elementów ciągu $t_1, \dots, t_j,$ $\dots, t_H$ takim, że jeżeli

$$t_{j_{l_1}} \in H^{l_1} \quad \text{i} \quad t_{j_{l_1}} \in \bar{H} \quad \text{oraz} \quad H^{l_1} \beta H^{l_2}, \quad \text{to} \quad t_{j_{l_1}} \neq t_{j_{l_2}}$$

gdzie $t_{j_{l_2}} \in \bar{H}$.

Ciąg $\bar{H}$ spełniający poniższe warunki będzie dopuszczalnym rozkładem zajęć[1]. Przez $\bar{\bar{H}}$ będziemy oznaczać zbiór wszystkich $\bar{H}$ spełniających te warunki.

Przyjmijmy, dalej, że C jest następującą rodziną zbiorów:

$$C = \{H^1, \dots, H^{m_3}, \ \{t_1\}, \dots, \{t_H\}\} .$$

Na rodzinie zbiorów C określamy relację β' spełniającą następujące warunki:

$$H^{l_1} \beta' H^{l_2} \equiv H^{l_1} \beta H^{l_2}$$

$$H^l \beta' \{t_j\} \equiv \{t_j\} \beta' H^l \doteq t_j \notin H^l$$

$$\{t_{j_1}\} \beta' \{t_{j_2}\} \equiv t_{j_1} \neq t_{j_2} .$$

1/ Dowód twierdzenia, na podstawie którego podano takie sformułowanie można znaleźć w pracy [108]

Graf $G = (C, \beta)$ gdzie C jest zbiorem wierzchołków a β zbiorem krawędzi będziemy nazywać grafem rozkładu zajęć. Ponieważ jest to graf niezorientowany, bez pętli oraz bez krawędzi wielokrotnych można go pokolorować, co jak wiadomo polega na szukaniu minimalnej liczby kolorowań /liczby chromatycznej/ wszystkich wierzchołków grafu, tak aby dwa sąsiednie wierzchołki nie miały jednakowych kolorowań. W pracy [108] podany jest dowód następującego twierdzenia:

Zbiór $\overset{=}{H}$ jest niepusty wtedy i tylko wtedy, gdy liczba chromatyczna grafu G jest równa liczbie H.
Jest to kryterium istnienia harmonogramu zajęć przy szukaniu rozwiązania za pomocą kolorowania grafu.
Każdemu pokolorowaniu grafu nG spełniającemu kryterium istnienia odpowiada jednoznacznie dopuszczalny rozkład zajęć $\bar{H} \in \overset{=}{H}$. Przyjmuje się tutaj, że każdemu oznaczeniu wierzchołka H^1 będzie odpowiadać określony odcinek czasu t_j, w którym będzie się odbywał przedmiot. Przy pomocy tej metody otrzymujemy wszystkie możliwe harmonogramy zajęć, spośród których wybieramy optymalny.
Istnieje jednak zasadnicza trudność w praktycznym zastosowaniu opisanej metody. Znane obecnie algorytmy kolorowania grafu są bardzo złożone i nawet jedno pokolorowanie tak dużego grafu jakim będzie w konsekwencji graf G jest nie do pokonania w realnym czasie przez maszynę cyfrową.
Metoda ta jest jednak często w literaturze omawiana szczególnie wtedy, gdy problem jest rozpatrywany przez matematyków numeryków [186], [190]. Dla nich metoda ta stanowi swego

rodzaju poligon doświadczalny. Starają się dociec, o których rozwiązaniach można sądzić, że nie są optymalne już na samym początku konstruowania rozwiązania.

Inną metodą rozwiązania omawianego problemu jest metoda szukania skojarzeń w grafie Königa, którą zaproponowano w pracy [37] . Przedstawiono tam uogólniony problem przydziału jako zadanie wyznaczania wszystkich skojarzeń w grafie Königa oraz przedstawiono algorytm rozwiązania tego typu zadań. Dalej zaprezentowano możliwość zastosowania tej metody do układania harmonogramu zajęć. Pokazano mianowicie, że układanie harmonogramu zajęć można zdefiniować jako pewien uogólniony problem przydziału. Budując dla zdefiniowanego problemu graf Königa, należy szukać skojarzeń spełniających pewne ograniczenia. Każde takie skojarzenie jest dopuszczalnym harmonogramem zajęć.
Metoda ta ma podobne wady jak omówiona wcześniej metoda kolorowania grafu. Przy tak dużej objętości tego zadania wyznaczenie jednego skojarzenia przy pomocy komputera w realnym czasie jest niemożliwe.

Są czynione próby zastosowania metod programowania matematycznego do rozwiązania omawianego problemu [116] , [162] , [163]. Jak dotychczas przy zastosowaniu tych metod rozwiązywane są przypadki bardzo uproszczone i nie wydaje się aby metody programowania matematycznego mogły być stosowane do zadań praktycznych.

Rozwiązanie zadania praktycznego możliwe jest obecnie jedynie przy pomocy metod heurystycznych. Podkreślają to wyraźnie tak znane autorytety naukowe jak D.Judin [125] , L.Tierechow [182]

Dociekania autora wydają się potwierdzać tę tezę, ponieważ dysponując kompletnymi materiałami o wielkości realnego zadania czynił próby rozwiązania zadania przy zastosowaniu metody kolorowania grafu oraz metody szukania skojarzeń w grafie Königa. Zakończyły się one niepowodzeniem.

W literaturze opisano szereg metod heurystycznych rozwiązania omawianego problemu [94] , [103] , [110] , [119] , [157] . Podane algorytmy w dużym stopniu odzwierciedlają specyfikę uczelni, dla których były opracowywane. Niektóre z nich uwzględniają ograniczenie bardzo szczegółowo [94] , [110] , /np. fakt, że niektórzy wykładowcy nie godzą się na zajęcia w salach daleko położonych/, co powoduje, że algorytm jest bardzo skomplikowany.

Poniżej pokazano zastosowanie heurystycznego sposobu postępowania, nazwanego w tej pracy metodą priorytetów, do rozwiązania zadania sformułowanego w § 6.1.

6.3. Układanie harmonogramu przy pomocy metody priorytetów

6.3.1. Wprowadzenie

Ważnym zagadnieniem przy opracowywaniu programu realizującego układanie harmonogramu jest sposób zapisu w pamięci komputera zbiorów opisanych w § 5.2. W języku profesjonalnym mówi się wtedy o formie zapisu danych wejściowych i wyjściowych.

Przy opracowywaniu programu zamieszczonego w Aneksie korzystano z zapisu macierzowego.[1] Poniżej przedstawimy w ten sposób zbiory określone w § 5.2, a następnie podamy algorytm rozwiązania biorąc za podstawę tak przedstawione zbiory.

Zgodnie z tym co zostało podane przy formułowaniu problemu /§ 6.1/ niektóre przedmioty ze zbioru D mogą być prowadzone dla kilku grup, które tworzą razem kierunek lub rok studiów. Oznaczymy większe jednostki organizacyjne niż grupa kolejnymi liczbami naturalnymi $1,2,\ldots,j,\ldots,n_1$.

W celu określenia jakie numery grup wchodzą w skład j-tej jednostki organizacyjnej zadamy macierz Q

$$Q = [q_{jm}] \qquad \begin{array}{l} j=1,2,\ldots,n_1 \\ m=1,2,\ldots,n_4 \end{array} \qquad (35)$$

gdzie

$$n_4 = \max_j n_j$$

$n_j = 2,3,\ldots$ i oznacza ilość grup wchodzących w skład j-tej jednostki organizacyjnej,

$q_{jm} > 0$ oznacza numer grupy, która wchodzi w skład j-tej jednostki organizacyjnej $/1 \leq q_{jm} \leq m_1/$.

Z kolei w celu określenia jakie numery sal tworzą k-ty typ sali zadamy macierz G

$$G = [g_{kj}] \qquad \begin{array}{l} k=1,2,\ldots,m_5 \\ j=1,2,\ldots,n_6 \end{array} \qquad (36)$$

---

[1] Taką samą formę zapisu zbiorów przyjęto przy przygotowywaniu programu na maszynę Odra 1204 realizującego układanie produkcji harmonogramu produkcji detali /por. [68]/.

gdzie

$$n_6 = \max_k j_k$$

$j_k = 1,2,3,\ldots$ i oznacza ilość sal tworzących k-ty typ sali

$g_{kj} > 0$ oznacza numer sali wchodzącej do k-tego typu sali $/1 \leq g_{kj} \leq m_3/$.

Dla każdego przedmiotu $d_l \in D$ zadany jest wektor

$$z_l = (J, \tau_l, w_r, s_k) \qquad (37)$$

przy czym interpretacja poszczególnych symboli stanowiących składowe tego wektora jest identyczna jak w § 6.1.

Dla każdej grupy, wykładowcy i sali zadajemy macierz C zwaną dalej odpowiednio macierzą czasową grupy, wykładowcy, sali.

$$C = [c_{wn}] \qquad (38)$$

gdzie:

$w = 1,2,\ldots,6$ i oznacza dzień tygodnia,

$n = 1,2,\ldots,n_3$ i oznacza numer godziny w ramach dnia tygodnia.

Na temat co oznacza element macierzy C będzie powiedziane poniżej.

Odpowiednim wskaźnikiem macierzy C odpowiadają oczywiście ściśle określone odcinki czasowe t_j $/j=1,2,\ldots,H/$ zdefiniowane w § 6.1.

Dla i-tej grupy $/i=1,2,\ldots,n_1/$ macierz czasową będziemy oznaczali symbolem $C_i = [c_{wn}]$, dla r-tego wykładowcy $/r=1,2,\ldots,m_2/$ $C'_r = [c'_{wn}]$ oraz dla p-tej sali $/p=1,2,\ldots,m_3/$

$C''_p = [c''_{wn}]$. Przed przystąpieniem do konstruowania rozwiązania poszczególne elementy c_{wn} w macierzach czasowych grup oraz wykładowców równają się jedności jeżeli grupa czy też wykładowca chciałby mieć zajęcia w czasie (w,n) a zero w przeciwnym przypadku. Wszystkie elementy macierzy czasowych sal będą się natomiast równały jedności.

W procesie konstruowania rozwiązania, wartości odpowiednich elementów w macierzach czasowych będą ulegać zmianie. I tak, jeżeli w dowolnym kroku konstruowania rozwiązania przedmiowoti d_l prowadzonemu w i-tej grupie $/i=1,2,\ldots,m_l/$ lub w $i_1,i_2,\ldots$ grupach został przyporządkowany przedział $[t_j, t_{j+\tau_l-1}]$ czyli według nowych oznaczeń przedział $[(w,n)\ ,(w,n+\tau_l-1)]$ oraz w tym przedziale p-ta sala, wtedy odpowiadająoe temu przedziałowi elementy macierzy czasowych:

- w macierzy czasowej grupy /lub macierzach czasowych grup/ elementy $c_{wn},\ldots,c_{w,n+\tau_l-1}$ będą się równały numerowi wykładowcy,
- w macierzy czasowej wykładowcy elementy $c'_{wn},\ldots,c'_{wn},\ldots,c'_{w,n+\tau_l-1}$ będą się równyły numerowi sali,
- w macierzy czasowej sali elementy $c''_{wn},\ldots,c''_{w,n+\tau_l-1}$ będą się równały numerowi przedmiotu.

Po rozwiązaniu zadania, co jest równoznaczne z przydzieleniem wszystkim przedmiotom $d_l \in D$ przedziałów czasowych oraz sal w tych przedziałach wszystkie dane o przydziale będą zawarte w macierzach czasowych grupy, wykładowców i sal.

Na podstawie tych danych można ocenić jakość otrzymanego rozwiązania wprowadzając pewne kryteria oceny.

6.3.2. Kryteria oceny

Pierwsza grupa kryteriów związana jest z minimalizacją "okienek" czyli przerw między zajęciami. Algorytm liczenia ilości "okienek" w macierzach czasowych przedstawimy za pomocą schematu blokowego /str. 154 /.

Podamy teraz następujące kryteria oceny:

a/ Kryterium minimalizacji ogólnej liczby okienek w macierzach czasowych grup

$$F_1 = \sum_{i=1}^{m_1} L_i$$

gdzie L_i - liczba okienek w i-tej grupie.

b/ Kryterium minimalizacji średniej liczby okienek w grupie

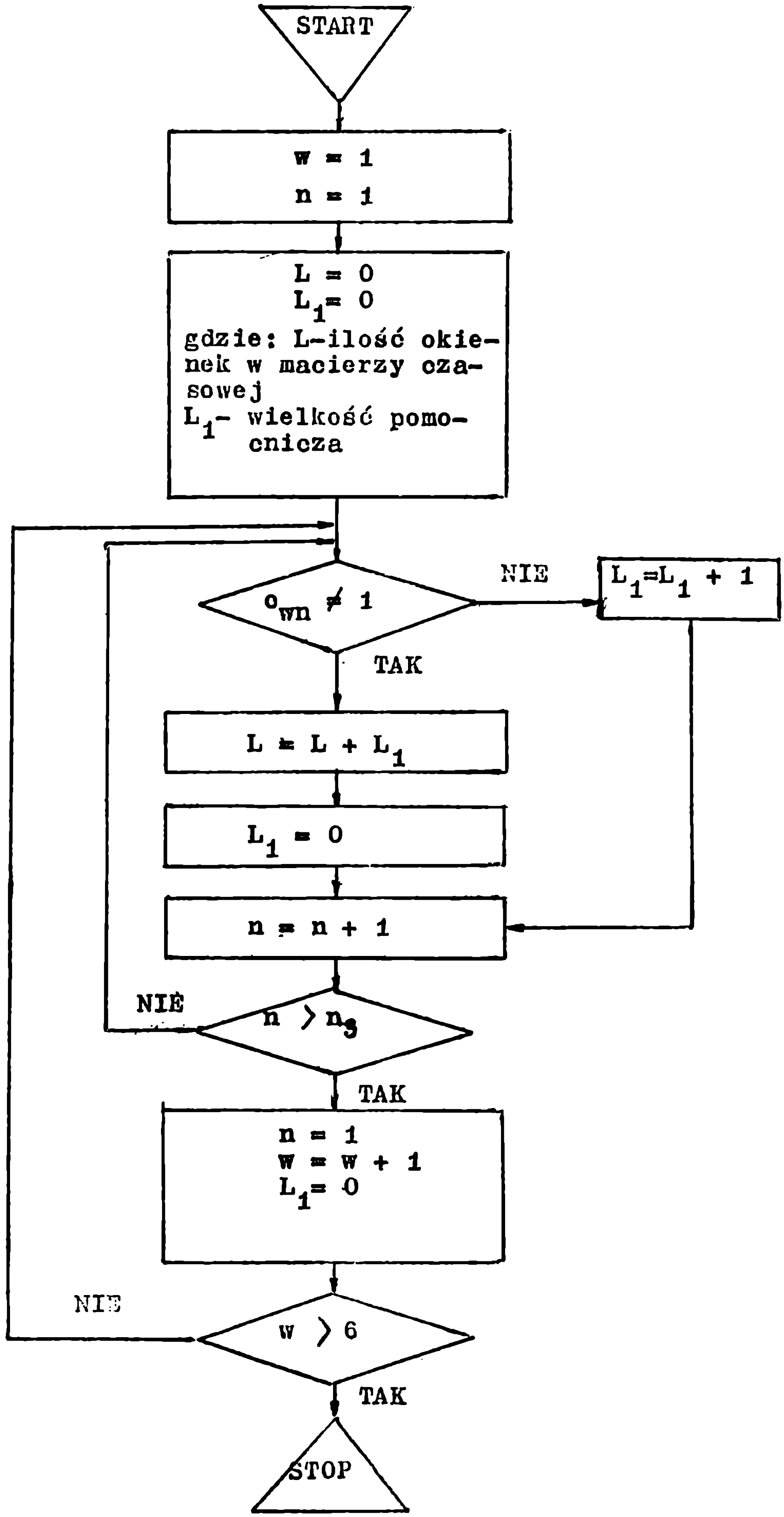
$$F_2 = \frac{1}{m_1} \sum_{i=1}^{m_1} L_i$$

c/ Kryterium minimalizacji maksymalnej liczby okienek w i-tej grupie

$$F_3 = \max_i L_i$$

d/ Kryterium minimalizacji ogólnej liczby okienek w macierzach czasowych wykładowców

$$F_4 = \sum_{r=1}^{m_2} L_r$$



START
w = 1
n = 1
L = 0
L1= 0
gdzie: L-ilość okie-
nek w macierzy cza-
sowej
L1- wielkość pomo-
cnicza
own ≠ 1
NIE
L1=L1 + 1
TAK
L = L + L1
L1 = 0
n = n + 1
NIE
n > n3
TAK
n = 1
w = w + 1
L1= 0
NIE
w > 6
TAK
STOP

e/ Kryterium minimalizacji średniej liczby okienek w macierzach wykładowców

$$F_5 = \frac{1}{n_2} \sum_{r=1}^{m_2} L_r$$

f/ Kryterium maksymalizacji średniego współczynnika wykorzystania sal

$$F_6 = \frac{1}{m^o_3} \sum_{p=1}^{m^o_3} \frac{S'_p}{K} ,$$

gdzie:

S'_p - ilość godzin zajęć odbywających się w p-tej sali

K - dysponowany fundusz czasu sali

m^o_3 - ilość sal, dla których współczynnik wykorzystania sali

$$\frac{S'_p}{K} > 0 ,$$

g/ Kryterium minimalizacji ogólnej liczby niewypełnionych życzeń w grupach.

Dla wyliczenia wartości tego kryterium należy zachować w pamięci macierze czasowe grup z kroku zerowego tj. przed przystąpieniem do konstruowania rozwiązania.

Liczbę niewypełnionych życzeń w grupie, którą oznaczymy symbolem G_i, liczymy w sposób następujący:

Dla każdego elementu $c_{wn} > 1$ z wynikowej macierzy czasowej grupy w macierzy czasowej z kroku zerowego sprawdzamy czy element ten był równy 1; jeżeli tak, to oznacza, że przydzielono zgodnie z życzeniem, jeżeli natomiast elem ten równa się zero wtedy zwiększamy wartość G_i o jeden.

$$F_7 = \sum_{i=1}^{m_1} G_i$$

h/ Kryterium minimalizacji ogólnej liczby niewypełnionych życzeń wykładowców.

Podobnie jak w poprzednim kryterium w pamięci maszyny należy zachować macierze czasowe wykładowców z kroku zerowego. Algorytm liczenia liczby niewypełnionych życzeń wykładowcy W_r jest identyczny jak przy kryterium poprzednim:

$$F_8 = \sum_{r=1}^{m_2} W_r$$

Podane wyżej przykłady ośmiu kryteriów można traktować jako tzw. kryteria cząstkowe. Ocena przy pomocy jednego tylko kryterium jest najczęściej niepełna. Otrzymane rozwiązanie należałoby oceniać za pomocą kilku z podanych wyżej kryteriów. Wydaje się, że można wtedy stosować kryterium globalne F_g liczone w sposób następujący:

$$F_g = \sum_{m} \gamma_m F_m$$

gdzie:

F_m - kryterium cząstkowe

γ_m - współczynnik określający stopień ważności m-tego kryterium.

