6 1976

informatyka

POLITECHNIKA ŚLĄSKA BIBLIOTEKA GLIWICE
P. 1877/76

SPIS TREŚCI NR 6/76

str.

Dla siebie budujemy na końcu. Rozmowa INFORMATYKI z dyrektorem Centrum ETOB doc. dr inż. Markiem Grochowskim i mgr inż. Wiesławem Mirowskim
Krystyn Bernatowicz 1

Wpływ algorytmów zwalczania blokad procesów obliczeniowych na efektywność wykorzystania resursów komputera
Jerzy Chmurzyński, Ludwik Urbaniak 2

OSIRIS — pakiet programowy do badań statystycznych
Maciej Ziarkowski 5

Ustalenie procedur kontrolno-weryfikacyjnych w procesie przetwarzania danych
Bernard Buśko 8

TEXTPAC — uniwersalny pakiet programów do przetwarzania danych tekstowych
Stanisława Korytowska 12

System informatyczny przedsiębiorstwa a metody ekonometryczne
Józef Hozer, Grzegorz Klonowski 16

Propozycja rozwiązania podstawowej bazy danych kartotekowych przedsiębiorstwa z zastosowaniem systemu kartotek bazowych
Władysław Wornalkiewicz 19

Wprowadzenie do języka programowania APL (4)
Andrzej J. K. Plewicki 22

Detekcja i lokalizacja defektów w sieciach logicznych
tłumaczył i opracował: *Jan Klimowicz* 26

KADRY INFORMATYKI

EWIKAN 2 — elektroniczny podsystem ewidencji kadr nauczycielskich w Polsce
Bogumił Stachura, Stanisław Urbanek 29

OŚRODKI INFORMATYKI PREZENTUJĄ

Informatyka narzędziem usprawniania organizacji i zarządzania gospodarką miejską województwa gdańskiego
Adam Langer, Jan Nowak 32

Z KRAJU

CYBER 72 pracuje w CYFRONECIE
Krystyn Bernatowicz 34

ZE ŚWIATA

Informatycy na nartach (mh) 28

Konferencja ICCAS 76 w Göteboru (WK) 31

Pamięć o pojemności 1 tryliona bitów (Sob.) 33

25-lecie działalności VUMS (WK) 33

Konferencja CONVERTATION INFORMATIQUE (WK) 37

SZTUKA KOMPUTEROWA

Operacja sumowania
Marek Hołyński 38

PRZEGLĄD WYDAWNICTW

Recenzja książki Teuvo Kohonena: Elementy elektronicznych maszyn cyfrowych
Bolesław Pochopień 39

Bibliografia wydawnictw polskich z dziedziny informatyki (A.K.) 40

TERMINOLOGIA

Uwagi i propozycje do terminologii struktur danych
Jan Kozak III okł.

OGŁOSZENIE IV okł.

WYDAWNICTWA CZASOPISM TECHNICZNYCH NOT
Warszawa
Czackiego 3/5

KOLEGIUM REDAKCYJNE
Redaktor Naczelny prof. dr hab. Leon ŁUKASZEWICZ
mgr Krystyn BERNATOWICZ, prof. dr hab. inż. Konrad FIAŁKOWSKI (zast. redaktora naczelnego), dr inż. Marek HOŁYŃSKI, Władysław KLEPACZ (zast. redaktora naczelnego)
doc. dr hab. Antoni MAZURKIEWICZ, doc. dr inż. Andrzej TARGOWSKI

Sekretarz Redakcji mgr Krystyna WROŃSKA Red. tech. Anna POTARSKA

RADA PROGRAMOWA
Mgr inż. Antoni BOSSOWSKI, doc. dr inż. JAN FELICKI, doc. dr inż. Zbigniew GACKOWSKI, doc. dr inż. Aleksander GOLIŃSKI, dr hab. Andrzej GRZYWAK, doc. dr hab. inż. Roman KULESZA, prof. dr hab. Leon ŁUKASZEWICZ, prof. dr hab. inż. Stanisław PASZKOWSKI (wiceprzewodniczący), prof. dr Tadeusz PECHE, mgr inż. Bronisław PIWOWAR, dr inż. Andrzej PLĄSKOWSKI, mgr inż. Tadeusz PODGÓRSKI (wiceprzewodniczący), prof. dr inż. Jerzy SEIDLER, prof. dr inż. Andrzej STRASZAK (przewodniczący), doc. Jerzy TRYBULSKI, dr Tadeusz WALCZAK, prof. dr inż. Stefan WĘGRZYN, dr inż. Jan Z. ŻYDOWO

Redakcja: 00-941 Warszawa, ul. Jasna 14/16, pokój 331, tel. 27-71-40 lub centrala 26-82-61 w. 285, dyżurny redakcji 10,00—13,00

Zakład Kolportażu WCT NOT, Warszawa, ul. Mazowiecka 12

Zakł. Graf. „Tamka". Z. 2. Zam. 233. Papier druk. sat. IV kl. 70 g. obj. 5,0 ark. druk. Nakład 6.500. J-26.

Cena egzemplarza zł 15.— INDEKS 36210/36124 Prenumerata roczna zł 180.—

Informatyka

zastosowania w gospodarce, technice i nauce

Nr 6
MIESIĘCZNIK
1976
ROK XI
czerwiec

P. 1877/76

ORGAN MINISTERSTWA NAUKI, SZKOLNICTWA WYŻSZEGO I TECHNIKI I POLSKIEGO KOMITETU AUTOMATYCZNEGO PRZETWARZANIA INFORMACJI NACZELNEJ ORGANIZACJI TECHNICZNEJ

Dla siebie budujemy na końcu

Rozmowa INFORMATYKI z dyrektorami Centrum ETOB doc. dr inż. Markiem Grochowskim i mgr inż. Wiesławem Mirowskim

INFORMATYKA: Nie od razu ETOB zbudowano...

Dyr. MIROWSKI: ...historia zaczyna się w 1952 roku, najpierw było Biuro Rozliczeń Budownictwa, którego powstanie datuje się w 1952 r., nieco później powstał przy Zjednoczeniu Biur Projektów — Ośrodek Zastosowań ETO w Projektowaniu (ETOPROJEKT), gdzie rozpoczął pracę pierwszy komputer w budownictwie ZAM 2.

Następny okres — to okres powstania oddziałów Biura, które z czasem (1969 rok) przekształciły się w ETOBY.

INFORMATYKA: Jaki daje to obraz na mapie kraju?

Dyr. MIROWSKI: Obecnie CENTRUM podlegają ETOBY w Warszawie, Bydgoszczy, Gdańsku, Krakowie, Łodzi, Poznaniu i Rzeszowie; poza tym istnieje sieć ośrodków satelitarnych, które z czasem się usamodzielnią — tymczasem korzystają z doświadczeń ośrodków koordynatorów, tj. tych które już wymieniłem, a także czasami z ich komputerów. I tak Ośrodki w Olsztynie i Białymstoku są podporządkowane gdańskiemu, w Kielcach i Tarnowie — krakowskiemu, we Wrocławiu, Wałbrzychu, Piotrkowie Trybunalskim — łódzkiemu, w Szczecinie i Zielonej Górze — poznańskiemu, w Lublinie — warszawskiemu, w Koszalinie — bydgoskiemu. Poza gestią CENTRUM znajduje się dotychczas jedyny ETOB katowicki; sądzę jednak, że już niebawem nastąpi połączenie. Sieć ETOB, o której mówimy integruje się poprzez CENTRUM ETOB na poziomie koordynacji zarządzania zadań i prac projektowych w odniesieniu do problemów resortowych, realizowanych przez Warszawski ETOBSYSTEM, będący integralną częścią CENTRUM.

INFORMATYKA: Wynika z tego, że sieć jest rozległa. Należy sądzić, że są to silne ośrodki?

Dyr. MIROWSKI: Szkopuł w tym, że są to często ośrodki prawie bez locum. Przyzwoite warunki lokalowe posiadają, jak dotąd, Poznań i Wrocław, znośne — Rzeszów i Kielce, reszta — to duże prowizoria...

INFORMATYKA: Budownictwo bez budynków?

Dyr. GROCHOWSKI: Buduje się sporo, ale jakoś tak się składa, że najtrudniej dla siebie. Mamy jednak nadzieję, że teraz się ruszy. Przewidujemy, że jeszcze w tym roku nowe obiekty zostaną oddane do eksploatacji w Lublinie i w Zielonej Górze, a w przyszłym roku w Gdańsku. W końcu musimy gdzieś pomieścić ludzi i komputery.

INFORMATYKA: Jakim sprzętem obecnie dysponujecie?