6.3.3. Kryteria istnienia rozwiązania dopuszczalnego

Zanim przejdziemy do szczegółowego omawiania algorytmu podamy kryteria jakie muszą być spełnione w każdym kroku konstruowania rozwiązania aby istniało rozwiązanie dopuszczalne. W omawianym tutaj zagadnieniu istnieje szereg ograniczeń /np. zadany przedział czasowy, w którym mogą się odbywać zajęcia, życzenia grup i wykładowców/, które powodują, że pewne przedmioty mogą się odbywać w ściśle określonych przedziałach czasowych $[t_j, t_{j+\tau_l-1}]$. Wymaga to badania w każdym kroku konstruowania rozwiązania czy dla przedmiotów niewprowadzonych do harmonogramu /tj. przedmiotów, którym nie przydzielono przedziału czasowego oraz sali, w której będzie się odbywał/ istnieje możliwość wprowadzenia ich do harmonogramu[1]. Na początku zdefiniujemy pojęcie "liczba dopuszczalnych położeń l-tego przedmiotu" pomocne przy formułowaniu kryteriów istnienia rozwiązania dopuszczalnego.

Dowolnemu przedmiotowi $d_l \in D$ można wyznaczyć tylko taki przedział czasowy, w którym zarówno grupa /lub grupy/ jak i wykładowca mogą odbywać zajęcie w tym przedziale oraz jest wolna w tym przedziale przynajmniej jedna sala. Liczebność zbiorów takich przedziałów czasowych będziemy nazywać *liczbą dopuszczalnych położeń przedmiotu* i oznaczać symbolem M_l. Liczba dopuszczalnych położeń przedmiotu $d_l \in D$ niewprowadzonego jeszcze do harmonogramu może ulegać zmianie w kolej-

---

[1] Podobne badania należałoby realizować w zagadnieniach rozważanych w rozdz. 4 oraz 5 gdyby był zadany przedział czasowy, w którym muszą być wykonywane operacje.

nych krokach konstruowania rozwiązania. Oznacza to konieczność korekty liczby M_1 obliczonej w kroku pierwszym. Sposób liczenia wielkości M_1 jest następujący:

1^o Identyfikujemy na podstawie wektora Z_1 wykładowcę oraz grupę, / lub grupy/, dla których prowadzony jest przedmiot d_1

2^o Wyliczamy macierz czasową

$$Y_1 = [y^1_{wn}] \qquad \begin{matrix} w = 1,2,\dots,6 \\ n = 1,2,\dots,n_5 \end{matrix} \tag{39}$$

w której:

$$y^1_{wn} = \begin{cases} 1 & \text{gdy } c_{wn} \text{ oraz } c'_{wn} \text{ równają się jedności} \\ 0 & \text{w przeciwnym przypadku} \end{cases}$$

gdzie:

c_{wn} - element macierzy czasowej grupy określonej w punkcie 1^o

c'_{wn} - element macierzy czasowej wykładowcy określonego w punkcie 1^o

3^o Gdy $\tau_1 = 1$ wtedy

$$M_1 = \sum_w \sum_n y^1_{wn} \tag{40}$$

przy czym sumujemy wszystkie takie elementy $y^1_{wn} = 1$, dla których istnieje przynajmniej jeden element $c'''_{wn} = 1$ w macierzy czasowej dowolnej sali ze zbioru sal należących do tego typu, w której powinien się odbywać przedmiot d_1.

4^0 Jeżeli natomiast $\tau_1 \geqslant 2$, wtedy w każdym wierszu macierzy czasowej Y_1 należy znaleźć wszystkie możliwe ciągi składające się z τ_1 kolejnych elementów przyjmujących wartości równe jedności. Następnie dla każdego takiego ciągu należy odszukać przynajmniej jeden ciąg analogicznych elementów w dowolnej macierzy czasowej sali należącej do tego typu sal, w której powinien odbywać się przedmiot d_1. Liczba tych ciągów w macierzy Y_1, które mają swoje odpowiedniki w macierzach sal będzie szukaną liczbą M_1. Początkowe elementy takich τ_1-elementowych ciągów w każdym wierszu macierzy Y_1 można wyznaczyć w sposób następujący:

$$\bar{y}^1_{wn} = 1 \quad \text{jeżeli} \quad y^1_{wn} = 1 \wedge y^1_{w,n+1} = 1, \wedge, \dots, \wedge y^1_{w,n+\tau_1-1} = 1 \tag{41}$$

dla każdego $w = 1,2,\dots,6$, $n = 1,2,\dots,n_5-\tau_1+1$.

Początkowym elementem ciągu będzie taki element y^1_{wn}, dla którego odpowiadający element $\bar{y}^1_{wn}$ w wyniku przekształcenia (41) przyjmie wartość jeden.

Zachodzi następująca równość:

$$M_1 = \sum_w \sum_n \bar{y}^1_{wn}$$

przy czym sumujemy tylko te elementy $\bar{y}^1_{wn} = 1$, dla których w macierzach czasowych sal k-tego typu /przyjmijmy, że przedmiot d_1 musi się odbywać w salach k-tego typu/ znajdziemy przynajmniej jeden ciąg $c''_{wn}, c''_{w,n+1}, \dots, c''_{w,n+\tau_1-1}$ o elementach równych jedności.

Przyjmiemy następujące oznaczenie: Y^1_j $(j=1,2,\dots,M_1)$ jest to ciąg $y^1_{wn}, y^1_{w,n+1},\dots,y^1_{w,n+\tau_1-1}$, dla którego w odpowiedniej macierzy czasowej sali elementy $c''_{wn}, c''_{w,n+1},\dots,c''_{w,n+\tau_1-1}$ równają się jedności.

W r-tym /r=1,2,.../ kroku konstruowania rozwiązania muszą być spełnione dwa warunki, aby w rezultacie można było otrzymać rozwiązanie dopuszczalne.

Warunek pierwszy mówi, że dla wszystkich przedmiotów d_1 nie wprowadzonych do harmonogramu w r-tym kroku $M_1 > 0$.
W celu sformułowania warunku drugiego przyjmijmy następujące oznaczenia:
Niech $D_r \subset D$ oznacza te spośród przedmiotów d_1, które są prowadzone przez r-tego wykładowcę, natomiast symbol D_1 analogicznie jak w § 6.1 oznacza te przedmioty d_1, które prowadzone są w 1-tej grupie. Symbolem $\bar{Y}^1$ oznaczymy ciąg utworzony w wyniku sumowania po "j" ciągów y^1_j. Zapis $\sum\limits_1 \bar{Y}^1$ z kolei oznaczać będzie sumowanie wartości elementów ciągu $\bar{Y}^1$

Warunek drugi można teraz sformułować następująco:
Dla każdej i-tej grupy, gdzie 1: $d_1 \in D_i$, j=1,2,...,M_1 oraz dla każdego r-tego wykładowcy, gdzie 1: $d_1 \in D_r$, j=1,2,...,M_1 musi być spełniona następująca równość:

$$\sum_1 \bar{Y}^1 - \sum_1 \tau_1 \geqslant 0 \tag{42}$$

Łatwo zauważyć, że podstawą do sformułowania tego warunku są wymagania stawiane każdemu harmonogramowi, które mówią, że każda grupa w każdym odcinku czasu może mieć tylko jedno zajęcie oraz, że każdy wykładowca w każdym odcinku czasu może mieć też tylko jedno zajęcie.

6.3.4. Prezentacja algorytmu

W algorytmie rozwiązania dwóch poprzednich zagadnień stosowane było postępowanie I, z kolei w algorytmie rozwiązania zagadnienia omawianego w tym rozdziale zastosujemy postępowanie I. Będzie to oznaczało, że w każdym kroku będziemy mieli do czynienia z konfliktem typu drugiego. Aby nie komplikować rozważań przyjmiemy, że przedmioty przyporządkowane każdej grupie można realizować w dowolnej kolejności. Wtedy zbiór przedmiotów gotowych do realizacji w r-tym kroku, oznaczymy go symbolem $Q(r)$, będą tworzyły przedmioty nie wprowadzone do harmonogramu.

Proces konstruowania rozwiązania przedstawimy w punktach.

1. Kładziemy $r = 1$. Zbiór $Q(1)$ w kroku pierwszym będzie się składał z elementów zbioru D.

2. Dla przedmiotów należących do zbioru $Q(r)$ sprawdzamy, czy spełnione są warunki na istnienie rozwiązania dopuszczalnego. Jeżeli jeden lub oba warunki nie są spełnione nie należy konstruować obliczenia, ponieważ w rozpatrywanym procesie konstruowania rozwiązania nie zostanie wygenerowane rozwiązanie dopuszczalne. W przeciwnym przypadku należy przejść do punktu następnego.

3. Wśród przedmiotów zbioru $Q(r)$ należy wybrać jeden przedmiot. Mamy wtedy do czynienia z konfliktem typu pierwszego. Sposoby likwidacji tego konfliktu podane będą poniżej.

4. Dla wybranego przedmiotu /niech to będzie l-ty przedmiot/ określamy ciągi $y^l_{wn}, y^l_{w,n+1}, \ldots, y^l_{w,n+\tau_l-1}$. Każdy taki ciąg oznaczaliśmy, jak wiadomo, symbolem Y^l_j /$j=1,2,\ldots,N_l$

Gdy $J_l > 1$ występuje konflikt typu drugiego. Sposoby likwidacji konfliktu typu drugiego podano również poniżej. W wyniku likwidacji konfliktu zostanie wybrany jeden ciąg. Określimy w ten sposób przedział czasu, w którym będzie się odbywał l-ty przedmiot.

5. Dla l-tego przedmiotu w typie sal, w którym musi się odbywać ten przedmiot przydzielamy tę wolną salę s_p /wtedy elementy macierzy czasowej sali $c''_{wn}, c''_{w,n+1}, \ldots, c''_{w,n+\tau_l-1}$ równają się jedności/, która ma najmniejszy wskaźnik p.

6. W ten sposób l-ty przedmiot został wprowadzony do harmonogramu. Fakt wprowadzenia do harmonogramu zostanie zapisany w ten sposób, że elementy macierzy czasowej grupy o wskaźnikach (w,n) ; $(w,n+1)$; ...;$(w,n+\tau_l-1)$ będą się równały numerowi wykładowcy, który prowadzi l-ty przedmiot, odpowiadające elementy macierzy czasowej wykładowcy - numerowi sali oraz odpowiadające elementy macierzy czasowej sali - numerowi przedmiotu.

   Ze zbioru $Q(r)$ "wykreślamy" wybrany przedmiot.

7. Jeżeli zbiór $Q(r)$ nie jest zbiorem pustym zwiększamy krok o jeden i przechodzimy do punktu 2 w przeciwnym przypadku harmonogram został ułożony.

Przy rozwiązywaniu realnych zadań bardzo często przyjmuje się, że życzenia grup i wykładowców nie muszą być bezwzględnie wypełnione. W takim przypadku gdy w r-tym kroku konstruowania rozwiązania okaże się, że dla pewnych przedmiotów ze zbioru $Q(r)$ nie są spełnione warunki na istnienie

'ozwiązania dopuszczalnego, wtedy można zaproponować nastę-
ıujący sposób postępowania.
'odzbiór takich przedmiotów należy wprowadzić do harmonogramu
ı kolejnych krokach r,r+1,r+2,... nie uwzględniając przy
;ym życzeń grup i wykładowców. Jeżeli mimo to dla pewnych
ırzedmiotów liczba dopuszczalnych położeń równa się zero
'warunek pierwszy/ wtedy oznacza to, że w procesie obliczeń
ıie zostanie wygenerowane rozwiązanie dopuszczalne.

6.3.5. Sposoby likwidacji konfliktów

Na początku podamy sposoby likwidacji konfliktu typu
ıierwszego, a następnie sposoby likwidacji konfliktu typu
lrugiego.
[ak wiadomo w przypadku konfliktu typu pierwszego ze zbioru
ǫ(r) należy wybrać jeden przedmiot. Wektor X, na którym
ıkreślamy funkcje priorytetu używane w poszczególnych sposo-
ıach likwidacji konfliktu typu pierwszego i zadawany dla
ıażdego elementu ze zbioru Q(r) będzie miał następujące
ıkładowe:

$$X = (x_1, x_2, x_3)$$

;dzie

x_1 - liczba dopuszczalnych położeń przedmiotu,

x_2 - ilość grup, dla których prowadzony jest przedmiot,

x_3 - czas trwania przedmiotu.

Oto przykłady funkcji priorytetu:

$1^o \quad w_1 = G_1(X) = \dfrac{x_1}{x_1 \cdot x_4}$

$2^o \quad w_2 = G_2(X) = x_1$

$3^o \quad w_3 = G_3(X) = x_2$

$4^o \quad w_4 = G_4(X) = x_3$.

Stosując powyższe funkcje priorytetu podamy następujące sposoby likwidacji konfliktu typu pierwszego.

Sposób A

a/ Wybieramy przedmiot, dla którego $w_2 = 1$,

b/ Gdy takich przedmiotów jest więcej jak jeden wybieramy ten przedmiot, dla którego funkcja $G_3(X)$ osiąga maksimum.

c/ W przypadku braku jednoznaczności wybieramy przedmiot, dla którego wartość funkcji $G_2(X)$ osiąga minimum.

d/ Przy dalszej niejednoznaczności wybieramy przedmiot losowo z jednakowym prawdopodobieństwem.

Sposób B

a/ Wybieramy przedmiot, dla którego funkcja $G_1(X)$ osiąga minimum.

b/ Gdy wystąpi niejednoznaczność wyboru, przedmiot wybieramy losowo z jednakowym prawdopodobieństwem.

Sposób C

a/ Ze zbioru $Q(r)$ wybieramy ten przedmiot, dla którego $w_2 = 1$.

b/ Przy braku jednoznaczności wybieramy kolejno: według maksymalnych wartości funkcji $G_3(X)$, według maksymalnych wartości funkcji $G_4(X)$, według minimalnych wartości funkcji $G_2(X)$, a następnie losowo z jednakowym prawdopodobieństwem.

Likwidacja konfliktu typu drugiego będzie polegała na wybraniu jednego elementu spośród $Y_1^l, Y_2^l, \ldots, Y_j^l, \ldots, Y_{M_l}^l$. Jak już zaznaczono, wybierając Y_j^l określamy w ten sposób przedział czasu, w którym będzie się odbywał l-ty przedmiot. Dla każdego elementu Y_j^l zadamy wektor X; na którym określamy funkcje priorytetu używane w podanych niżej sposobach likwidacji konfliktu typu drugiego. Wektor X' będzie się składał z następujących składowych

$$X' = (x_1', x_2', x_3')$$

gdzie

x_1' - liczba realnych sal, w których może się odbywać l-ty przedmiot, gdy wybrany zostanie element Y_j^l

x_2' - liczba "okienek" w macierzy czasowej, gdy dla l-tego przedmiotu zostanie wybrany element Y_j^l; gdy przedmiot jest prowadzony w kilku grupach wtedy x_2' oznacza średnią liczbę "okienek",

x_3' - liczba "okienek" w macierzy czasowej wykładowcy, gdy dla l-tego przedmiotu zostanie wybrany element Y_j^l.

Podamy teraz następujące funkcje priorytetu:

$$w_1' = G_1'(X') = \frac{x_2'}{x_1'}$$

$$w_2' = G_2'(X') = x_1'$$

$$w_3' = G_3'(X') = x_2'$$

$$w_4' = G_4'(X') = x_3' .$$

Sposoby likwidacji konfliktu typu drugiego będziemy oznaczali symbolami A' oraz B'.

Sposób A'

a/ Wybieramy taki element Y_j^1, dla którego funkcja $G_1'(X')$ osiąga minimum.

b/ W przypadku niejednoznaczności wybieramy według minimalnych wartości funkcji $G_4'(X')$ a następnie losowo.

Sposób B'

Element Y_j^1 wybieramy według minimalnych wartości funkcji $G_3'(X')$ a następnie, w przypadku braku jednoznaczności, wybieramy według minimalnych wartości funkcji $G_4'(X')$, maksymalnych wartości funkcji $G_2'(X')$ oraz losowo.

6.3.6. Opis eksperymentów komputerowych

Jak zaznaczono we wstępie pracy, autor starał się poznać istotę problemu programowania przedsięwzięć czasowych badając zadania występujące w realnej rzeczywistości.
Na podstawie uzyskanych doświadczeń przekonano się, że zaprojektowanie i oprogramowanie systemów dotyczących układania harmonogramów produkcji jest bardzo pracochłonne. Wymaga to przede wszystkim utworzenia w pamięci komputera tzw. bazy normatywnej. Poza tym ze względu na rozmiary zadania potrzebny jest silny zespół projektantów i programistów, którzy by zaprojektowali i oprogramowali system realizujący układanie harmonogramów. Z tych względów nie podjęto próby rozwiązania realnego zadania na maszynie cyfrowej. Dła weryfikacji algorytmu opisanego w rozdziale 4 przeprowadzono eksperyment w małej skali.
Przygotowano w tym celu program w języku ALGOL. Realizując ten program na komputerze Odra 1204 wyliczono szereg harmonogramów biorąc za podstawę dane liczbowe z pozycji [95], [164], [177].
W algorytmie na podstawie którego opracowano program przyjęto postępowanie I oraz zastosowano dwa sposoby likwidacji konfliktu typu pierwszego /por. [68]/.
Analizując otrzymane harmonogramy nie można generalnie stwierdzić że jeden ze sposbów likwidacji daje lepsze wyniki /jako kryterium oceny przyjęto funkcję F_1, czyli minimalizację czasu obróbki wszystkich detali/.
Próbę rozwiązania [illegible] go [illegible] na maszyni cyfrowej podjęto biorąc za podstawę zadanie układania harmonogramu zajęć [illegible] Ako-

demii Ekonomicznej we Wrocławiu. W tym celu zastosowano następujące postępowanie.

Po zapoznaniu się ze specyfiką układania harmonogramu zajęć w Akademii Ekonomicznej we Wrocławiu, przygotowano algorytm "ręcznego" układania harmonogramu zajęć i według tego algorytmu ułożono "ręcznie" realny harmonogram na semestr zimowy w roku akademickim 1973/74. Uzyskane w ten sposób doświadczenia oraz zasób wiedzy teoretycznej pozwoliły przygotować algorytm maszynowego układania harmonogramu zajęć.
W celu weryfikacji ważniejszych tez algorytmu przygotowano eksperyment w małej skali. Programiści z Ośrodka Obliczeniowego Instytutu Cybernetyki Ekonomicznej uruchomili zestaw programów w języku ALGOL na komputerze Odra 1204 [194].
Ułożenie realnego harmonogramu dla uczelni na komputerze Odra 1204 jest niemożliwe, ponieważ duża liczba danych wejściowych a także wyjściowych /ułożony harmonogram/ wymaga pamięci zewnętrznych o dużej pojemności oraz zastosowania urządzeń do wprowadzania danych z kart perforowanych.
Przy pomocy wspomnianego zestawu programów układano harmonogramy dla kilku grup, kilku grup wykładowców oraz sal i kilkunastu przedmiotów. Ponieważ eksperyment wykazał, że przy pomocy zaproponowanego algorytmu możliwe jest generowanie rozwiązań dopuszczalnych przystąpiono do prac nad realnym harmonogramem.