Dyr. MIROWSKI: 7 komputerów MIŃSK 32, ODRA 1204, 4 komputery ODRA 1304 oraz 8 ODRA 1305; wśród tych ostatnich pierwsza w Polsce ODRA z pamięcią 128 k (w ETOB Warszawa), w tym roku będzie jeszcze większy komputer, też ODRA, lecz z pamięcią 256 k. Pozwoli to myśleć o teledacji ...

INFORMATYKA: Najpierw chyba trzeba wyposażyć Ośrodki?

Dyr. GROCHOWSKI: Przewidujemy, że rocznie będziemy instalowali 5 zestawów ODRA 1305. Jeżeli wygaśnie ich produkcja, wówczas instalować będziemy RIADÓW tyle samo z tym, że dla naszych potrzeb powinny to być najmniej komputery klasy R 32. Sądzę, że to założenie nie będzie kłopotliwe w realizacji. Jeżeli zaś mowa o teledacji, to oczywiście przesłanki do jej planowania są dość mgliste. Do sprzętu krajowego, którego jeszcze nie ma, ustawiła się już duża kolejka, może uda się więc kupić sprzęt za dewizy?

Dyr. MIROWSKI: Tymczasem będziemy wdrażali systemy wielodostępne oparte o transmisję lokalną. W obecnym planie pięcioletnim zamierzamy uruchomić takie systemy w Warszawie, Poznaniu i Gdańsku. Komputerami bazowymi będą ODRY 1305 z dużą pamięcią i bułgarskimi dyskami oraz końcówkami niestety z importu, a jeżeli nasz przemysł dostarczy odpowiednie urządzenia, to krajowymi.

INFORMATYKA: A więc są to sprawy jeszcze odległe?

Dyr. GROCHOWSKI: Tak, chociaż niezwykle pilne. Technika przetwarzania wsadowego jest już dla nas za wolna. Potrzebna nam jest informacja szybka, niech nawet droższa, bo uzyskiwana większym kosztem, ale szybsza.

INFORMATYKA: Taką potrzebę narzuca struktura systemów eksploatowanych w ETOBIE, a więc WEKTORA i AWIZO-MOC

Dyr. MIROWSKI: Przede wszystkim, ale nie tylko ...

Dyr. GROCHOWSKI: Robimy zresztą nowszą wersję WEKTORA, dwuletnia eksploatacja dostarczyła nam materiału do przemyśleń. Chciałbym tu nadmienić, że WEKTOR cieszy się dużym uznaniem ekspertów zagranicznych.

Dyr. MIROWSKI: Najważniejsze jednak zadania w dziedzinie projektowania i programowania będą wynikały z realizacji przez ETOBSYSTEM trzech problemów resortowych i jednego branżowego. Mówiąc bardziej szczegółowo: mamy do opracowania kompleksowy system zarządzania w kombinatach budowy domów, skomputeryzowany system zarządzania w przedsiębiorstwach przemysłowych resortu budownictwa, komputeryzację systemu zarządzania w przedsiębiorstwach budowlanych. Są to problemy resortowe. Problem branżowy — to system informatyczny transportu budownictwa. Nadto do najważniejszych prac, które powinny być rychło zrealizowane, zaliczyć należy również ukończenie systemu banku informacji budownictwa i wybrane systemy sterowania procesami technologicznymi w przemyśle materiałów budowlanych (na pierwszy ogień — sterowanie procesem technologicznym produkcji cementu i szkła).

INFORMATYKA: To są zadania dla ETOBSYSTEMU i zaplecza naukowo-badawczego. A co będą robić pozostałe ETOBY?

Dyr. GROCHOWSKI: Starczy i dla nich. To, że większość zadań do wykonania przypada na ETOBSYSTEM jest wynikiem jego możliwości — skoncentrowaliśmy tam duży potencjał — kadrę 200 projektantów i programistów. A że mamy do wykonania wiele zadań, część z nich będzie rozwiązywana w poszczególnych ETOBACH. Dążymy jednak do tego, aby prace nie dublowały się, natomiast opracowane systemy powinny być maksymalnie powielarne. W związku z tym przeprowadziliśmy dystrybucję tematów na ETOBY przy założeniu, że opracowywać one będą systemy uniwersalne dla całej naszej sieci. I tak np. ETOB Łódź zajmuje się systemem gospodarki materiałowej, ETOB Warszawa — zatrudnieniem i płacami, ETOB Kraków — gospodarką maszynami, ETOB Poznań — gospodarką środkami trwałymi.