Przyjęto następujące ustalenia:

a/ Harmonogram zajęć będzie układany dla studiów dziennych.
W salach, w których odbywają się zajęcia ze studentami
studiów zaocznych odpowiednie godziny będą niedostępne

/inaczej zablokowane/ przy układaniu harmonogramu przez maszynę /w pamięci maszyny odpowiednie elementy macierzy czasowych sal będą się wtedy równały zeru/.

b/ Przy układaniu harmonogramu nie będą uwzględnione informacje o tym, który wykładowca prowadzi dany przedmiot, ponieważ bardzo często jeszcze na początku semestru następują zmiany w obsadzie poszczególnych przedmiotów. Przy układaniu harmonogramu "ręcznie" przez pracowników Działu Nauczania również nie uwzględnia się tych informacji.

c/ Przy pomocy "maszynowego" układania harmonogramu nie będą wprowadzane do harmonogramu pewne specyficzne przedmioty. Dotyczy to seminariów, laboratoriów oraz języka rosyjskiego. Aby przedmioty te mogły być wprowadzane przez maszynę potrzebne są zmiany w trybie odbywania się przedmiotów.
Seminaria lub laboratoria prowadzone są w kilku lub kilkunastu osobowych grupach w skład których wchodzą studenci z różnych grup studenckich danego roku przy czym zajęcia te odbywają się najczęściej w gabinecie wykładowcy. Według obowiązującej na Uczelni praktyki przedmiotom tym przydzielane godziny, w których studenci nie odbywają zajęć z innych przedmiotów.
Zajęcia z języka rosyjskiego są wprowadzane do harmonogramu w pewien specyficzny sposób. Łączy się 3 lub 4 grupy studenckie w tzw. b l o k /brak jednoznacznego kryterium łączenia grup/ i dla bloku przydziela się 5-6 sal audytoryjnych.
W Studium Języków Obcych dokonuje się przydziału studentów z bloku do poszczególnych sal. Aby utrzymać taką zasadę

wprowadzenia do harmonogramu tego przedmiotu, należałoby znacznie rozszerzyć i skomplikować program.

W rozdziale 6 został opisany algorytm układania harmonogramu zajęć. Algorytm ten należy traktować jako pewnego rodzaju postępowanie wzorcowe. Biorąc pod uwagę opisane wyżej ustalenia a także mając na uwadze zmniejszanie czasu obliczeń na maszynie, zarówno przed przystąpieniem do pisania programów jak i w trakcie uruchamiania programów na danych próbnych dokonano modyfikacji algorytmu, nie zmieniając jego zasadniczych tez.

Aby zmniejszyć czas obliczeń na komputerze w algorytmie przyjęto że jako pierwsze zostaną wprowadzone przedmioty prowadzone dla więcej jak jednej grupy /wykłady/ a następnie przedmioty prowadzone dla jednej grupy /ćwiczenia/. Przy wybieraniu przedmiotów nie brano pod uwagę liczby dopuszczalnych położeń.

W algorytmie uwzględniono ograniczenia brane pod uwagę przy ręcznym układaniu harmonogramu, takie jak:

- liczba wykładów w jednym dniu nie powinna przekraczać trzech,
- ilość godzin trwania zajęć w grupie w jednym dniu nie powinna przekraczać ośmiu,
- w sobotę zajęcia nie powinny się w zasadzie odbywać.

W celu realizacji algorytmu na maszynie Odra 1305 /z pamięcią zewnętrzną na taśmach magnetycznych/, autor pracy przygotował dwa programy w języku FORTRAN /program I zał.1 oraz program II zał.2/.

Za pomocą programu I układane są dwa warianty harmonogramu. Algorytm generowania pierwszego wariantu tym różni się od algorytmu generowania drugiego wariantu, że przy ustalaniu przedział czasu, w którym będzie się odbywał przedmiot, w pierwszym wariancie preferowano początkowe dni tygodnia, natomiast w drugim najwcześniejsze godziny.

Za pomocą programu II odbywa się przyporządkowanie zakodowanym informacjom odpowiednich tekstów i drukowanie ułożonych harmonogramów.

Dla zapisania tzw. danych wejściowych zaprojektowano szereg dokumentów wejściowych[1]. Dane wejściowe w programie I zapisane są w następujących dokumentach /patrz Aneks w pracy/:

- karta kierunków /zał.3/,
- karta grup /zał.4/,
- karta sali /zał.5/,
- karta przedmiotu /zał.6/.

Dane zawarte w tych dokumentach przenoszone są do pamięci maszyny za pomocą kart perforowanych. Na podstawie danych wejściowych, przy pomocy programu I, tworzone są macierze czasowe grup i sal, zapisywane następnie w pamięci taśmowej. Jak już zaznaczono przy opisie algorytmu, w macierzach tych zapisywane są także dane o ułożonym harmonogramie.

Dane wejściowe dla programu II zawarte są na taśmie magnetycznej /ułożony harmonogram/ oraz w następujących dokumentach/patrz załączniki/:

- kody przedmiotów /zał.7/,
- kody sal /zał.8/,
- kody kierunków /zał.9/.

---

[1] Dokumenty wejściowe i wynikowe /tabulogramy/ od strony graficznej zaprojektował J.Korczak.

Analogicznie jak w programie I dane z trzech rodzajów dokumentów wczytywane są do pamięci z kart perforowanych.
W wyniku działania programu II drukowane są dwa rodzaje dokumentów wynikowych
- plan zajęć dydaktycznych dla grup,
- wykaz obciążeń sal dydaktycznych.
Dokumenty wynikowe zaprojektowano zgodnie z życzeniami przyszłych użytkowników - pracowników Działu Nauczania.
W tabulogramie pn. wykaz obciążeń sal dydaktycznych drkowane są tzw. k a r t e c z k i. Karteczki te będą wkładane do specjalnej tablicy istniejącej w Dziale Nauczania.

Oba programy były testowane na realnych danych z Akademii Ekonomicznej[1]. Rozpatrywanych było 91 grup studenckich, 48 sal oraz 412 przedmiotów. W celu przeniesienia danych wejściowych na karty perforowane należało wydziurkować około 1500 kart. Czas generowania jednego wariantu harmonogramu zajęć na semestr zimowy roku 1974/75 wynosi około 2 godzin pracy komputera Odra 1305.
Biorąc pod uwagę kryterium minimalizacji ogólnej liczby okienek w macierzach czasowych grup /por. §6.3.2/ lepszym jest I wariant harmonogramu.
Ponieważ pełny wydruk jednego wariantu dla rozpatrywanego przykładu zajmuje 170 stron na papierze drukarki wierszowej, zrezygnowano z załączenia do pracy pełnych wydruków obu wariantów harmonogramu.

aneksie do pracy znajdują się natomiast takie same fragmenty wydruku wariantu pierwszego oraz wariantu drugiego.

---
[1] Dużą pomoc zebraniu ogromnej ilości danych okazała J.Stando-[illegible].

I tak załączniki Nr 10 oraz Nr 11 stanowią fragmenty dokumentów wynikowych z pierwszego wariantu, z kolei załączniki Nr 12 oraz Nr 13 z drugiego wariantu.

W związku z prowadzonym na realnych danych eksperymentem można sformułować następujące wnioski:

a/ Programy realizujące układanie harmonogramu zajęć powinny wykorzystywać dyski jako pamięć zewnętrzną.
W programie I, przy kwalifikowaniu przedmiotów, próbowano uwzględnić ilość dopuszczalnych położeń przedmiotu. Generowanie harmonogramu dla przykładu zawierającego 1/3 rozpatrywanych /por.str.172/ przedmiotów trwało 3,5 godziny.

b/ W pierwszym etapie prac nad systemem komputerowego układania harmonogramu zajęć nie należy rozpatrywać wykładowców. Zakres systemu powinien być rozszerzony po wdrożeniu i pewnym okresie eksploatacji.

c/ Do opracowania systemu należałoby utworzyć kilkuosobowy zespół składający się z 2-3 wysokokwalifikowanych programistów oraz pracowników Działu Nauczania. Na wykonanie tej pracy zespół potrzebowałby około jednego roku czasu.

d/ Uruchomienie programów wymagałoby około 50 godz. pracy komputera.

ZAKOŃCZENIE

Wnioski dotyczące problematyki omówionej w niniejszej pracy przedstawimy w następujących punktach:

1. Podjęta w pracy próba sformułowania przedmiotu teorii przedsięwzięć czasowych i zadań rozpatrywanych w tej teorii wskazuje, że szereg zadań, rozpatrywanych w literaturze oddzielnie można zaliczyć do tej klasy zadań. Niejednokrotnie akcentowane w pracy trudności natury numerycznej w stosunku do metod dokładnych powodują, że nawet do rozwiązania zadań o średnich rozmiarach trzeba stosować metody przybliżone /metody Monte-Carlo, metody heurystyczne/. Rozwiązanie zadania o dużych rozmiarach można uzyskać jedynie za pomocą metody heurystycznej.

2. Do rozwiązywania szerokiej klasy zadań o dużych rozmiarach w teorii przedsięwzięć czasowych zaproponowano pewien schemat postępowania zwany metodą priorytetów. U podstaw tej metody leżą zasady jakimi kieruje się człowiek rozwiązując w praktyce tego typu zadania.

3. Większość zadań rozważanych w teorii przedsięwzięć czasowych posiada bieżący, operatywny charakter. Szybkie otrzymanie rozwiązania na maszynie ma wtedy duże znaczenie. Metoda priorytetów spełnia ten postulat.

4. Do rozwiązania każdego konkretnego zagadnienia za pomocą metody priorytetów należy podchodzić indywidualnie wykorzystując jego cechy charakterystyczne. Dotyczy to w szczególności sposobu likwidacji konfliktów.

5. W opisanym w pracy schemacie postępowania zadawany jest pewien sposób likwidacji konfliktów, według którego likwidowany jest konflikt w każdym kroku konstruowania rozwiązania. Powoduje,to, że algorytm konstruowania rozwiązania jest prosty i jak pokazały eksperymenty na komputerze Odra 1305, przy pomocy tej metody można wygenerować rozwiązanie w sensownym okresie czasu.
W związku z coraz doskonalszymi pamięciami zewnętrznymi, które także u nas w kraju pojawią się zapewne masowo w niedalekiej przyszłości, należałoby opisany schemat postępowania doskonalić.
Wydaje się, że można by wtedy stosować następujące postępowanie: najlepiej realizować każdy krok konstruowania rozwiązania tj. w każdym r-tym kroku wśród m możliwych sposobów likwidacji konfliktu wybrać taki, że

$$F_{opt}(r) = \min_{m} F_m(r)$$

gdzie F - funkcja-kryterium.

Jest rzeczą oczywistą, że optymalna realizacja każdego kroku konstruowania rozwiązania nie gwarantuje uzyskania optymalnego rozwiązania. Wydaje się jednak, że otrzymane w ten sposób rozwiązanie będzie lepsze od rozwiązania uzyskanego przy pomocy arbitralnego sposobu likwidacji konfliktów.

Autor pracy w dalszych dociekaniach naukowych chciałby się zająć problemami określenia metod, za pomocą których w każdym kroku konstruowania rozwiązania wybierany jest wariant optymalny.

Pierwsze próby zastosowania w tym celu metod programowania matematycznego nie przyniosły pozytywnych rezultatów, ponieważ w każdym kroku otrzymuje się zadania o dużej ilości zmiennych i ograniczeń. Autor sądzi jednak, że metody w których łączone są podejścia heurystyczne i podejścia stosowane w metodach dokładnych mają dużą perspektywę.

6. Problematyce rozważanej w niniejszej pracy poświęcona jest bogata literatura fachowa. Świadczy o tym zamieszczony w końcu pracy spis literatury. W związku z niewielką ilością opublikowanych pozycji w języku polskim, korzystano głównie ze źródeł radzieckich, ponieważ organizacja procesu produkcji oraz organizacja procesu dydaktycznego jest identyczna

zarówno u nas w kraju jak i w ZSRR.

7. Ważnym zagadnieniem jest forma prezentacji algorytmów proponowanych w niniejszej pracy. Rozpatrywane algorytmy zapisane w języku FORTRAN zamieszczone są w Aneksie. Do prezentacji algorytmów stosowano werbalny zapis gdyż, zapis w języku algorytmicznym mimo swej ścisłości jest mało czytelny, natomiast schematy blokowe i tablice decyzyjne w przypadku rozpatrywanych w pracy algorytmów zawierających dużo predykatów są szczegółowe i również mało czytelne.

8. Wykorzystanie w praktyce zaproponowanych algorytmów wymaga zaangażowania dużego zespołu projektantów i programistów, zapewnienia środków finansowych oraz sprzętu z tzw. pamięciami o dostępie bezpośrednim.

L I T E R A T U R A

[1] Adamiecki K., Harmonizacja pracy, Warszawa 1948

[2] Adamiecki K., O nauce organizacji, Pisma wybrane, Warszawa 1970

[3] Afonin L.A. i inni, Imitacja raboty stroitielnoj dla sostawlenja kalendarnogo raspisanija, W: Waprosy kibernetiki - situacjonnoe uprawlenije teoria i praktika, Moskwa 1974

[4] Aleksandrow J.A., Ewristiczeskoje programmirowanje kak instrumient issledowanija w słożnych sredach, W: Statistika i elektronno - wyczislitelnaja tiechnika w ekonomikie cz.III, Moskwa 1970

[5] Aleksandrow J.A., O niekotorych predposyłkach ewristiczeskogo programmirowanija, W: Statistika i elektronno-wyczislitelnaja tiechnika w ekonomikie cz.II, Moskwa 1968

[6] Aleksandrow J.A., Osnowy tieorii ewristiczeskich rieszenij, Moskwa 1975

[7] Alfierowa Z.W., Matematiczeskoje obezpieczenije ekonomiczeskich rasczetow s ispolzowanijem tieorii grafow, Moskwa 1974

[8] Alfierowa E.A., Jeżowa W.P., Primienienie tieorii grafow w ekonomiczeskich issledowanijach, Moskwa 1971

[9] Almond M., An Algoritm for Constructing the University Timetables, "The Computer Journal", 1966, nr 4

[10] Apter M., Komputery a psychika. Symulacja zachowania, Warszawa 1973

[11] Aronowicz A.B., Ob odnoj zadacze kalendarnogo płanirowanije, "Ekonomika i Matematiczeskije Mietody", 1968, nr 3

[12] Aronowicz A.B., O wybore optimalnych kombinacji łokalnych prawił kalendarnogo płanirowanija, "Ekonomika i matematiczeskije mietody", 1970, nr 1

[13] Automatyczne przetwarzanie informacji, praca zbiorowa pod red. Z. Hellwiga, Warszawa 1973

[14] Awtomatizirowannaja sistiema uprawlenija, praca zbiorowa pod red. O.W. Kozłowej, tom 1 i 2, Moskwa 1972

[15] Badania operacyjne w nowoczesnym zarządzaniu, praca zbiorowa pod red. T.Kasprzaka, Warszawa 1974

[16] Balut S.J., Scheduling to Minimize the Number of Late Jobs when Set-up and Processing Times are Uncertain, "Management Since", 1973, nr 11

[17] Baraaclough E.D., The Application of a Digitial Computer to the Construction of Timetables, "The Computer Journal", 1965, nr 2

[18] Baran-Jarosz B. i inni, Wybrane metody rozwiązywania problemów uszeregowania prac na maszynach, Matematyka Stosowana, 1974, nr 2

[19] Bartosiewicz S., O technice stosowania metody PERT, "Przegląd Statystyczny" 1966, nr 1

[20] Bartosiewicz S., Porządkowanie węzłów sieci PERT, "Przegląd Statystyczny", 1966, nr 3

[21] Bellman R., Dinamiczeskoje programmirowanije, Moskwa 1960

[22] Bellman R., Mathematical Aspects of Scheduling Theory, "J. Soc. Indust. Appl. Math.", 1966, nr 3

[23] Bellman R., Dreyfus S., Programowanie dynamiczne, Warszawa 1967

[24] Bergman W.I., Grafy w zadaczach uprawlenija proizwodstwom, Moskwa 1974

[25] Berus W., Organizacja pracy komórek planowo-rozdzielczych Warszawa 1970

[26] Biegel J., Uprawlenje proizwodstwom. Koliczestwiennyj podchod, Moskwa 1973

[27] Biek N.N., Golenko D.I., Statisticzeskije mietody optymalizacji w ekonomiczeskich issledowanijach, Moskwa 1971

[28] Biełkina M.W., Gromowa W.I., Optimalizacja posledowatielnosti wypołnienia opieracji /przegląd/, "Awtomatika i Tielemiechanika" 1969, nr 3

[29] Biełow I.S., Stolin J.N., Ałgoritm w odnomarszrutnoj zadacze kalendarnogo płanirowanija, W: Matematiczeskaja ekonomika i funkcjonalnyj analiz, Moskwa 1974

[30] Biezzubow J.I., Optimizacja wyczislitelnogo processa w CWM pri determinirowannom potokie zajawok, "Techniczeskaja kibernetika" 1975, nr 3

[31] Brown P.G., Lomnicki Z.A., Some applications of the "Branch-and Bound" Algorithm to the Maschine Scheduling Problem", Operational Research Quarterly", 1966, nr 2

[32] Bukietyński W., Problemy optymalizacji dyskretnej / w druku/

[33] Burdiuk W.J., O zadacze m stankow $(m > 2)$, "Kibiernetika" 1969, nr 3

[34] Burdiuk W.J., Odna ekstremalnaja zadacza tieorii raspisanji, "Kibiernetika", 1973, nr 3

[35] Burdiuk W.J., Szkurba W.W., Tieoria raspisanji. Zadaczi i mietody reszenji, "Kibiernetika", 1971, nr 1

[36] Burkow W.N., Łowieckij S.J., Mietody reszenija ekstremalnych zadacz kombinatornogo tipa, "Awtomatika i Tielemiechanika", 1968, nr 11

[37] Burlaga H., Uogólnienie problemu przydziału, "Biuletyn Wojskowy Akademii Technicznej", 1972, nr 10

[38] Burstall R.M., A Heuristic Method for a Job-Scheduling Problem, "Operational Research Quartely" 1966, nr 3

[39] Busacker R., Saaty T., Konieczny je grafy i sieci, Moskwa 1974

[40] Całłagowa O.N., Ob odnoj zadacze sostawlenija raspisanija, W: Matiematiczeskije mietody reszenija ekonomiczeskich zadacz nr 5, Moskwa 1974

[41] Conway R., Maxwell W., Miller L., Theory of Scheduling, New York 1967

[42] Cybernetyka zarządzania w systemach ekonomicznych, praca zbiorowa pod red. T.Kasprzaka, Warszawa 1972

[43] Czebotarew A.S., Posledowatelnaja optimizacja w odnoj zadacze kalendarnogo płanirowanija, "Techniczeskaja Kibernetika", 1975, nr 1

[44] Czerniawskij A.L., Ałgoritmy dla reszenija kombinatornych zadacz, osnowannyje na mietodzie niejawsnogo pierebora, "Awtomatika i Tielemiechanika", 1972, nr 2 oraz 1972 nr 3

[45] Czerniawskij A.L., Ewristiczeskaja programma dla sostawlenja grafika dwiżenija pojezdow, "Awtomatika i Tielemiechanika" 1971 nr 1 oraz 1971 nr 3