INFORMATYKA: Są to zamierzenia potrzebne i spore. Co będzie warunkowało ich wykonanie?

Dyr. MIROWSKI: Mamy wypracowaną przemyślaną koncepcję. Opiera się ona na paru podstawowych założeniach. Jeżeli zrealizowane zostaną podstawowe warunki — zamierzenia te zrealizujemy. Te warunki to głównie: sprzęt i budynki. Sprzęt — to problem naszego przemysłu, budynki — macierzstyego resortu... Oby zasada budowania dla siebie na końcu — została tym razem przełamana.

Z ramienia INFORMATYKI rozmawiał

Krystyn Bernatowicz

JERZY CHMURZYŃSKI
Katedra Maszyn Matematycznych
Wydział Cybernetyki WAT

LUDWIK URBANIAK
Instytut Automatyzacji
Systemów Zarządzania WAT
Warszawa

Wpływ algorytmów zwalczania blokad procesów obliczeniowych na efektywność wykorzystania resursów komputera

Rozwój systemów operacyjnych komputerów cyfrowych, obserwowany na przestrzeni ostatnich 10 lat, charakteryzuje się wzrastającą ilością współbieżnie realizowanych obliczeń [1]. Zasadniczą przyczyną takiego ukierunkowania jest dążenie do zwiększenia współczynników efektywności wykorzystania elementów komputera (np. procesorów fizycznych, pamięci operacyjnej, urządzeń zewnętrznych). Elementy te dalej nazywać będziemy resursami komputera, lub krócej, **resursami.**

Współczynnik efektywności wykorzystania resursu rozumieć będziemy jako stosunek czasu wykorzystania tego resursu podczas wykonywania programów w komputerze do całkowitego czasu realizacji programów. Systemy operacyjne jednokrotnego przetwarzania wsadowego zapewniają sekwencyjne wprowadzenie do komputera i wykonywanie kolejnych programów z tzw. wsadu. W każdym więc momencie pracy komputera wykonywany jest co najwyżej jeden program. Zasadniczym mankamentem tego sposobu organizowania pracy komputera jest to, że resursy, nie wykorzystywane podczas aktualnie wykonywanego programu, pozostają bezczynne przez cały czas wykonywania tego programu.

Mgr inż. LUDWIK URBANIAK ukończył Wydział Cybernetyki WAT w 1975 r. Obecnie pracuje w Instytucie Automatyzacji Systemów Zarządzania WAT. Zajmuje się oprogramowaniem podstawowym komputerów i minikomputerów.

Nowoczesne, tzw. krotne systemy operacyjne, zapewniają wykonywanie więcej niż jednego programu jednocześnie. W tego typu systemach resursy, które nie są wykorzystywane podczas wykonywania jednego programu, mogą być użyte do wykonywania innych programów. W ten sposób dla krotnych systemów operacyjnych wzrasta stopień wykorzystania resursów komputera.

Najnowszą techniką programowania jest programowanie **wieloprocesowe [3, 5].** Przetwarzanie wieloprocesowe pozwala widzieć realizację programów w postaci zbioru współbieżnych procesów obliczeniowych.

Procesy te komunikują się między sobą, synchronizują wzajemnie oraz wykorzystują wspólną pulę resursów komputera [3]. Wykorzystanie resursów przez procesy odbywa się w następujących krokach:

Z Autorem JERZYM CHMURZYŃSKIM zapoznaliśmy Czytelników w nrze 1/76

- żądanie resursu
- wykorzystanie resursu
- zwolnienie resursu.

Nadzór nad wykorzystaniem resursu jest zadaniem systemu operacyjnego komputera. Oddziaływanie procesu na resurs w powyższy sposób nazywać będziemy **rutynowym.** Proces może wykorzystywać resurs dopiero wtedy, gdy w wyniku żądania tego resursu uzyskał do niego dostęp.

Żądanie aktualnie zajętego resursu powoduje opóźnianie procesu do czasu zwolnienia tego resursu. W szczególnym przypadku zdarzyć się może, że rozdział resursów komputera oraz aktualne ich żądania są przyczyną opóźniania dwóch lub większej ilości procesów. Jeżeli nie istnieje możliwość przerwania opóźniania tych procesów w sposób rutynowy, to sytuację taką nazywamy **blokadą** procesów [4]. Blokada procesów trwać będzie tak długo, aż jakaś specjalna, zewnętrzna akcja (systemu operacyjnego lub operatora komputera) nie spowoduje jej rozładowania.

Przyczyną powstania blokad mogą być nie tylko fatalne przypadki rozdziału resursów, ale również niewłaściwe zasady synchronizacji procesów współbieżnych. Zablokowaniu ulec mogą procesy wymieniające sygnały za pomocą zmiennych synchronizujących typu warunek lub semafor [3].

Przykład 1 ilustruje przypadek powstania blokady procesów.

Przykład 1

Rozważmy przypadek równoczesnej realizacji dwóch programów PRO1 i PRO2 w komputerze ODRA 1304/1305. Programy te napisane zostały w języku ALGOL.

```
'PROGRAM' (PRO 1)
'INPUT' 1 = CRO
'OUTPUT' 1 = LPO
'BEGIN'
'INTEGER' K ;
SELECT INPUT (1) ;
'COMMENT' OPERACJA NUMER 1 ;
K : = READ ;
SELECT OUTPUT (1) ;
'COMMENT' OPERACJA NUMER 4 ;
PRINT (K, 3, 0) ;
FREE INPUT ;
FREE OUTPUT ;
'END'
'PROGRAM' (PRO 2)
'INPUT' 1 = CRO
'OUTPUT' 1 = LPO
'BEGIN'
'INTEGER' L ;
L : = 100 ;
SELECT OUTPUT (1) ;
'COMMENT' OPERACJA NUMER 2 ;
PRINT (L, 3, 0) ;
SELECT INPUT (1) ;
'COMMENT' OPERACJA NUMER 3 ;
L : = READ ;
PRINT (L, 3, 0) ;
FREE INPUT ;
FREE OUTPUT ;
'END'
```

Procesy realizacji programów PRO1 i PRO2 nazwiemy odpowiednio P1 i P2.

Zakładając, że komputer dysponuje tylko jednym czytnikiem kart CR oraz jedną drukarką wierszową LP, prześledzimy następującą sekwencję operacji procesu P1 i P2 wykorzystujących resursy podczas wykonywania programów PRO1 i PRO2:

a) operacja numer 1 — proces P1 żąda i uzyskuje dostęp do czytnika kart CR

b) operacja numer 2 — proces P2 żąda i uzyskuje dostęp do drukarki wierszowej LP

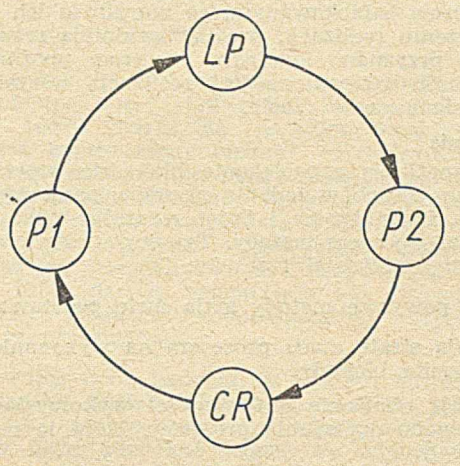
c) operacja numer 3 — proces P2 żąda czytnika kart; ponieważ czytnik jest w posiadaniu procesu P1, żądanie procesu P2 nie zostaje spełnione

d) operacja numer 4 — proces P1 żąda drukarki wierszowej; ponieważ drukarka jest w posiadaniu procesu P2 żądanie procesu P1 nie zostaje spełnione.

Wtrącenie operacji 2 i 3 pomiędzy operacje 1 i 4 może nastąpić np. w wyniku chwilowego braku kart w czytniku CR w momencie realizacji operacji numer 1.

Przedstawiona sekwencja operacji $1 \div 4$ prowadzi do zablokowania procesów P1 i P2. Sytuację powyższą ilustruje graf skierowany, przedstawiony na rysunku 1. W grafie tym wyróżniono dwa rodzaje węzłów:

— węzły resursów CR i LP
— węzły procesów P1 i P2



Rys. 1. Graf blokady procesów P1 r P2

Łuk prowadzący od węzła resursu do węzła procesu oznacza, że proces ten uzyskał dostęp do resursu. Łuk prowadzący od węzła procesu do węzła resursu oznacza, że proces ten żąda wskazany resurs.