[46] Czernowa G.W., Zadacza opredelenija posledowatelnosti obrabotki detalej imiejuszczich odinakowyje technołogiczeskije marszruty, "Primienienie Matematiki w Ekonomiki" 1967, nr 4

[47] Dumler S.A., Uprawlenje proizwodstwom i kibiernietika, Moskwa 1969

[48] Ekonomiko-matematiczeskije modeli, praca zbiorowa pod red. N.P.Fiedorenko, Moskwa 1969

[49] Elementy programowania nieliniowego i dynamicznego, praca zbiorowa pod red. W.Bukietyńskiego, Wrocław 1974

[50] Elementy rachunku ękonomicznego, praca zbiorowa pod red. Z.Hellwiga, Warszawa 1972

[51] Evans G.W., Wallace G.F., Sutherland G.L., Symulacja na maszynach cyfrowych, Warszawa 1973

[52] Filar W., Badania operacyjne a problemy zaopatrzenia i obsługi wojsk, Warszawa 1973

[53] Flachsmeyer J., Kombinatoryka, Warszawa 1974

[54] Franczuk W.J., Niekotoryje ałgoritmy reszenija zadacz kalendarnogo płanirowanija, W: Matiematiczeskije mietody, reszenija ekonomiczeskich zadacz, nr 5, Moskwa 1974

[55] Fridman A.A., Botjakow A.A., Diskretnyje zadaczi i mietody wietwiej i granic, "Ekonomika i Matiematiczeskije Mietody, 1974, nr 3

[56] Giedymin O., Metody optymalizacji w planowaniu sieciowym, Warszawa 1974

[57] Gieorgobiani D.A., Ob odnoj zadacze sostawlenija raspisanji, W: Woprosy optimalnogo płanirowanija, Tbilisi 1970

[58] Gierasimow N.J., Płanirowanije proizwodstwiennoj programmy maszinostroitelnogo predprijatja, Moskwa 1971

[59] Glebow N.J., Pierepelnica W.A., O niżniej i wierchniej ocenkach dla odnoj zadaczi tieorii raspisanji, W: Issledowania po kibiernietike, red. Liapunow A.A., Moskwa 1970

[60] Głuszkow W.M. i inni, Człowiek i wyczislitielnaja tiechnika, Kijów 1971

[61] Głuszkow W.M. Wwiedienije w ASU, Kijów 1974

[62] Golenko D.I., Statisticzeskije mietody sietiewogo płanirowanija i uprawlenija, Moskwa 1968

[63] Golenko D.I., Statisticzeskije mietody w ekonomiczeskich sistiemach, Moskwa 1970

[64] Golenko D.I., Statisticzeskije modeli w uprawlenii proizwodstwom, Moskwa 1973

[65] Gordjiczuk A.R., Podczasowa T.P., Postrojenije kalendarnogo płan-grafika raboty cecha /uczastka/, "Miechanizacja i awtomatizacja uprawlenija", 1969 nr 3

[66] Gorfan K.L., Russman L.B., Niekotoryje priłożenija

zadaczi o rasstanowkie prioriteta, "Ekonomika i Matiematiczeskije Mietody", 1969, nr 3

[67] Gospodarowicz A., Korczak J., Komputerowy system planowania i kontroli produkcji, W: Zarys systemów informatycznych w przedsiębiorstwie przemysłowym, praca zbiorowa pod red. E.Niedzielskiej, Wrocław 1973

[68] Gospodarowicz A., Wyznaczanie harmonogramu wykonywania operacji za pomocą metody przybliżonej, "Przegląd Statystyczny", 1975, nr 1

[69] Gospodarowicz A., Zagadnienie układania harmonogramu zajęć przy pomocy komputera, Prace Naukowe WSE we Wrocławiu 1974, nr 54

[70] Gospodarowicz A., Zastosowanie pewnej metody heurystycznej do rozwiązywania zagadnień z teorii przedsięwzięć czasowych, Prace Naukowe AE we Wrocławiu 1974 nr 62

[71] Grabowski A., Planowanie produkcji detali wytwarzanych systemem partiowym w warunkach przeciążenia maszyn, "Przegląd Statystyczny" 1970, nr 3-4

[72] Grinieva S.N., Kalendarnoje płanirowje sborocznych rabot w miełkosierijnom i jedinicznom proizwodstwie, "Ekonomika i Matiematiczeskije Mietody", 1970 nr 4

[73] Grudzewski W.M., Kierunki optymalizacji planów sieciowych, "Problemy Organizacji", 1971, nr 1

[74] Grudzewski W., Szamkołowicz L., Zastosowanie teorii grafów i metod sieciowych w planowaniu przedsięwzięć organizacyjno-technicznych, Wrocław 1974

[75] Haus B., Formy organizacji pracy w przemyśle, Warszawa 1964

[76] Hellwig Z., Elementy rachunku prawdopodobieństwa i statystyki matematycznej, Warszawa 1974

[77] Hellwig Z., Statystyczne modele niekonfliktowych systemów organizacyjnych, "Przegląd Statystyczny", 1963, 1

[78] Hellwig Z., Szwarc W., O tak zwanym problemie komiwojażera, "Przegląd Statystyczny", 1960 nr 2

[79] Hanusz T., Planowanie wykonawcze produkcji w przedsiębiorstwie przemysłowym, Warszawa 1972

[80] Harary F., Tieoria grafów, Moskwa 1973

[81] Idźkiewicz A. Pert. Metody analizy sieciowej, Warszawa 1967

[82] Ignasiak E., Programowanie sieciowe, Warszawa 1972

[83] Igudin R.W., Zadaczi teorii raspisanij na transportie i ałgoritmy ich reszenija, "Ekonomika i matematiczeskije mietody", 1975, nr 3

[84] Injutina K.W., Kurowskij W.N., Modeli zadacz płanirowanija proizwodstwa i materialno-techniczeskogo obespeczenija w ASUP, Moskwa 1975

[85] Ioffe E.G., Ałgoritm dla opredelenija wsiech optimalnych raspisanij w dwuchopieracionnoj zadacze, "Awtomatika i Tielemiechanika", 1973 nr 7

[86] Ioffe E.G., Ob odnom ałgoritmie rasczota kalendarnych raspisanji dla proizwodstwiennogo uczastka, "Ekonomika i Matiematiczeskije Mietody", 1971, nr 6

[87] Issledowanija po diskretnoj matiematikie, red. S.A.Fridman, Moskwa 1973

[88] Jakowlew J.I., Maszinnaja imitacja, Moskwa 1975

[89] Jakowlew J.I., Maszinnoje modelirowanije, "Ekonomika i Matematiczeskije Mietody", 1968, nr 2

[90] Jankowska-Zorychta Z., Modele sekwencyjne i ich zastosowanie w planowaniu optymalnej organizacji w dyskretnych procesach produkcyjnych, Warszawa 1973

[91] Jankowska-Zorychta Z., Modele uszeregowania prac na maszynach, "Przegląd Statystyczny", 1973, nr 1

[92] Jędrzejowicz P., Rozwiązywanie sytuacji decyzyjnych przedsiębiorstwa żeglugowego za pomoc metod teorii gier i programowania heurystycznego, praca habilitacyjna, Gdynia 1973

[93] Kabysz W.L., Bernsztein A.S., Elwow S.N., Matematiczeskije mietody reszenija ekonomiczeskich zadacz w otrasiach narodnogo chozjajstwa, Kijów 1973

[94] Kagan B.M., Czurcin W.N., Algoritm sostawlenia raspisanija uczebnych zanjatii w sistemie płanirowanija uczebnogo procesa ASU-MIIT, Prace Naukowe Moskiewskiego Instytutu Inżynierów Kolejnictwa /MIIT/ nr 395, Moskwa 1971

[95] Kalendarnoje płanirowanie, red. I.F.Muth, G.L.Thompson, Moskwa 1966

[96] Kantorowicz L.W., Optimalnoje płanirowanije i niereszeszennyje zadaczi, "Ekonomika i organizacja promyszlennogo proizwodstwa", 1975, nr 5

[97] Kantorowicz L.W., Gorstko A.B., Optimalnyje reszenija w ekonomikie, Moskwa 1972

[98] Karagodova E.A., Masliuk G.F., Ob odnoj postanowkie zadaczi kalendarnogo płanirowanija, "Mechanizacja Uczeta i Wyczislitelnych Rabot", 1967, nr 6

[99] Kasprzak T., Funkcje kryterium opisane na porządkowanych polach preferencji, Warszawa 1968

[100] Kemeny J., Snell J., Kibierneticzeskoje modelirowanije, Moskwa 1972

[101] Kiselew J.W., Lisicyn W.A., Woprosy postrojenija modelej predpocztienija, "Techniczeskaja Kibernetika", 1974, nr 5

[102] Kładow T.K., Liwszic E.M., O zadacze minimalizacji sztrafow pri sostawlenji raspisanija, "Kibiernetika", 1968, nr 6

[103] Kolny J., Uwagi o algorytmie wykonywania rozkładu zajęć przez elektroniczną maszynę cyfrową /EMC/, "Dydaktyka Szkoły Wyższej", 1972, nr 3

[104] Korbut A.A., Finkelsztejn J.J., Programowanie dyskretne, Warszawa 1974

[105] Kowalski K., Algorytm układania harmonogramu zajęć dla

wyższej uczelni, Prace Naukowe Instytutu Cybernetyki Politechniki Wrocławskiej, nr 7, Wrocław 1972

[106] Kozyriew W.P., O predstawlenij grafow siecjami, W: Woprosy Kibernetiki - trudy seminara po kombinatornoj matematikie, Moskwa 1973

[107] Kryński H., Zastosowanie matematyki w ekonomii, Warszawa 1973

[108] Kreczmar A., Kryterium istnienia i algorytm układania rozkładu zajęć szkolnych, Uniwersytet Warszawski, zeszyt 23, Warszawa 1970

[109] Krzysztofiak U., Główne relacje produkcyjne polskiego przemysłu okrętowego, Gdańsk 1969

[110] Kubale M., Problemy komputerowego układania rozkładów zajęć dla szkół wyższych, Algorytmy, Warszawa 1973

[111] Kuksa A.I., Raspisanija na grafach i niezawisimyje množestwa grafow, "Kibernetika", 1970, nr 4

[112] Kuratowski K., Wstęp do teorii mnogości i topologii, Warszawa 1962

[113] Kuzin B.I., Algoritm postrojenija optimalnogo kalendarnogo grafika nieprerywnoj raboty oborudowanija w usłowiach paralelno-posledowatelnogo wida dwiżenija proizwodstwa, "Primienieije Matematiki w Ekonomike", Leningrad 1970, nr 6

[114] Kuzin L.T., Osnowy kibiernetiki, Moskwa 1973

[115] Lawer E.L., Wood D.E., Branch-and-Bound Methods. A Survey, "Operations Research", 1966, nr 4

[116] Lawire N.L., An Integer Linear Programming Model of a School Timetabling Problem, "The Computer Journal", 1969, nr 12

[117] Liewner J.W., Optimalnoje płanirowanje obrabotki obiektow na rjadie maszin, "Awtomatika i Tielemechanika", 1969, nr 12

[118] Lindsay P., Norman D., Piererabotka informacji u czełowieka, Moskwa 1974

[119] Litwin U.O., Samonienko W.O., Reszenije odnoj zadaczi sostawlenija uczebnogo raspisanija na CWM Mir, "Mechanizacja Uczeta i Wyczislitielnych Rabot", 1971, nr 12

[120] Liwszic E.M., Issledowanije niekotorych ałgoritmow sietiewych modelej, "Ekonomika i Matematiczeskije Mietody", 1968, nr 5

[121] Łapin A.N., Tri ewristiczeskije schiemy predstawlenija funkcji mnogich pieremiennych, W: Waprosy kibernetiki - adaptiwnyje sistiemy, Moskwa 1974

[122] Łytkin J.P., Ob obrabotnoj zadacze kalendarnogo płanirowanija, W: Matiematiczeskije mietody reszenija ekonomiczeskich zadacz, nr 5, Moskwa 1974

[123] Mały słownik cybernetyczny, red. M.Kempisty, Warszawa 1973

[124] Maszyny matematyczne i myślenie, praca zbiorowa pod red. E.Fiegenbauma, J.Feldmana, Warszawa 1972

[125] Matiematika i kibiernetika w ekonomikie, słowar - sprawocznik, Moskwa 1975

[126] Matiematiczeskije mietody w płanirowanii otraslej i predprijatij, praca zbiorowa pod red. I.G.Popowa, Moskwa 1973

[127] McDonough A.M., Systemy scentralizowane, Planowanie i kontrola, Warszawa 1973

[128] Metody heurezy, referaty na I krajowe sympozjum, Warszawa 1974

[129] Metody heurezy, referaty na II krajowe sympozjum, Warszawa 1975

[130] Michiejew I.A., Ob odnoj zadaczy postrojenja raspisanja, "Kibiernetika", 1970, nr 3

[131] Miezwirszwili T.E., Rasczet kalendarnych planow zagruszki oborudowanija w sieryjnom proizwodstwie, "Mechanizacja Uczeta i Wyczislitelnych Rabot", 1972, nr 15

[132] Mironosieckij N.B., Ekonomiko-matematiczeskije metody kalendarnogo płanirowanija, Nowosybirsk 1973

[133] Mirosznniczenko G.P., K waprosu o sposobie postrojenija sietiewogo grafika, "Ekonomika i Matematiczeskie Mietody", 1970, nr 3

[134] Moisiejew N.N., Czislennyje mietody w teorii optimalnych sistiem, Moskwa 1971

[135] Mołczan W.A., Kozyriew A.W., O podgatowkie ischodnoj informacji dla optimizacji uczebnych płanow wuza s pomoszczju EWM "Minsk 22", "Maszinnaja obrabotka informacji", 1973, nr 16

[136] Mołczan W.A., Szelest W.A., Astrachow W.I., Niekotoryje woprosy awtomatizacji sostawlenija raspisanija uczebnych zaniatii wuza s pomoszczju EWM "Minsk 22", "Miechanizacja Uczeta i Wyczislitelnych Rabot", 1971, nr 13

[137] Nykowski I., Planowanie produkcji detali jako uogólnione zadanie transportowe, "Przegląd Statystyczny", 1970, nr 3-4

[138] Organizacja i planowanie w przedsiębiorstwie przemysłowym, praca zbiorowa pod red. A.Grossmana, Warszawa-Wrocław 1972

[139] Oskołowa S.J., Oskołow I.O., Primienienije niektorych ewristiczeskich mietodow k reszeniju zadacz kalendarnogo płanirowanija /przegląd/ "Awtomatika i Tielemechanika", 1968, nr 2

[140] Ostasiewicz W., Zastosowanie kombinatoryki do rozwiązywania zagadnień ekstremalnych, praca doktorska, WSE Wrocław 1973

[141] Oziernoj W.M., Riabow L.P., Ewristiczeskij mietod optimizacji posledowatielnosti wypołnienija opieracji, "Awtomatika i Tielemiechanika", 1967, nr 12

[142] Pawlak Z., Matematyczne aspekty procesu produkcyjnego, Warszawa 1969

[143] Perl J., On finding all solutions of the partitioning problem, W: Management of research and education, Wrocław, 1975

[144] Podinowskij W.W., Gawriłow W.U., Optimizacja po posledowatelno primeniajemym kryteriam, Moskwa 1975

[145] Podczasowa T.P., Ob ocenkach i wybore prawił predpodczenija w zadaczach kalendarnogo płanirowanija, W: Awtomatizirowannyje sistiemy uprawlenija predprijatiew, Kijów 1968

[146] Polya G., Jak rozwiązać, Warszawa 1964

[147] Polya G., Odkrycie matematyczne, Warszawa 1975

[148] Portugal W.M., i inni, Obszczezawodskoje kalendarnoje płanirowanije s ograniczenijami po resursam, W: Prikładnaja matematika i kibiernetika, Moskwa 1973

[149] Portugal W.M., Primienienije kombinatornogo mietoda dla reszenija zadaczi sostawlenija raspisanija, "Kibiernetika", 1967, nr 4

[150] Pospiełow D.A., Puszkin B.N., Myszlenije i awtomaty, Moskwa 1972

[151] Priłuckij M.H., Szewczenko W.N., Ob odnom podchodzie k sostawlenju optimalnogo plana rabot, W: Issledowania po kibiernetikie, red. A.A.Liapunow, Moskwa 1970

[152] Primienienije elektronno-wyczislitelnych maszin w uprawlenii proizwodstwom, praca zbiorowa pod red. O.W.Kozłowej, Moskwa 1966

[153] Primienienije EWM w uczebnom procesie, pod red. A.I.Bierga, Moskwa 1969

[154] Programmirowannoje obuczenie i mietody awtomatizacji uczebnogo procesa, Moskwa 1968

[155] Pulczyn W., Elementy teorii grafów, Warszawa 1968

[156] Puszkin B., Heurystyka, Warszawa 1970

[157] Raczyński S., Zastosowanie komputera Odra 1304 do układania harmonogramu zajęć dydaktycznych w szkole wyższej, "Informatyka" 1973, nr 3

[158] Radczik I.A., O dwoch ekstremalnych zadaczach sietiewogo planirowanija, "Ekonomika i Matematiczeskije Mietody", 1968, nr 3

[159] Radzikowski W., Metody symulacyjne w zarządzaniu, Warszawa 1972

[160] Rastrigin L.A., Słuczajnyj poisk - problemy puti i pierspektiwy, W: Waprosy kibiernetiki - problemy słuczajnogo poiska, Moskwa 1973

[161] Safronienko W.A., Matematiczeskoje i elektronnoje modelirowanje zadaczi optimalnogo kalendarnogo płanirowanija, Minsk 1972

[162] Schmidt G., Ströhlein T., A Boolean Matrix Iteration in Timetable Construction, Prace Naukowe Technicznego Uniwersytetu w Monachium, 1974, nr 7406

[163] Schmidt G., Ströhlein T., Enige operative Ausätze zur Lösung von Studenplanproblem, Prace Naukowe Technicznego Uniwersytetu w Monachium, 1973, nr 7312

[164] Siemionow A.I., Portugal W.M., Zadaczi tieorii raspisanji w kalendarnom płanirowanji miełkosierijnogo proizwodstwa, Moskwa 1972

[165] Slagle J., Isskusstwiennyj intielekt, Moskwa 1973

[166] Smith G., Sefton L.M., On Lions Counter Example for Gotlieb's Method for the Construction of School Timetables, "Communications ACM", 1974 nr 4

[167] Solich R., Metoda wyznaczania kolejności produkcji - różnych kompletów detali, Prace COPAN, Warszawa 1971, nr 47

[168] Solich R., Zadanie planowania kalendarzowego dla grup detali, Prace COPAN, Warszawa 1973, nr 109

[169] Steczkowski J., Statystyczna procedura określania struktury zbiorowości, Kraków 1970

[170] Sytnik W.F., Ispolzowanije procedury Monte-Carlo dla reszenija zadaczi kalendarnogo płanirowanija, "Issledowanije opieracji i ASU", Kijów 1973, nr 1

[171] Sytnik W.F., Optimalnyje i statisticzeskije modeli płanirowania proizwodstwa, Kijów 1972

[172] Szamkołowicz L., O podstawach teorii planowania sieciowego, Prace Naukowe Instytutu Matematycznego i Fizyki Teoretycznej Politechniki Wrocławskiej, 1971, nr 4

[173] Szczerbak A.F., Lewin G.L., Wołczek B.A., K woprosu reszenija odnogo kłassa zadacz tieorii raspisanij, "Kibiernetika", 1973, nr 5

[174] Szkurba W.W., Interwały oczerednosti w zadaczach uporjadoczenja, "Kibiernetika", 1970, nr 2

[175] Szkurba W.W., O zadaczach uporjadoczenja, "Kibiernetika", 1967, nr 2

[176] Szurba W.W., Wyczislitelnyje schiemy reszenja zadacz tieorii raspisanji, "Kibiernetika", 1965, nr 3

[177] Szkurba W.W. i inni, Zadaczi kalendarnogo płanirowanja i mietody ich reszenja, Kijów 1966

[178] Szubkina I.P., Kompleks zadacz opieratiwnogo płanirowanija proizwodstwa, "Ekonomika i Matematiczeskije Mietody", 1974, nr 3

[179] Tanajew W.S., Szkurba W.W., Wwedenje w tieorju raspisanji, Moskwa 1975

[180] Thornton B., Timetabling and Scheduling Problems, "The Computer Journal", 1966, nr 9

[181] Tieriechow L., Metody ekonomiczno-matematyczne, Warszawa 1970

[182] Tierechow L.L., Mołczan W.A., Waprosy automatizirowannogo sostawlenija raspisanija uczebnych zanijatii s pomoszczju EWM "Minsk 22", "Mechanizacja Uczeta i Wyczislitelnych Rabot", 1972, nr 14

[183] Titow W.W., Zadacza kalendarnogo płanirowanija i jejo pribliżennoje reszenije, W: Optimalnoje płanirowanije, nr 11, Nowosybirsk 1968

[184] Tornau S., Logiczny wstęp do matematyki, Kraków 1974

[185] Wagner H., Osnowy issledowanja operacji, Moskwa 1972

[186] Welsh B.J., Powell M.B., An Upper Bound for the Chromatic Number of a Graf and its Applications to Timetabling Problems, "The Computer Journal", 1967, nr 1

[187] Wieduta N.I., Safronienko W.A., Matematiczeskaja model kalendarnogo płanirowanja serijnogo proizwodstwa, "Ekonomika i Matematiczeskije Mietody", 1970, nr 6

[188] Wołczewski J., Zastosowanie elektronicznej techniki obliczeniowej w planowaniu i ewidencji produkcji, Warszawa 1972

[189] Wołoszczenko A.B., Ob odnoj zadaczi teorii raspisanji, "Mechanizacja Uczeta i Wyczislitelnych Rabot", 1970, nr 11

[190] Wood D.C., A Technique for Colouring a Graph Applicable to Large Scale Timetabling Problems, "The Computer Journal", 1969, nr 4

[191] Woronow A.A., Issledowanije operacji i uprawlenije, Moskwa 1970

[192] Yule A.P., Extensious to the Heuristic Algorithm for University Timetables, "The Computer Journal", 1968, nr 4

[193] Zawielskij M.G., Optimalnoje płanirowanje na predprijatii, Moskwa 1970

[194] Zestaw programów nt. "Układanie rozkładu zajęć w szkole wyższej", Instytut Metod Rachunku Ekonomicznego WSE Wrocław 1974

[195] Zuchowickij S.I., Radczik I.A., Matematiczeskije mietody sietiewogo płanirowanija, Moskwa 1965

[196] Zykow A.A., Tieoria koniecznych grafow, Nowosybirsk 1969

A N E K S

Zał. 1

PROGRAM I

```
      MASTER ZAJEC
      INTEGER WTS,WWP,WCH,WD,WWK,D,DP,KS(80),KIER(5),A,WWK1,WWG,WZS
      DIMENSION MCK(10,6,12),MCS(10,6,10),WTS(60),A(14),KP(200,11),
     1WWP(89),WCH(89),WD(89),KK(10,6),NK(10,5),WWK(5),D(6,10),DP(6,10),
     1KX(28,12),KY(28,10),MCG(10,6,12),WWK1(5),WWG(20),WZS(10)
      COMMON /NOW/WCH,WWP,N,M,WD,L,N1,IY,I,I1
C
      READ(1,2) LWWK
    2 FORMAT (I2)
      READ(1,3)((KK(I,J),J=1,6),I=1,LWWK)
    3 FORMAT (I2,5I1)
      IZ=0
      K=0
      DO 4 I=1,LWWK
      DO 4 J=1,5
      IF(KK(I,J+1))0,4,0
      K=K+1
      NK(I,J)=K
    4 CONTINUE
      LK=K
      READ(1,447) LWKX
  447 FORMAT(I3)
      READ(1,448)((KX(I,J),J=1,12),I=1,LWKX)
  448 FORMAT (I3,I2,10I2)
      K=0
      DO 449 I=1,LWKX
      DO 449 J=1,10
      IF(KX(I,J+2)) 0,450,0
      K=K+1
      KY(I,J)=K
      GO TO 449
  450 KY(I,J)=0
  449 CONTINUE
      LKG=K
      READ(1,7) LSAL
    7 FORMAT(I2)
      REWIND 3
      KA=3
      LC=0
  301 DO 18 I=1,10
      DO 18 J=1,6
      DO 17 K=1,10
      MCK(I,J,12)=0
   17 MCK(I,J,K)=1
   18 MCK(I,J,11)=0
      WRITE (3) MCK
      LC=LC+10
      IF(LC.LT.LK) GO TO 301
      NS=0
  299 DO 8 I=1,10
      DO 8 J=1,6
C
      DO 8 K=1,10
    8 MCS(I,J,K)=1
      IR=1
    9 NS=NS+1
      READ (1,10) (KS(I),I=1,40)
   10 FORMAT (I3,I1,38I2)
      WTS(NS)=KS(1)/100
      IF(KS(2).EQ.0) GO TO 12
      DO 11 J=1,KS(2)
```

```
      IF(J.NE.1) GO TO 13
      I=3
      GO TO 14
   13 I=I+KS(I+1)+2
   14 LB=KS(I+1)
      DO 15 K=1,LB
      M=KS(I+1+K)
      M1=KS(I)
   15 MCS(IR,M1,M)=0
C
   11 CONTINUE
      IF(IR.EQ.10) GO TO 302
   12 IR=IR+1
      IF(NS.EQ.LSAL) GO TO 302
      GO TO 9
  302 WRITE(3) MCS
      IF(NS.LT.LSAL) GO TO 299
      ENDFILE 3
      LP=0
   19 LP=LP+1
      READ(1,20) (A(I),I=1,14)
   20 FORMAT (I3,2I2,10I1,I3)
      IF(A(1).EQ.999) GO TO 21
      IF(A(1).EQ.888) GO TO 480
      KP(LP,1)=A(1)
      KP(LP,2)=A(3)
      KP(LP,3)=A(4)
      KP(LP,4)=A(5)
      KP(LP,5)=A(6)
      KP(LP,6)=A(7)
      KP(LP,7)=A(8)
      KP(LP,8)=A(9)
      KP(LP,9)=A(11)
      KP(LP,10)=A(12)
      KP(LP,11)=A(14)
      GO TO 19
  480 LP=LP-1
      GO TO 481
   21 LP=LP-1
      LP1=LP
      GO TO 19
  481 KIER(1)=0
      KIER(2)=4
      KIER(3)=3
      KIER(4)=2
      KIER(5)=1
      KW=0
  222 KW=KW+1
      DO 117 J=1,51
      WD(J)=-1
  117 CONTINUE
      IZ=KIER(KW)
      K1=0
      IF(IZ.EQ.1) GO TO 482
      J=1
      N=LP1
      GO TO 483
  482 J=LP1+1
      N=LP
```

```
483 DO 23 I=J,N
    IF(KP(I,3).NE.IX) GO TO 23
    K1=K1+1
    WWP(K1)=I
 23 CONTINUE
    IF(K1.EQ.0) GO TO 22
    IF(K1.EQ.1) GO TO 43
    DO 24 I=1,K1
    N=WWP(I)
    IF(KP(N,11)) 0,0,25
    WCH(I)=KP(N,11)
    GO TO 24
 25 WCH(I)=1
 24 CONTINUE
    N=1
    M=K1
    CALL MALE
    L=1
 32 N1=K1
    I=1
    I1=0
    CALL PODPR
    IF(IY.EQ.4) GO TO 43
    K2=L
    I=I-1
    DO 28 J=1,I
    N=WD(J)
    M=WWP(N)
    WCH(N)=KP(M,9)
 28 CONTINUE
    N=WD(1)
    M=WD(I)
    CALL SORT
    K3=M
    L=N
 36 N1=K3
    I=1
    I1=0
    CALL PODPR
    IF(IY.EQ.3) GO TO 31
    L=K2+1
    GO TO 32
 31 K4=L
    I=I-1
    DO 33 J=1,I
    N=WD(J)
    M=WWP(N)
    WCH(N)=KP(M,10)
 33 CONTINUE
    N=WD(1)
    M=WD(I)
    CALL MALE
    K5=M
    L=N
 40 N1=K5
    I=1
    I1=0
    CALL PODPR
    IF(IY.EQ.3) GO TO 35
    L=K4+1
    GO TO 36
```

```
   35 K6=L
      I=I-1
      DO 37 J=1,I
      N=WD(J)
      M=WWP(N)
      WCH(N)=KP(M,2)
   37 CONTINUE
      N=WD(1)
      M=WD(I)
      CALL SORT
32112 K7=M
      L=N
   42 N1=K7
      I=1
      I1=0
      CALL PODPR
      IF(IY.EQ.3) GO TO 39
      L=K6+1
      GO TO 40
   39 K8=L
      I=I-1
      DO 41 J=1,I
      N=WD(J)
      M=WWP(N)
      WCH(N)=KP(M,1)
   41 CONTINUE
      N=WD(1)
      M=WD(I)
      CALL SORT
      L=K8+1
      GO TO 42
   43 WRITE(2,148)(WWP(I),I=1,52)
  148 FORMAT(/52I2)
      I=0
  444 I=I+1
      IF(KA.EQ.3) GO TO 411
      REWIND 4
      GO TO 412
  411 REWIND 3
  412 LC=0
      LQ=-9
      N=WWP(I)
      WRITE(2,54)(N)
   54 FORMAT(/9HNR.PRZEDM,I2)
      DO 45 J=1,LWWK
      IF(KK(J,1).EQ.KP(N,2)) GO TO 46
   45 CONTINUE
   46 M=KK(J,1)
      IF(KP(N,3)) 47,0,47
      LZK=0
      DO 48 L=1,5
      IF( K(J,L))49,4[illegible],0
        K(L)=  (J,L)
      WWK1(L)=KK(J,L+1)
      LZK=LZK+1
C
   48 CONTINUE
      GO TO
   47 LZK=0
```

```
      DO 50 K=1,KP(N,3)
      DO 51 L=2,6
      IF(KK(J,L).EQ.KP(N,K+3)) GO TO 52
   51 CONTINUE
   52 WWK(K)=NK(J,L-1)
      WWK1(K)=KK(J,L)
      LZK=LZK+1
   50 CONTINUE
   49 WRITE(2,53)(WWK(L),L=1,5)
   53 FORMAT (/5I2)
      IF(IZ.NE.1) GO TO 484
      K3=0
      DO 485 K=1,LZK
      M1=M*10+WWK1(K)
      DO 486 L=1,LWKX
      IF(M1.EQ.KX(L,1)) GO TO 487
  486 CONTINUE
      WRITE (2,488)
  488 FORMAT(10H BLAD NR 5)
      STOP 9494
  487 DO 489 J=1,KX(L,2)
      K3=K3+1
  489 WWG(K3)=KY(L,J)
  485 CONTINUE
  484 DO 60 K=1,6
      DO 60 L=1,10
      D(K,L)=1
   60 DP(K,L)=0
      IF(IZ.NE.1) GO TO 490
      J1=1
  491 M=WWG(J1)
      DO 492 K=1,6
      DO 492 L=1,10
      D(K,L)=1
  492 DP(K,L)=0
      IF(M.GT.LKG) GO TO 306
      GO TO 305
  490 DO 56 J=1,LZK
      M=WWK(J)
      IF(M.GT.LK) GO TO 306
  305 IF(LC.LT.M) GO TO 309
      IF(LQ.LE.M) GO TO 310
      IF(KA.EQ.3) GO TO 414
      REWIND 4
      GO TO 415
  414 REWIND 3
  415 LC=0
      LQ=-9
  309 IF(KA.EQ.3) GO TO 416
      READ(4) MCK
      GO TO 417
  416 READ(3) MCK
  417 LC=LC+10
      LQ=LQ+10
      GO TO 305
  306 WRITE (2,307)
  307 FORMAT(9HBLAD NR 1)
      STOP 20
  310 DO 311 J=1,10
      LI=LQ-1+J
      IF(LI.EQ.M) GO TO 312
```

```
  311 CONTINUE
      WRITE (2,314)
  314 FORMAT(9H BLADNR 2 )
      STOP 21
  312 DO 57 K=1,6
      IF(IZ.EQ.1) GO TO 493
      IF (MCK(MQ,K,11).EQ.3) GO TO 59
      GO TO 494
  493 M23=MCK(MQ,K,12)+KP(N,10)
      IF(MCK(MQ,K,12).GE.8.OR.M23.GT.8) GO TO 59
  494 DO 58 L=1,10
      IF(MCK(MQ,K,L).EQ.1) GO TO 308
      D(K,L)=0
      GO TO 58
  308 D(K,L)=D(K,L)*MCK(MQ,K,L)
   58 CONTINUE
   57 CONTINUE
      IF(IZ.EQ.1) GO TO 63
   56 CONTINUE
      GO TO 63
   59 DO 62 L=1,10
   62 D(K,L)=0
      GO TO 57
   63 M=KP(N,10)
      DO 67 K=1,6
      DO 65 L=1,10
      IF(D(K,L)) 0,65,0
      IF(L+M.GT.11) GO TO 65
      IF(M.EQ.1) GO TO 121
      M2=M-1
      DO 66 J=1,M2
      IF(D(K,J+L).NE.1) GO TO 65
   66 CONTINUE
  121 DP(K,L)=1
   65 CONTINUE
   67 CONTINUE
  496 IF(IZ.EQ.1) GO TO 495
      IF(LC.LT.LK) GO TO 363
      GO TO 365
  495 IF(LC.LT.LKG) GO TO 363
      GO TO 365
  363 IF(KA.EQ.3) GO TO 361
      READ(4) MCK
      GO TO 362
  361 READ(3) MCK
  362 LC=LC+10
      GO TO 496
  365 KC=0
      KQ=-9
      A(1)=0
      K=0
   69 K=K+1
      DO 70 L=1,1
      A(1)=A(1)+DP(K,L)
   70 CONTINUE
C
      IF(K.LT.6) GO TO 69
      K11=0
C
```

```
    IF(A(1)) 0,0,71
 87 WRITE(2,172)(KP(N,L),L=1,11)
172 FORMAT(16H NIE WPROWADZONO,11I3)
    IF(IZ,NE.1) GO TO 497
    WRITE(2,498) WWG(J1)
498 FORMAT(9H GRUPA NR,I3)
497 GO TO 44
 71 IF(KP(N,11)) 72,72,0
    A(1)=KP(N,11)/100
    A(1)=A(1)*100
    MS=KP(N,11)-A(1)
    ML=MS
    K11=1
    GO TO 81
 72 L=60
 74 IF(WTS(L).EQ.KP(N,9)) GO TO 73
    L=L-1
    IF(L) 75,75,0
    GO TO 74
 75 WRITE(2,76)
 76 FORMAT(/9H COS ZLE /)
 73 NS=L
    K5=0
    DO 499 K4=1,NS
    IF(WTS(K4)) 0,0,499
    K5=K5+1
    WZS(K5)=K4
499 CONTINUE
    ML=1
    MS=NS
    IF(IZ,NE.1) GO TO 81
500 DO 502 K=1,5
    DO 501 L=1,10
    IF(DP(K,L),EQ.1) GO TO 79
501 CONTINUE
502 CONTINUE
    K=6
    DO 503 L=1,10
    IF(DP(K,L).EQ.1) GO TO 79
503 CONTINUE
    GO TO 84
 81 DO 77 K=1,6
    DO 78 L=1,10
    IF(DP(K,L).EQ.1) GO TO 79
 78 CONTINUE
 77 CONTINUE
    GO TO 84
 79 MZ=M+L-1
317 IF(KC,LT.MS) GO TO 315
    IF(KQ,LE.MS) GO TO 316
    IF(KA.EQ.3) GO TO 366
    REWIND 4
    GO TO 367
366 REWIND 3
367 KC=0
    KQ=-9
    LC=0
368 IF(KA.EQ.3) GO TO 369
    READ (4) MCK
    GO TO 370
369 READ (3) MCK
```

```
370 LC=LC+10
    IF(IZ.EQ.1) GO TO 504
    IF(LC.LT.LK) GO TO 368
    GO TO 315
504 IF(LC.LT.LKG) GO TO 368
315 IF(KA.EQ.3) GO TO 371
    READ (4) MCS
    GO TO 372
371 READ (3) MCS
372 KC=KC+10
    KQ=KQ+10
    GO TO 317
316 DO 318 MQ=1,10
    LI=KQ-1+MQ
    IF(LI.EQ.MS) GO TO 319
318 CONTINUE
319 IF(K11.EQ.1) GO TO 521
    DO 505 K6=1,KS
    IF(WZS(K6).EQ.MS) GO TO 82
505 CONTINUE
521 DO 80 J=L,MZ
    IF (MCS(MQ,K,J).EQ.1) GO TO 80
    GO TO 82
 80 CONTINUE
    GO TO 85
 82 MS=MS-1
    IF(MS.GE.ML) GO TO 79
    IF(ML.EQ.1) GO TO 83
    MS=ML
    IF(IZ.EQ.1) GO TO 506
    GO TO 78
506 GO TO 502
 83 MS=NS
    IF(IZ.EQ.1) GO TO 506
    GO TO 78
 84 WRITE(2,86)
 86 FORMAT(/21H NIE ZNALEZIONO SALI I)
    GO TO 87
 85 MZ=M+L-1
    IF(IZ.EQ.1) GO TO 373
    DO 89 LN=1,LZK
373 IF(KA.EQ.3) GO TO 374
    REWIND 4
    REWIND 3
    READ (4) MCK
    GO TO 375
374 REWIND 3
    REWIND 4
    READ (3) MCK
375 LC=10
    LQ=1
    IF(IZ.EQ.1) GO TO 508
    LD=WWK(LQ)
    GO TO 326
508 LD=WWG(J1)
326 IF(LC.LT.LD) GO TO 324
    IF(LQ.LE.LD) GO TO 325
    STOP 9101
324 IF(KA.EQ.3) GO TO 376
```

```
      WRITE(3) MCK
      READ (4) MCK
      LC=LC+10
      LQ=LQ+10
      GO TO 326
  376 WRITE (4) MCK
      READ (3) MCK
      LC=LC+10
      LQ=LQ+10
      GO TO 326
  325 DO 327 MQ=1,10
      LI=LQ-1+MQ
      IF(LI.EQ.LD) GO TO 328
  327 CONTINUE
  328 DO 90 J=L,MZ
      ML=MS*1000+N+1
      MCK(MQ,K,J)=ML
   90 CONTINUE
      IF(M.GE.6) GO TO 263
      MCK(MQ,K,11)=MCK(MQ,K,11)+1
      MCK(MQ,K,12)=MCK(MQ,K,12)+M
      GO TO 329
  263 MCK(MQ,K,11)=3
      MCK(MQ,K,12)=8
  329 IF(KA.EQ.3) GO TO 377
      WRITE (3) MCK
      GO TO 378
  377 WRITE (4) MCK
  378 IF(IZ.NE.1) GO TO 507
      IF(LC.LT.LKG) GO TO 379
      GO TO 380
  507 IF(LC.LT.LK) GO TO 379
  380 KC=0
      WRITE(2,602) LC
  602 FORMAT(3H LC,I3)
      IF(KA.EQ.3) GO TO 389
  387 READ(4) MCS
      KC=KC+10
      GO TO 388
  389 READ(3) MCS
      KC=KC+10
      GO TO 388
  391 WRITE (4) MCS
      GO TO 389
  392 WRITE (3) MCS
      GO TO 387
  321 IF(KA.EQ.3) GO TO 391
      GO TO 392
  388 IF(KC.LT.MS) GO TO 321
      KQ=KC-9
      DO 322 MQ=1,10
      LI=KQ-1+MQ
      IF(LI.EQ.MS) GO TO 323
  322 CONTINUE
      STOP 1945
 323  DO 88 J=L,MZ
      IF(IZ.EQ.1) GO TO 514
      MCS(MQ,K,J)=N+1
      GO TO 88
  514 MCS(MQ,K,J)=N*1000+NWO(J1)
C
```

```
   88 CONTINUE
      IF(KA.EQ.3) GO TO 393
      WRITE (3) MCS
  396 IF(KC.LT.LSAL) GO TO 394
      ENDFILE 3
      WRITE(2,601) KC
  601 FORMAT(4H KC3,I3)
      KA=3
      GO TO 395
  394 READ (4) MCS
      KC=KC+10
      WRITE(3) MCS
      GO TO 396
  393 WRITE(4) MCS
  397 IF(KC.LT.LSAL) GO TO 398
      ENDFILE 4
      KA=4
      WRITE(2,603) KC
  603 FORMAT(3H KC,I3)
      GO TO 395
  398 READ (3) MCS
      KC=KC+10
      WRITE(4) MCS
      GO TO 397
  395 IF(IZ.EQ.1) GO TO 44
      GO TO 89
  379 IF(KA.EQ.3) GO TO 383
      READ(4) MCK
      WRITE (3) MCK
      LC=LC+10
      GO TO 378
  383 READ (3) MCK
      WRITE (4) MCK
      LC=LC+10
      GO TO 378
   89 CONTINUE
      DO  92 LN=1,LZK
      LD=WWK(LN)
   92 CONTINUE
C
   44 IF(IZ.NE.1) GO TO 510
      J1=J1+1
      REWIND 3
      REWIND 4
      LC=0
      LQ=-9
      IF(J1.LE.K3) GO TO 491
  510 IF(I.LT.K1) GO TO 444
C
   22 IF(KW.LT.5) GO TO 222
      REWIND 3
      REWIND 4
      IF(IZ.EQ.1) GO TO 511
      I=1
      IR=1
      REWIND 3
      REWIND 4
      K2=0
      LC=0
```

```
451 DO 452 J=1,LWWK
    DO 452 K=1,5
    IF(NK(J,K).EQ.I) GO TO 454
452 CONTINUE
454 MQ=KK(J,1)*10+KK(J,K+1)
    DO 455 J=1,LWKX
    IF(MQ.EQ.KX(J,1)) GO TO 456
455 CONTINUE
    WRITE(2,457)
457 FORMAT (9H BLAD NR4)
    STOP 9292
456 M=KX(J,2)
    IF(LC.LT.I) GO TO 459
467 DO 462 K1=1,10
    LI=LC-10+K1
    IF(LI.EQ.I) GO TO 463
462 CONTINUE
463 IF(IR.GT.10) GO TO 460
    DO 461 J=1,6
    DO 461 K=1,12
461 MCG(IR,J,K)=MCK(K1,J,K)
    M=M-1
    K2=K2+1
    IR=IR+1
    IF(M) 0,0,463
    I=I+1
    IF(I.LE.LK) GO TO 451
    WRITE(2,464) K2
464 FORMAT(11H ILOSC GRUP,I3)
    IF(KA.EQ.3) GO TO 522
    WRITE(3) MCG
    GO TO 523
522 WRITE(4) MCG
523 LC=0
    IZ=1
    LC=0
    IF(KA.EQ.3) GO TO 470
471 READ(4) MCS
    WRITE(3) MCS
    LC=LC+10
    IF(LC.LT.LSAL) GO TO 471
    KA=3
    ENDFILE 3
    GO TO 481
470 READ(3) MCS
    WRITE (4) MCS
    LC=LC+10
    IF(LC.LT.LSAL) GO TO 470
    KA=4
    ENDFILE 4
    GO TO 481
459 IF(KA.EQ.3) GO TO 465
    READ(4) MCK
    GO TO 466
465 READ(3) MCK
466 LC=LC+10
    GO TO 467
460 IF(KA.EQ.3) GO TO 468
    WRITE (3) MCG
    GO TO 469
468 WRITE (4) MCG
```

```
 469 IR=1
     GO TO 463
 511 IF(KA.EQ.3) GO TO 401
     WRITE (3) ((KP(I,J),J=1,11),I=1,80)
     WRITE(3)((KP(I,J),J=1,11),I=81,170)
     WRITE(3)((KP(I,J),J=1,11),I=171,200),KK,NK,LWWK,LSAL,LKG,LWKX,
    1KX,KY,LP1
     LC=0
 330 READ(4)MCK
     WRITE(2,445)(((MCK(I,J,K),K=1,12),J=1,6),I=1,10)
     WRITE(3) MCK
     LC=LC+10
     IF(LC.LT.LKG) GO TO 330
     LC=0
 331 READ(4)MCS
     WRITE(2,446)(((MCS(I,J,K),K=1,10),J=1,6),I=1,10)
 446 FORMAT(10I3)
     WRITE(3) MCS
     LC=LC+10
     IF(LC.LT.LSAL) GO TO 331
     KA=3
     GO TO 402
 401 WRITE (4) ((KP(I,J),J=1,11),I=1,80)
     WRITE(4)((KP(I,J),J=1,11),I=81,170)
     WRITE(4)((KP(I,J),J=1,11),I=171,200),KK,NK,LWWK,LSAL,LKG,LWKX,
    1KX,KY,LP1
     LC=0
 403 READ (3) MCK
     WRITE(2,445)(((MCK(I,J,K),K=1,12),J=1,6),I=1,10)
 445 FORMAT(12I3)
     WRITE (4)MCK
     LC=LC+10
     IF(LC.LT.LKG) GO TO 403
     LC=0
 404 READ(3)MCS
     WRITE(2,446)(((MCS(I,J,K),K=1,10),J=1,6),I=1,10)
     WRITE(4) MCS
     LC=LC+10
     IF(LC.LT.LSAL) GO TO 404
     KA=4
     WRITE(2,1121)
1121 FORMAT(////28H TASME MAG. WYNIKI2 ZACHOWAC)
     ENDFILE 4
     GO TO 405
 402 ENDFILE 3
     WRITE (2,407)
 407 FORMAT(////28H TASME MAG. WYNIKI1 ZACHOWAC)
 405 STOP 1212
     END

END OF SEGMENT, LENGTH  5490, NAME  ZAJEC
```

```
      SUBROUTINE PODPR
      INTEGER WCH(89),WWP(89),N,M,WD(89),L
      COMMON /NOW/WCH,WWP,N,M,WD,L,N1,IY,I,I1
   22 IF(L.GE.N1) GO TO 31
      IF(WCH(L).EQ.WCH(L+1)) GO TO 23
      IF(I1.GT.0) GO TO 29
   26 L=L+1
      GO TO 22
   23 IF(I1.GT.0) GO TO 24
      WD(I)=L
      WD(I+1)=L+1
      I=I+2
      GO TO 25
   24 WD(I)=L+1
      I=I+1
   25 I1=I1+1
      GO TO 26
   27 IY=4
   28 RETURN
   29 IY=3
      GO TO 28
   31 IF(I1.GT.0) GO TO 29
      GO TO 27
      END

END OF SEGMENT, LENGTH   171, NAME  PODPR
```

```
      SUBROUTINE SORT
      INTEGER WCH(89),WWP(89)
      COMMON /NOW/WCH,WWP,N,M,WD,L,N1,IY,I,I1
      DO 21 I=N,M
      IA=WCH(I)
      J1=I
      J=I+1
   23 IF(J.GT.M) GO TO 22
      IF(IA.LE.WCH(J)) GO TO 20
      IA=WCH(J)
      J1=J
   20 J=J+1
      GO TO 23
   22 WCH(J1)=WCH(I)
      IA1=WWP(I)
      WCH(I)=IA
      WWP(I)=WWP(J1)
      WWP(J1)=IA1
   21 CONTINUE
      RETURN
      END

END OF SEGMENT, LENGTH    140, NAME  SORT
```

```
      SUBROUTINE MALE
      INTEGER WCH(89),WWP(89)
      COMMON /NOW/WCH,WWP,N,M,WD,L,N1,IY,I,I1
      DO 21 I=N,M
      IA=WCH(I)
      J1=I
      J=I+1
   23 IF(J.GT.M) GO TO 22
      IF(WCH(J).LE.IA) GO TO 20
      IA=WCH(J)
      J1=J
   20 J=J+1
      GO TO 23
   22 WCH(J1)=WCH(I)
      IA1=WWP(I)
      WCH(I)=IA
      WWP(I)=WWP(J1)
      WWP(J1)=IA1
   21 CONTINUE
      RETURN
      END

END OF SEGMENT, LENGTH   139, NAME  MALE
```

PROGRAM II

```
      MASTER DRUK
      INTEGER MG(14),MT(1400),KS(50),WWK(5),KD(6),KDW(2),MZK(2)/WWG(20)
      DIMENSION MCS(10,6,10),MCK(10,6,12),KP(200,11),KK(10,6),NK(10,5),
     -LD(10,11),KX(28,12),KY(28,10)
      REWIND 3
      READ(3) ((KP(I,J),J=1,11),I=1,80)
      READ(3)((KP(I,J),J=1,11),I=81,170)
      READ(3)((KP(I,J),J=1,11),I=171,200),KK,NK,LWWK,LSAL,LKG,LWKX,
     1KX,KY,LP1
      NR1=37
      NR2=91
      K1=1
    3 READ(1,2) (M1,M2)
    2 FORMAT (A3,A4)
      KS(K1)=M2
      K1=K1+1
      IF(K1.LE.LSAL) GO TO 3
      READ(1,6) LP
    6 FORMAT(I3)
      K1=1
      J=0
    4 READ(1,5) (MG(I),I=1,14)
    5 FORMAT (A3,A2,6(A4,A3))
      DO 8 J1=1,12
      J=J+1
      MT(J)=MG(J1+2)
C
    8 CONTINUE
      K1=K1+1
      IF(K1.LE.LP) GO TO 4
      READ (1,16) LKW
      READ (1,17)((LD(I,J),J=1,11),I=1,LKW)
   16 FORMAT(I2)
   17 FORMAT(I2,I1,A3,I1,A3,I1,A3,I1,A3,I1,A3)
      DATA KD(1),KD(2),KD(3),KD(4),KD(5),KD(6),KDW(1),KDW(2)/3HPON,
     13HWTO,3HSRO,3HCZW,3HPIA,3HSOB,2HGN,2HIE/
      DATA MZK(1),MZK(2)/4HWYKL,4HCWI./
  600 I=1
  601 FORMAT(1H1,15X,88(1H-))
      IR=1
      READ (3) MCK
  627 WRITE(2,601)
      WRITE(2,602)
      IF(I.LE.10) GO TO 802
      I=1
      READ (3) MCK
  802 M4=0
  602 FORMAT(1H8,15X,1HI,24X,1HI,34X,1HI,26X,1HI)
      WRITE(2,603)
  603 FORMAT(1H8,15X,28HI   AKADEMIA EKONOMICZNA   I   ,
     -60HT-2,PLAN ZAJEC DYDAKTYCZNYCH     I ZAKLADY NAUKOWO-BADAWCZE I)
      DO 604 J=1,LWKX
      DO 604 J1=1,10
      IF(KY(J,J1).EQ.IR) GO TO 605
  604 CONTINUE
  605 M5=KX(J,J1+2)
      L=KX(J,1)/10
      L1=KX(J,1)-L*10
      DO 901 M=1,LWWK
      IF(KK(M,1).EQ.L) GO TO 902
  901 CONTINUE
```

```
902 DO 903 N=1,5
    IF(KK(M,N+1).EQ.L1) GO TO 904
903 CONTINUE
904 DO 607 J=1,LKW
    IF(L.EQ.LD(J,1)) GO TO 608
607 CONTINUE
    M4=-1
608 M1=L/10
    M2=M1*10
    M2=L-M2
    M3=N*2+1
    IF(M4.EQ.-1) GO TO 609
    WRITE(2,610)KDW(M1),M2,LD(J,M3)
610 FORMAT(1H8,15X,38HI     IM.OSKARA LANGEGO   I    WYDZIAL ,A2,2X,
   -4HROK ,I1,2X,5HKIER ,A3,3X,28HI   OSRODEK KOMPUTEROWY    I)
    GO TO 611
609 WRITE (2,612)KDW(M1),M2
612 FORMAT(1H8,15X,38HI     IM.OSKARA LANGEGO   I    WYDZIAL ,A2,2X,
   -4HROK ,I1,13X,28HI   OSRODEK  KOMPUTEROWY   I)
611 WRITE(2,613)M5
613 FORMAT(1H8,15X,1HI,9X,7HWROCLAW,8X,1HI,13X,
   -6HGRUPA ,I2,13X,1HI,8X,7HWROCLAW,11X,1HI)
    WRITE(2,614)
614 FORMAT(1H8,15X,1HI,6X,15HDZIAL NAUCZANIA,3X,1HI,11X,
   -14HSEMESTR ZIMOWY,9X,1HI,26X,1HI)
    WRITE(2,602)
    WRITE(2,615)
615 FORMAT(1H8,15X,88(1H-))
    WRITE(2,616)
616 FORMAT(1H8,15X,1HI,11X,1HI,9X,1HI,8X,1HI,20X,
   -1HI,10X,1HI,23X,1HI)
    WRITE(2,617)
617 FORMAT(1H9,15X,1HI,3X,
   -60HDZIEN    I GODZINY I   SALA   I   NAZWA PRZEDMIOTU   I   RODZAJ   I,
   -7X,5HUWAGI,11X,1HI)
    WRITE(2,618)
618 FORMAT(1H8,15X,23HI   TYGODNIA I   ZAJEC   I,
   -8X,1HI,20X,1HI,3X,8HZAJEC  I,23X,1HI)
    WRITE(2,615)
    K=1
619 L=1
620 IF(MCK(I,K,L).LE.1) GO TO 621
    M1=MCK(I,K,L)/1000
    N=M1*1000
    N=MCK(I,K,L)-N-1
    IF(KP(N,1).EQ.NR1.OR.KP(N,1).EQ.NR2.OR.N.GT.LP1) GO TO 622
    NZ=1
    GO TO 623
622 NZ=2
623 N1=KP(N,1)
    M=KP(N,10)
    LG=L+7
    LG1=LG+M

    WRITE(2,616)
    WRITE(2,624)KD(K),LG,LG1,KS(M1),MT(NP),MT(NP+1),MT(NP+2),
   -MT(NP+3),MZK(NZ)
624 FORMAT(1H8,15X,1HI,4X,A3,4X,3HI    ,I2,1H-,I2,5H  I   ,A4,
   -6H  I    ,A4,A3,A4,A3,3X,1HI,3X,A4,3X,1HI,23X,1HI)
```

```
      WRITE(2,625)MT(NP+4),MT(NP+5),MT(NP+6),MT(NP+7)
  625 FORMAT(1H8,15X,1HI,11X,1HI,9X,1HI,8X,1HI,3X,A4,A3,A4,A3,
     -3X,1HI,10X,1HI,23X,1HI)
      WRITE(2,625)MT(NP+8),MT(NP+9),MT(NP+10),MT(NP+11)
      WRITE(2,616)
      L=L+M
      GO TO 626
  621 L=L+1
  626 IF(L.LE.10) GO TO 620
      K=K+1
      IF(K.LE.6) GO TO 619
      I=I+1
      WRITE(2,615)
      IR=IR+1
      IF(IR.LE.LKG) GO TO 627
      I=1
      IR=1
      READ(3)MCS
    7 I1=0
      WRITE(2,9)
    9 FORMAT(1H1,120(1H-))
      IF(I.LE.10) GO TO 803
      I=1
      READ(3)MCS
  803 WRITE (2,11)
      LX=0
   11 FORMAT(1H8,25X,1HI,62X,1HI,30X,1HI)
      WRITE(2,12)
   12 FORMAT(1H8,4X,22HAKADEMIA EKONOMICZNA I,13X,
     -40HT-1 WYKAZ OBCIAZEN SAL DYDAKTYCZNYCH     ,9X,
     -32HI    ZAKLADY NAUKOWO-BADAWCZE     I)
      WRITE(2,11)
      WRITE(2,14)
   14 FORMAT(1H8,10X,7HWROCLAW,8X,1HI,23X,14HSEMESTR ZIMOWY,25X,
     -1HI,11X,7HWROCLAW,12X,1HI)
      WRITE(2,11)
      WRITE(2,15) KS(IR)
   15 FORMAT(1H8,7X,19HDZIAL NAUCZANIA   I,26X,5HSALA ,A4,27X,
     -1HI,30X,1HI)
      WRITE (2,11)
      WRITE (2,10)
   10 FORMAT(1H8,120(1H-))
      K=1
   19 L=1
   65 IF(MCS(I,K,L).LE.1) GO TO 66
      IF(MCS(I,K,L).GT.999) GO TO 905
      N=MCS(I,K,L)-1
      GO TO 906
  905 N=MCS(I,K,L)/1000
      NX=MCS(I,K,L)-N*1000
  906 M=KP(N,10)
      M1=KP(N,2)/10
      M2=M1*10
      M2=KP(N,2)-M2
      N1=KP(N,1)
      IF(NR1.EQ.N1.OR.NR2.EQ. 1.OR.N.GT.LP1) GO TO 81
      NZ=1
      GO TO 82
   81 NZ=2
   82 LG=L+7
      LG1=LG+M
```

```
      IF(MCS(I,K,L).GT.999) GO TO 907
      IF(KP(N,3).EQ.0) GO TO 31
      DO 32 M3=1,LKW
      IF(LD(M3,1).EQ.KP(N,2)) GO TO 33
   32 CONTINUE
      WRITE(2,34)
   34 FORMAT(10H8BLAD NR-1)
  907 DO 908 LZ=1,LWKX
      DO 908 LQ=1,10
      IF(NX.EQ.KY(LZ,LQ)) GO TO 909
  908 CONTINUE
  909 NX1=KX(LZ,1)/10
      NY=KX(LZ,LQ+2)
      DO 910 M3=1,LKW
      IF(LD(M3,1).EQ.NX1) GO TO 911
  910 CONTINUE
      LX=1
      GO TO 31
  911 NX2=KX(LZ,1)-NX1*10
      DO 912 M5=1,5
      IF(LD(M3,M5*2).EQ.NX2) GO TO 914
  912 CONTINUE
      WRITE(2,34)
      STOP 9192
  914 WWK(1)=2*M5+1
      M6=1
      GO TO 31
   33 DO 35 M4=1,KP(N,3)
      DO 36 M5=1,5
      IF(LD(M3,M5*2).EQ.KP(N,M4+3)) GO TO 37
   36 CONTINUE
   37 WWK(M4)=2*M5+1
   35 CONTINUE
      M6=KP(N,3)
   31 GO TO (20,21,22,23,24,25),M
   20 WRITE(2,27)
   27 FORMAT(1H8,16X,1HI)
      WRITE(2,28) KS(IR),KD(K)
   28 FORMAT(1H8,6H SALA ,A4,2X,A3,2H I)
      WRITE(2,29) LG,LG1
   29 FORMAT (1H8,8H   GODZ.,I2,1H-,I2,4H   I)
      IF(MCS(I,K,L).GT.999) GO TO 916
      IF(KP(N,3).NE.0) GO TO 38
      WRITE(2,39) KDW(M1),M2
   39 FORMAT(1H8,1H ,A2,1H-,I1,12H CALY ROK  I)
      WRITE(2,27)
      GO TO 49
  916 IF(LX.NE.1) GO TO 38
      WRITE(2,918) KDW(M1),M2
  918 FORMAT(1H8,1H ,A2,1H-,I1,11X,1HI)
  920 WRITE(2,919)NY
  919 FORMAT(1H8,7H GRUPA ,I2,7X,1HI)
      GO TO 49
   38 GO TO (40,41,41,41), M6
   40 M7=WWK(1)
      WRITE(2,42) KDW(M1),M2,LD(M3,M7)
      IF(MCS(I,K,L).GT.999) GO TO 920
      WRITE(2,27)
      GO TO 49
```

```
 41  M7=WWK(1)
     M8=WWK(2)
     WRITE(2,43) KDW(M1),M2,LD(M3,M7),LD(M3,M8)
 43  FORMAT(1H8,1X,A2,1H-,I1,1X,A3,2X,A3,2X,1HI)
     GO TO (44,44,45,46),M6
 44  WRITE (2,27)
     GO TO 49
 45  M7=WWK(3)
     WRITE (2,47) LD(M3,M7)
 47  FORMAT(1H8,1X,A3,12X,1HI)
     GO TO 49
 46  M7=WWK(3)
     M8=WWK(4)
     WRITE(2,48) LD(M3,M7),LD(M3,M8)
 48  FORMAT(1H8,1X,A3,2X,A3,7X,1HI)
 49  WRITE(2,50) MZK(NZ)
 50  FORMAT(1H8,1X,A4,11X,1HI)
     NP=1+(N1-1)*12
     WRITE(2,51) MT(NP),MT(NP+1),MT(NP+2),MT(NP+3)
 51  FORMAT(1H8,1X,A4,A3,A4,A3,2H I)
     WRITE(2,52) MT(NP+4),MT(NP+5),MT(NP+6),MT(NP+7)
 52  FORMAT(1H8,1X,A4,A3,A4,A3,2H I)
     WRITE(2,53) MT(NP+8),MT(NP+9),MT(NP+10),MT(NP+11)
 53  FORMAT(1H8,1X,A4,A3,A4,A3,2H I)
     WRITE (2,54)
 54  FORMAT(1H8,17(1H-))
     GO TO 70
 21  WRITE (2,127)
127  FORMAT(1H8,2(16X,1HI))
     WRITE (2,128) KS(IR),KD(K),KS(IR),KD(K)
128  FORMAT(1H8,2(6H SALA ,A4,2X,A3,2H I))
     WRITE (2,129) LG,LG1,LG,LG1
129  FORMAT(1H8,2(8H   GODZ.,I2,1H-,I2,4H   I))
     IF(MCS(I,K,L).GT.999) GO TO 921
     IF(KP(N,3).NE.0) GO TO 138
     WRITE (2,139) (KDW(M1),M2,J1=1,2)
     WRITE(2,127)
     GO TO 149
921  IF(LX.NE.1) GO TO 138
     WRITE(2,922)(KDW(M1),M2,J1=1,2)
922  FORMAT(1H8,2(1H ,A2,1H-,I1,11X,1HI))
923  WRITE(2,924)NY,NY
924  FORMAT(1H8,2(7H GRUPA ,I2,7X,1HI))
     GO TO 149
138  GO TO (140,141,141,141),M6
140  M7=WWK(1)
     WRITE (2,142) KDW(M1),M2,LD(M3,M7),KDW(M1),M2,LD(M3,M7)
142  FORMAT(1H8,2(1X,A2,1H-,I1,1X,A3,7X,1HI))
     IF(MCS(I,K,L).GT.999) GO TO 923
     WRITE (2,127)
     GO TO 149
141  M7=WWK(1)
     M8=WWK(2)
     WRITE(2,143)(KDW(M1),M2,LD(M3,M7),LD(M3,M8),J1=1,2)
143  FORMAT(1H8,2(1X,A2,1H-,I1,1X,A3,2X,A3,2X,1HI))
     GO TO (144,144,145,146),M6
144  WRITE(2,127)
     GO TO 149
145  M7=WWK(3)
     WRITE (2,147)LD(M3,M7),LD(M3,M7)
147  FORMAT(1H8,2(1X,A3,12X,1HI))
```

```
      GO TO 149
  146 M7=WWK(3)
      M8=WWK(4)
      WRITE(2,148) (LD(M3,M7),LD(M3,M8),J1=1,2)
  148 FORMAT(1H8,2(1X,A3,2X,A3,7X,1HI))
  150 FORMAT(1H8,2(1X,A4,11X,1HI))
  149 WRITE(2,150) (MZK(NZ),J1=1,2)
      DO 152 NW=1,3
      NP=1+(N1-1)*12+(NW-1)*4
      WRITE(2,151)(MT(NP),MT(NP+1),MT(NP+2),MT(NP+3),J1=1,2)
  151 FORMAT(1H8,2(1X,A4,A3,A4,A3,2H I))
  152 CONTINUE
      WRITE (2,127)
      WRITE (2,154)
      GO TO 70
   22 WRITE (2,227)
  227 FORMAT(1H8,3(16X,1HI))
      WRITE (2,228) (KS(IR),KD(K),J1=1,3)
  228 FORMAT(1H8,3(6H SALA ,A4,2X,A3,2H I))
      WRITE (2,229) (LG,LG1,J1=1,3)
  229 FORMAT(1H8,3(8H   GODZ.,I2,1H-,I2,4H   I))
      IF(MCS(I,K,L).GT.999) GO TO 925
      IF(KP(N,3).NE.0) GO TO 238
      WRITE(2,239) (KDW(M1),M2,J1=1,3)
  239 FORMAT(1H8,3(1H ,A2,1H-,I1,12H CALY ROK  I))
      WRITE (2,227)
      GO TO 249
  925 IF(LX.NE.1) GO TO 238
      WRITE(2,926)(KDW(M1),M2,J1=1,3)
  926 FORMAT(1H8,3(1H ,A2,1H-,I1,11X,1HI))
  927 WRITE(2,928)(NY,J1=1,3)
  928 FORMAT(1H8,3(7H GRUPA ,I2,7X,1HI))
      GO TO 249
  238 GO TO (240 241,241,241),M6
  240 M7=WWK(1)
      WRITE (2,242) (KDW(M1),M2,LD(M3,M7),J1=1,3)
  242 FORMAT(1H8,3(1X,A2,1H-,I1,1X,A3,7X,1HI))
      IF(MCS(I,K,L).GT.999) GO TO 927
      WRITE (2,227)
      GO TO 249
  241 M7=WWK(1)
      M8=WWK(2)
      WRITE (2,243) (KDW(M1),M2,LD(M3,M7),LD(M3,M8),J1=1,3)
  243 FORMAT(1H8,3(1X,A2,1H-,I1,1X,A3,2X,A3,2X,1HI))
      GO TO (244,244,245,246),M6
  244 WRITE (2,227)
      GO TO 249
  245 M7=WWK(3)
      WRITE (2,247) (LD(M3,M7),J1=1,3)
  247 FORMAT(1H8,3(1X,A3,12X,1HI))
      GO TO 249
  246 M7=WWK(3)
      M8=WWK(4)
      WRITE(2,248) (LD(M3,M7),LD(M3,M8),J1=1,3)
  249 WRITE(2,250) (MZK(NZ),J1=1,3)
      DO 252 NW=1,3
      NP=1+(N1-1)*12+(NW-1)*4
      WRITE (2,251) (MT(NP),MT(NP+1),MT(NP+2),MT(NP+3),J1=1,3)
  252 CONTINUE
```

```
      WRITE (2,227)
      WRITE (2,254)
      GO TO 70
   23 WRITE (2,327)
      WRITE (2,328) (KS(IR),KD(K),J1=1,4)
      WRITE (2,329) (LG,LG1,J1=1,4)
      IF(MCS(I,K,L).GT.999) GO TO 929
      IF(KP(N,3).NE.0) GO TO 338
      WRITE (2,339) (KDW(M1),M2,J1=1,4)
      WRITE (2,327)
      GO TO 349
  929 IF(LX.NE.1) GO TO 338
      WRITE(2,930)(KDW(M1),M2,J1=1,4)
  930 FORMAT(1H8,4(1H ,A2,1H-,I1,11X,1HI))
  931 WRITE (2,932)(NY,J1=1,4)
  932 FORMAT(1H8,4(7H GRUPA ,I2,7X,1HI))
      GO TO 349
  338 GO TO (340,341,341,341),M6
  340 M7=WWK(1)
      WRITE (2,342) (KDW(M1),M2,LD(M3,M7),J1=1,4)
      IF(MCS(I,K,L).GT.999) GO TO 931
      WRITE(2,327)
      GO TO 349
  341 M7=WWK(1)
      M8=WWK(2)
      WRITE (2,343)(KDW(M1),M2,LD(M3,M7),LD(M3,M8),J1=1,4)
      GO TO (344,344,345,346),M6
  344 WRITE (2,327)
      GO TO 349
  345 M7=WWK(3)
      WRITE (2,347) (LD(M3,M7),J1=1,4)
      GO TO 349
  346 M7=WWK(3)
      M8=WWK(4)
      WRITE (2,348) (LD(M3,M7),LD(M3,M8),J1=1,4)
  349 WRITE(2,350) (MZK(NZ),J1=1,4)
      DO 352 NW=1,3
      NP=1+(N1-1)*12+(NW-1)*4
      WRITE (2,351) (MT(NP),MT(NP+1),MT(NP+2),MT(NP+3),J1=1,4)
  352 CONTINUE
      WRITE (2,354)
      GO TO 70
   24 WRITE (2,427)
      WRITE(2,428) (KS(IR),KD(K),J1=1,5)
      WRITE (2,429) (LG,LG1,J1=1,5)
      IF(MCS(I,K,L).GT.999) GO TO 933
      IF(KP(N,3).NE.0) GO TO 438
      WRITE (2,439) (KDW(M1),M2,J1=1,5)
      WRITE (2,427)
      GO TO 449
  933 IF(LX.NE.1) GO TO 449
      WRITE(2,934)(KDW(M1),M2,J1=1,5)
  934 FORMAT(1H8,5(1H ,A2,1H-,I1,11X,1HI))
  935 WRITE(2,936)(NY,J1=1,5)
  936 FORMAT(1H8,5(7H GRUPA ,I2,7X,1HI))
      GO TO 449
  438 GO TO (440,441,441,441),M6
  440 M7=WWK(1)
      WRITE (2,442) (KDW(M1),M2,LD(M3,M7),J1=1,5)
      IF(MCS(I,K,L).GT.999) GO TO 935
      WRITE (2,427)
```

```
      GO TO 449
441   M7=WWK(1)
      M8=WWK(2)
      WRITE (2,443) (KDW(M1),M2,LD(M3,M7),LD(M3,M8),J=1,5)
      GO TO (444,444,445,446),M6
444   WRITE (2,427)
      GO TO 449
445   M7=WWK(3)
      WRITE(2,447) (LD(M3,M7),J1=1,5)
      GO TO 449
446   M7=WWK(3)
      M8=WWK(4)
      WRITE (2,448)(LD(M3,M7),LD(M3,M8),J=1,5)
449   WRITE(2,450) (MZK(NZ),J1=1,5)
      DO 452 NW=1,3
      NP=1+(N1-1)*12+(NW-1)*4
      WRITE (2,451) (MT(NP),MT(NP+1),MT(NP+2),MT(NP+3),J1=1,5)
452   CONTNIUE
      WRITE (2,427)
      WRITE (2,454)
      GO TO 70
 25   WRITE (2,527)
      WRITE (2,528) (KS(IR),KD(K),J1=1,6)
      WRITE (2,529) (LG,LG1,J1=1,6)
      IF(MCS(I,K,L).GT.999) GO TO 937
      IF(KP(N,3).NE.0) GO TO 538
      WRITE(2,539) (KDW(M1),M2,J1=1,6)
      WRITE (2,527)
      GO TO 549
937   IF(LX.NE.1) GO TO 549
      WRITE(2,938)(KDW(M1),M2,J1=1,6)
938   FORMAT(1H8,6(1H ,A2,1H-,I1,11X,1HI))
939   WRITE(2,940)(NY,J1=1,6)
940   FORMAT(1H8,6(7H GRUPA ,I2,7X,1HI))
      GO TO 549
538   GO TO (540,541,541,541),M6
540   M7=WWK(1)
      WRITE(2,542) (KDW(M1),M2,LD(M3,M7),J1=1,6)
      IF(MCS(I,K,L).GT.999) GO TO 939
      WRITE (2,527)
      GO TO 549
541   M7=WWK(1)
      M8=WWK(2)
      WRITE(2,543) (KDW(M1),M2,LD(M3,M7),LD(M3,M8),J1=1,6)
      GO TO (544,544,545,546),M6
544   WRITE (2,527)
      GO TO 549
545   M7=WWK(3)
      WRITE (2,547) (LD(M3,M7),J1=1,6)
546   M7=WWK(3)
      M8=WWK(4)
549   WRITE(2,550)(MZK(NZ),J1=1,5)
      WRITE (2,548) (LD(M3,M7),LD(M3,M8),J1=1,6)
      DO 552 NW=1,3
      NP=1+(N1-1)*12+(NW-1)*4
      WRITE (2,551) (MT(NP),MT(NP+1),MT(NP+2),MT(NP+3),J1=1,6)
552   CONTINUE
      WRITE (2,527)
      WRITE (2,554)
```

```
 70 I1=I1+1
    L=L+M
    GO TO 67
 66 L=L+1
 67 IF(L.LE.10) GO TO 65
    K=K+1
    IF(K.LE.6) GO TO 19
    IF(I1.GT.0) GO TO 68
    WRITE (2,69)
 68 I=I+1
    IR=IR+1
    IF(IR.LE.LSAL) GO TO 7
248 FORMAT(1H8,3(1X,A3,2X,A3,7X,1HI))
250 FORMAT(1H8,3(1X,A4,11X,1HI))
251 FORMAT (1H8,3(1X,A4,A3,A4,A3,2H I))
254 FORMAT(1H8,51(1H-))
327 FORMAT(1H8,4(16X,1HI))
328 FORMAT(1H8,4(6H SALA ,A4,2X,A3,2H I))
329 FORMAT(1H8,4(8H   GODZ.,I2,1H-,I2,4H   I))
339 FORMAT(1H8,4(1X,A2,1H-,I1,12H CALY ROK  I))
342 FORMAT(1H8,4(1X,A2,1H-,I1,1X,A3,7X,1HI))
343 FORMAT(1H8,4(1X,A2,1H-,I1,1X,A3,2X,A3,2X,1HI))
347 FORMAT(1H8,4(1X,A3,12X,1HI))
348 FORMAT (1H8,4(1X,A3,2X,A3,7X,1HI))
350 FORMAT(1H8,4(1X,A4,11X,1HI))
351 FORMAT (1H8,4(1X,A4,A3,A4,A3,2H I))
354 FORMAT(1H8,68(1H-))
427 FORMAT(1H8,5(16X,1HI))
428 FORMAT(1H8,5(6H SALA ,A4,2X,A3,2H I))
429 FORMAT(1H8,5(8H   GODZ.,I2,1H-,I2,4H   I))
439 FORMAT (1H8,5(1X,A2,1H-,I1,12H CALY ROK  I))
442 FORMAT(1H8,5(1X,A2,1H-, 1,1X,A3,7X,1HI))
443 FORMAT(1H8,5(1X,A2,1H-,I1,1X,A3,2X,A3,2X,1HI))
447 FORMAT (1H8,5(1X,A3,12X,1HI))
448 FORMAT(1H8,5(1X,A3,2X,A3,7X,1HI))
450 FORMAT(1H8,5(1X,A4,11X,1HI))
451 FORMAT(1H8,5(1X,A4,A3,A4,A3,2H I))
454 FORMAT (1H8,85(1H-))
527 FORMAT(1H8,6(16X,1HI))
528 FORMAT (1H8,6(6H SALA ,A4,2X,A3,2H I))
529 FORMAT (1H8,6(8H   GODZ.,I2,1H-,I2,4H   I))
539 FORMAT(1H8,6(1X,A2,1H-,I1,12H CALY ROK  I))
542 FORMAT(1H8,6(1X,A2,1H-,I1,1X,A3,7X,1HI))
543 FORMAT(1H8,6(1X,A2,1H-,I1,1X,A3,2X,A3,2X,1HI))
547 FORMAT(1H8,6(1X,A3,12X,1HI))
548 FORMAT (1H8,6(1X,A3,2X,A3,7X,1HI))
550 FORMAT(1H8,6(1X,A4,11X,1HI))
551 FORMAT(1H8,6(1X,A4,A3,A4,A3,2H I))
 42 FORMAT(1H8,1H ,A2,1H-,I1,1X,A3,7X,1HI)
554 FORMAT(1H8,102(1H-))
 69 FORMAT (11H8SALA WOLNA)
154 FORMAT(1H8,34(1H-))
139 FORMAT(1H8,2(1X,A2,1H-,I1,12H CALY ROK  I))
    STOP 1212
    END


END OF SEGMENT, LENGTH  4264, NAME  DRUK
```

Zał. 3

KARTA KIERUNKÓW

Akademia Ekonomiczna Wrocław	Liczba wierszy	

Kod wydziału	Rok	Kody kierunków				

WROCŁAW

LICZ. EGZ

LICZBA WIERSZY

KODY KIERUNKÓW

WYDZ. ROK KIER.

LICZBA GRUP NA KIER.

NUMERY ADMINISTRACYJNE GRUP

SPORZ.

AKADEMIA EKONOMICZNA
WROCŁAW

KARTA SALI

NR DOK.
Zał. 5

NAZWA SALI

LICZBA MIEJSC

A

KOD SALI

LICZ. DNI BLOKOW.

KOD DNIA TYGODNIA

LICZ. GODZ. BLOKOW. W DNIU

KODY GODZIN BLOKOWANYCH

B C D E

UWAGA:
KODY DNI

F G H

PON - 1
WTO - 2
ŚRO - 3
CZW - 4
PIĄ - 5
SOB - 6
NIE - 7

I J K

L Ł M

N O P

R S T

U W X

GODZINY	7-8	8-9	9-10	10-11	11-12	12-13	13-14	14-15	15-16	16-17	17-18	18-19	19-20
KODY GODZIN	00	01	02	03	04	05	06	07	08	09	10	11	12

SPORZĄDZIŁ DATA SPRAWDZIŁ DATA

Zał. 6

AKADEMIA EKONOMICZNA WROCŁAW

KARTA PRZEDMIOTU

NR. DOK.

NAZWA PRZEDMIOTU

INSTYTUT , ZAKŁAD

A KOD PRZEDM. KOD ZAKŁ.

B

WYDZIAŁ ROK

LICZBA KIERUNKÓW C

KIERUNKI

D E F G H

KODY KIERUNKÓW

RODZAJ ZAJĘĆ

I KOD ZAJĘĆ

J

K CZAS TRWANIA ZAJĘĆ

L

M LICZBA SAL LUB KOD SAL

SPORZĄDZIŁ	DATA	SPRAWDZIŁ	DATA

AKADEMIA EKONOMICZNA WROCŁAW	KODY PRZEDMIOTÓW	NR. EGZ.
		LICZ. EGZ.

KOD PRZEDM.	JEDN. ORG. INST.	JEDN. ORG. ZAKŁ.	NAZWA PRZEDMIOTU	UWAGI

SPORZĄDZIŁ	DATA	SPRAWDZIŁ	DATA

WROCŁAW

KODY SAL

LICZ. EGZ.

KOD SALI	NAZWA SALI		KOD SALI	NAZWA SALI		KOD SALI	NAZWA SALI

SPORZĄDZIŁ DATA SPRAWDZIŁ DATA

Zał. 9

Akademia Ekonomiczna Wrocław			KODY KIERUNKÓW				Nr. egz. Licz. egz.	
Kod wydziału i roku	Kod kier.	Nazwa kier.	Kod kier.	Nazwa kier.	Kod kier.	Nazwa kier.	Kod kier.	Nazwa kier.

AKADEMIA EKONOMICZNA IM. OSKARA LANGEGO WROCLAW DZIAL NAUCZANIA	T-Z. PLAN ZAJEC DYDAKTYCZNYCH WYDZIAL GN ROK 3 KIER FIN GRUPA 9 SEMESTR ZIMOWY	ZAKLADY NAUKOWO-BADAWCZE OSRODEK KOMPUTEROWY WROCLAW

DZIEN TYGODNIA	GODZINY ZAJEC	SALA	NAZWA PRZEDMIOTU	RODZAJ ZAJEC	UWAGI
PON	8-10	STYS	HISTORIA MYSLI EKONOMICZNEJ	WYKL	
PON	11-13	JZA.	JEZYKI ZACHODNIE	CWI.	
PON	13-15	SWF.	WYCHOWANIE FIZYCZNE	CWI.	
PON	15-17	A316	POLITYKA EKONOMICZNA I PLANOWANIE	CWI.	
WTO	8-10	PAWI	POLITYKA EKONOMICZNA I PLANOWANIE	WYKL	
WTO	10-11	B302	FINANSE	CWI.	
WTO	11-12	STYS	TEORIA ORGANIZACJI I ZARZADZANIA	WYKL	
WTO	12-13	A205	FINANSE	WYKL	
WTO	13-14	B217	TEORIA ORGANIZACJI I ZARZADZANIA	CWI.	
WTO	14-16	A101	PODSTAWY I ZASTOSOWANIE ETO	CWI.	
SRO	8-10	A106	PODSTAWY I ZASTOSOWANIE ETO	WYKL	
SRO	10-16	WOJS	WOJSKO	WYKL	
CZW	8-10	A129	INSTYTUCJE FINANSOWE	WYKL	
CZW	10-12	RYS.	PIENIADZ I KREDYT W KAPITALIZMIE	WYKL	
CZW	12-13	H415	RACHUNKOWOSC PRZEDS. PRZEMYSLOWYCH	WYKL	
CZW	13-15	H408	RACHUNKOWOSC PRZEDS. PRZEMYSLOWYCH	CWI.	

WROCLAW DZIAL NAUCZANIA	GRUPA 10 SEMESTR ZIMOWY	WROCLAW

DZIEN TYGODNIA	GODZINY ZAJEC	SALA	NAZWA PRZEDMIOTU	RODZAJ ZAJEC	UWAGI
PON	8-10	STYS	HISTORIA MYSLI EKONOMICZNEJ	WYKL	
PON	11-13	JZA.	JEZYKI ZACHODNIE	CWI.	
PON	13-15	SWF.	WYCHOWANIE FIZYCZNE	CWI.	
PON	15-17	A107	POLITYKA EKONOMICZNA I PLANOWANIE	CWI.	
WTO	8-10	PAWI	POLITYKA EKONOMICZNA I PLANOWANIE	WYKL	
WTO	10-12	A129	FINANSE	WYKL	
WTO	12-14	A129	STATYSTYKA MATEMATYCZNA	WYKL	
WTO	14-15	A318	PRZEPLYWY MIEDZYGALEZIOWE	CWI.	
SRO	8-10	A129	TEORETYCZNE PODSTAWY RACHUNKOWOSCI	WYKL	
SRO	11-12	STYS	METODY NUMERYCZNE	WYKL	
SRO	12-13	PAWI	PRZEPLYWY MIEDZYGALEZIOWE	WYKL	
SRO	13-16	A327	TEORETYCZNE PODSTAWY RACHUNKOWOSCI	CWI.	
CZW	8-14	WOJS	WOJSKO	WYKL	
PIA	8-10	H407	EKONOMIKA PRZEMYSLU	CWI.	
PIA	10-12	H407	METODY NUMERYCZNE	CWI.	
PIA	12-14	H410	STATYSTYKA MATEMATYCZNA	CWI.	

AKADEMIA EKONOMICZNA WROCLAW DZIAL NAUCZANIA	T-1 WYKAZ OBCIAZEN SAL DYDAKTYCZNYCH SEMESTR ZIMOWY SALA H408	ZAKLADY NAUKOWO-BADAWCZE WROCLAW

SALA H408 PON GODZ,14-16 GN-2 EOP GRUPA 2 CWI, EKONOMIA POLITYCZNA SOCJALIZMU	SALA H408 PON GODZ,14-16 GN-2 EOP GRUPA 2 CWI, EKONOMIA POLITYCZNA SOCJALIZMU
SALA H408 WTO GODZ, 8-10 GN-2 CEI GRUPA 10 CWI, MARKSISTOWSKA FILOZOFIA I SOCJOLOGIA	SALA H408 WTO GODZ, 8-10 GN-2 CEI GRUPA 10 CWI, MARKSISTOWSKA FILOZOFIA I SOCJOLOGIA
SALA H408 WTO GODZ,10-12 GN-2 EOP GRUPA 2 CWI, MARKSISTOWSKA FILOZOFIA I SOCJOLOGIA	SALA H408 WTO GODZ,10-12 GN-2 EOP GRUPA 2 CWI, MARKSISTOWSKA FILOZOFIA I SOCJOLOGIA
SALA H408 WTO GODZ,13-15 GN-2 EOU GRUPA 6 CWI, MARKSISTOWSKA FILOZOFIA I SOCJOLOGIA	SALA H408 WTO GODZ,13-15 GN-2 EOU GRUPA 6 CWI, MARKSISTOWSKA FILOZOFIA I SOCJOLOGIA
SALA H408 WTO GODZ,15-17 GN-2 EOP GRUPA 2 CWI, STATYSTYKA	SALA H408 WTO GODZ,15-17 GN-2 EOP GRUPA 2 CWI, STATYSTYKA
SALA H408 SRO GODZ, 8-10 IE-2 SPO GRUPA 2 CWI, MARKSISTOWSKA FILOZOFIA I SOCJOLOGIA	SALA H408 SRO GODZ, 8-10 IE-2 SPO GRUPA 2 CWI, MARKSISTOWSKA FILOZOFIA I SOCJOLOGIA
SALA H408 SRO GODZ,11-12 IE-4 CHE GRUPA 7 CWI, FINANSE PRZEDSIEBIOR, PRZEMYSLOWYCH	

SALA H408 SRO GODZ,12-15 IE-5 SPO GRUPA 2 CWI, RACHUNEK KOSZTOW	SALA H408 SRO GODZ,12-15 IE-5 SPO GRUPA 2 CWI, RACHUNEK KOSZTOW	SALA H408 SRO GODZ,12-15 IE-5 SPO GRUPA 2 CWI, RACHUNEK KOSZTOW
SALA H408 SRO GODZ,15-17 GN-4 EK GRUPA 11 CWI, ALGOL	SALA H408 SRO GODZ,15-17 GN-4 EK GRUPA 11 CWI, ALGOL	
SALA H408 CZW GODZ, 8-11 IE-3 SPO GRUPA 2 CWI, ORGANIZACJA PRACY W PRZEM, SPOZYWCZYM	SALA H408 CZW GODZ, 8-11 IE-3 SPO GRUPA 2 CWI, ORGANIZACJA PRACY W PRZEM, SPOZYWCZYM	SALA H408 CZW GODZ, 8-11 IE-3 SPO GRUPA 2 CWI, ORGANIZACJA PRACY W PRZEM, SPOZYWCZYM
SALA H408 CZW GODZ,11-13 IE-3 CHE GRUPA 8 CWI, INZYNIERIA CHEMICZNA	SALA H408 CZW GODZ,11-13 IE-3 CHE GRUPA 8 CWI, INZYNIERIA CHEMICZNA	
SALA H408 CZW GODZ,13-15 GN-3 FIN GRUPA 9 CWI, RACHUNKOWOSC PRZEDS, PRZEMY SLOWYCH	SALA H408 CZW GODZ,13-15 GN-3 FIN GRUPA 9 CWI, RACHUNKOWOSC PRZEDS, PRZEMY SLOWYCH	
SALA H408 PIA GODZ, 8-11 GN-3 EH GRUPA 7 CWI, RACHUNKOWOSC PRZEDS, HANDLO WYCH	SALA H408 PIA GODZ, 8-11 GN-3 EH GRUPA 7 CWI, RACHUNKOWOSC PRZEDS, HANDLO WYCH	SALA H408 PIA GODZ, 8-11 GN-3 EH GRUPA 7 CWI, RACHUNKOWOSC PRZEDS, HANDLO WYCH
SALA H408 PIA GODZ,12-14 GN-3 EK GRUPA 11 CWI, STATYSTYKA MATEMATYCZNA	SALA H408 PIA GODZ,12-14 GN-3 EK GRUPA 11 CWI, STATYSTYKA MATEMATYCZNA	
SALA H408 PIA GODZ,14-16 GN-4 EK GRUPA 11 CWI, PROGRAMOWANIE MATEMATYCZNE	SALA H408 PIA GODZ,14-16 GN-4 EK GRUPA 11 CWI, PROGRAMOWANIE MATEMATYCZNE	

Zał. 12

AKADEMIA EKONOMICZNA IM. OSKARA LANGEGO WROCLAW DZIAL NAUCZANIA	T-2. PLAN ZAJEC DYDAKTYCZNYCH WYDZIAL GN ROK 3 KIER FIN GRUPA 9 SEMESTR ZIMOWY	ZAKLADY NAUKOWO-BADAWCZE OSRODEK KOMPUTEROWY WROCLAW

DZIEN TYGODNIA	GODZINY ZAJEC	SALA	NAZWA PRZEDMIKTU	RODZAJ ZAJEC	UWAGI
PON	8-10	STYS	HISTORIA MYSLI EKONOMICZNEJ	WYKL	
PON	10-11	B308	FINANSE	CWI.	
PON	11-13	JZA.	JEZYKI ZACHODNIE	CWI.	
PON	13-15	SWF.	WYCHOWANIE FIZYCZNE	CWI.	
WTO	8-10	PAWI	POLITYKA EKONOMICZNA I PLANOWANIE	WYKL	
WTO	10-11	B308	TEORIA ORGANIZACJI I ZARZADZANIA	CWI.	
WTO	11-12	STYS	TEORIA ORGANIZACJI I ZARZADZANIA	WYKL	
WTO	12-13	A205	FINANSE	WYKL	
SRO	8-10	A106	PODSTAWY I ZASTOSOWANIE ETO	WYKL	
SRO	10-16	WOJS	WOJSKO	WYKL	
CZW	8-10	A129	INSTYTUCJE FINANSOWE	WYKL	
CZW	10-12	RYS.	PIENIADZ I KREDYT W KA-PITALIZMIE	WYKL	
CZW	12-13	H415	RACHUNKOWOSC PRZEDS. PRZEMYSLOWYCH	WYKL	
PIA	8-10	H404	POLITYKA EKONOMICZNA I PLANOWANIE	CWI.	
PIA	10-12	H406	PODSTAWY I ZASTOSOWANIE ETO	CWI.	
PIA	12-14	H408	RACHUNKOWOSC PRZEDS. PRZEMY...	CWI.	

AKADEMIA EKONOMICZNA IM. OSKARA LANGEGO WROCLAW DZIAL NAUCZANIA	T-2. PLAN ZAJEC DYDAKTYCZNYCH WYDZIAL GN ROK 3 KIER [illegible] GRUPA 10 SEMESTR ZIMOWY	ZAKLADY NAUKOWO-BADAWCZE OSRODEK KOMPUTEROWY WROCLAW

DZIEN TYGODNIA	GODZINY ZAJEC	SALA	NAZWA PRZEDMIKTU	RODZAJ ZAJEC	UWAGI
PON	8-10	STYS	HISTORIA MYSLI EKONOMICZNEJ	WYKL	
PON	10-11	B304	PRZEPLYWY MIEDZYGALEZIO-SE	CWI.	
PON	11-13	JZA.	JEZYKI ZACHODNIE	CWI.	
PON	13-15	SWF.	WYCHOWANIE FIZYCZNE	CWI.	
WTO	8-10	PAWI	POLITYKA EKONOMICZNA I PLANOWANIE	WYKL	
WTO	10-12	A129	FINANSE	WYKL	
WTO	12-14	A129	STATYSTYKA MATEMATYCZNA	WYKL	
WTO	14-16	H403	STATYSTYKA MATEMATYCZNA	CWI.	
SRO	8-10	A129	TEORETYCZNE PODSTAWY RACHUNKOWOSCI	WYKL	
SRO	11-12	STYS	METODY NUMERYCZNE	WYKL	
SRO	12-13	PAWI	PRZEPLYWY MIEDZYGALEZIO-SE	WYKL	
SRO	13-15	H404	EKONOMIKA PRZEMYSLU	CWI.	
CZW	8-14	WOJS	WOJSKO	WYKL	
PIA	8-10	H403	POLITYKA EKONOMICZNA I PLANOWANIE	CWI.	
PIA	11-13	H405	TEORETYCZNE PODSTAWY RACHUNKOWOSCI	CWI.	
PIA	13-15	H405	METODY NUMERYCZNE	CWI.	

AKADEMIA EKONOMICZNA WROCLAW DZIAL NAUCZANIA

T-1 WYKAZ OBCIAZEN SAL DYDAKTYCZNYCH

SAMESTR ZIMOWY

SALA H408

ZAKLADY NAUKOWO-BADAWCZE WROCLAW

SALA H408 PON GODZ.10-11 GN-3 EP GRUPA 2 CWI. FINANSE		
SALA H408 PON GODZ.14-16 GN-4 EK GRUPA 11 CWI. EKONOMETRIA	SALA H408 PON GODZ.14-16 GN-4 EK GRUPA 11 CWI. EKONOMETRIA	
SALA H408 WTO GODZ. 8-10 GN-2 CEI GRUPA 10 CWI. EKONOMIA POLITYCZNA SOCJALIZMU	SALA H408 WTO GODZ. 8-10 GN-2 CEI GRUPA 10 CWI. EKONOMIA POLITYCZNA SOCJALIZMU	
SALA H408 WTO GODZ.10-11 GN-3 EP GRUPA 2 CWI. TEORIA ORGANIZACJI I ZARZADZANIA		
SALA H408 WTO GODZ.13-15 GN-3 EP GRUPA 2 CWI. PODSTAWY I ZASTOSOWANIE ETO	SALA H408 WTO GODZ.13-15 GN-3 EP GRUPA 2 CWI. PODSTAWY I ZASTOSOWANIE ETO	
SALA H408 SRO GODZ. 8-10 GN-2 EOP GRUPA 2 CWI. EKONOMIA POLITYCZNA SOCJALIZMU	SALA H408 SRO GODZ. 8-10 GN-2 EOP GRUPA 2 CWI. EKONOMIA POLITYCZNA SOCJALIZMU	
SALA H408 SRO GODZ.10-12 GN-2 EOU GRUPA 6 CWI. STATYSTYKA	SALA H408 SRO GODZ.10-12 GN-2 EOU GRUPA 6 CWI. STATYSTYKA	
SALA H408 SRO GODZ.12-13 GN-4 EK GRUPA 11 CWI. CYBERNETYKA EKONOMICZNA		
SALA H408 SRO GODZ.14-16 GN-3 EP GRUPA 2 CWI. RACHUNKOWOSC PRZEDS. PRZEMY SLOSYCH	SALA H408 SRO GODZ.14-16 GN-3 EP GRUPA 2 CWI. RACHUNKOWOSC PRZEDS. PRZEMY SLOWYCH	
SALA H408 CZW GODZ. 8-11 IE-5 SPO GRUPA 2 CWI. RACHUNEK KOSZTOW	SALA H408 CZW GODZ. 8-11 IE-5 SPO GRUPA 2 CWI. RACHUNEK KOSZTOW	SALA H408 CZW GODZ. 8-11 IE-5 SPO GRUPA 2 CWI. RACHUNEK KOSZTOW
SALA H408 CZW GODZ.12-14 GN-3 EP GRUPA 2 CWI. POLITYKA EKONOMICZNA I PLANOWANIE	SALA H408 CZW GODZ.12-14 GN-3 EP GRUPA 2 CWI. POLITYKA EKONOMICZNA I PLANOWANIE	
SALA H408 CZW GODZ.14-15 GN-3 EP GRUPA 2 CWI. EKONOMIKA PRZEMYSLU		
SALA H408 PIA GODZ. 8-10 GN-2 PF GRUPA 8 CWI. MARKSISTOWSKA FILOZOFIA I SOCJOLOGIA	SALA H408 PIA GODZ. 8-10 GN-2 PF GRUPA 8 CWI. MARKSISTOWSKA FILOZOFIA I SOCJOLOGIA	
SALA H408 PIA GODZ.10-12 GN-3 EH GRUPA 7 CWI. PODSTAWY I ZASTOSOWANIE ETO	SALA H408 PIA GODZ.10-12 GN-3 EH GRUPA 7 CWI. PODSTAWY I ZASTOSOWANIE ETO	
SALA H408 PIA GODZ.12-14 GN-3 FIN GRUPA 9 CWI. RACHUNKOWOSC PRZEDS. PRZEMY SLOWYCH	SALA H408 PIA GODZ.12-14 GN-3 FIN GRUPA 9 CWI. RACHUNKOWOSC PRZEDS. PRZEMY SLOWYCH	