Przedstawiony przykład blokady może być rozładowany drogą odebrania jednego z resursów, np. drukarki wierszowej procesowi P2 i przydzielenia jej procesowi P1. Operator komputera ODRA 1304/1305 dokonać może tych czynności za pomocą odpowiednich dyrektyw operatorskich.

Niebezpieczeństwa wynikające z blokad procesów są dwojakiego rodzaju. Po pierwsze, przez czas trwania blokady część resursów nie jest wykorzystywana, przez co pogarszają się współczynniki efektywności wykorzystania tych resursów. Drugim, znacznie groźniejszym, jest przypadek blokad w systemach komputerowych uwarunkowanych czasowo; blokada może być tu przyczyną nadmiernego wydłużania się czasu reakcji systemu komputerowego, a nawet utraty kontroli nad zdarzeniami zachodzącymi w środowisku tego systemu.

Wskazane następstwa blokad procesów wskazują na potrzebę poszukiwania metod ich zwalczania.

METODY ZWALCZANIA BLOKAD PROCESÓW

W zależności od dysponowanej informacji na temat przyszłych żądań resursów przez procesy istnieją różne metody rozwiązywania problemu blokad.

1. Przy posiadaniu informacji o przyszłych żądaniach procesów, wykorzystanie jej do stworzenia sekwencji żądań, która nie dopuszcza do występowania blokady, czyli **unikanie** blokady.

2. Przy braku tej informacji:

a) dopuszczenie do wystąpienia blokady, a następnie jej **detekcja** i usunięcie lub

b) wprowadzenie takich ograniczeń na sposoby żądania resursów przez procesy i ich przyznawania, że blokada nie może wystąpić, czyli **zapobieganie blokadzie.**

We wszystkich metodach do zwalczania blokad służą odpowiednie algorytmy systemu operacyjnego [2, 3, 4, 5].

Poniżej przedstawiono krótką charakterystykę tych metod.

Unikanie:
W momencie każdego zgłoszenia żądania resursów przez proces sprawdzana jest możliwość wystąpienia blokady w

następstwie spełnienia tego żądania. Jeśli wystąpienie blokady jest możliwe, wtedy żądanie nie jest spełnione, a proces jest opóźniony.

Detekcja

W przypadku zastosowania tej metody dopuszcza się możliwość wystąpienia blokady, zapewniając wykrycie jej przez system operacyjny. Usunięcie blokady polega na wywłaszczeniu jednego lub więcej procesów zablokowanych. Wywłaszczeniem nazywamy odebranie procesowi zajmowanych przez niego resursów. Te resursy przydzielone są innym procesom zablokowanym, co umożliwia ich realizację. Po zakończeniu realizacji, procesy zwalniają resursy, które mogą być przyznane kolejnym procesom opóźnionym itd. W ten sposób zapewniona jest realizacja wszystkich procesów w skończonym czasie.

Zapobieganie

Metoda umożliwia zapobieganie blokadzie procesów przez ograniczenie sposobów żądania resursów przez procesy. Procesy mogą żądać typów i ilości resursów tylko w sposób zgodny z algorytmem metody. Typowym rozwiązaniem jest hierarchiczny przydział resursów [3].

Wszystkie powyższe metody mają dwie podstawowe wady:

1. powodują stratę czasu procesora na wykonanie algorytmu zwalczania blokady,

2. powodują obniżenie efektywności wykorzystania resursów, z powodu opóźniania procesów, które te resursy zajmują.

Wybór konkretnego algorytmu powinien być ściśle uzależniony od właściwości systemu, w jakim algorytm ten ma być zastosowany. Obecnie brak jest ogólnej metody określającej przydatność poszczególnych algorytmów zwalczania blokad do pracy w różnych środowiskach [6].

PORÓWNANIE WPŁYWU ALGORYTMÓW UNIKANIA I DETEKCJI BLOKADY PROCESÓW NA EFEKTYWNOŚĆ WYKORZYSTANIA RESURSÓW

W zależności od algorytmu zwalczania blokady, jego wpływ na efektywność wykorzystania resursów będzie różny. W celu ilościowej oceny tego wpływu wprowadzimy **współczynnik nieefektywności wykorzystania resursów WNE.** Okres czasu, w którym ilość resursów zajmowanych przez procesy opóźnione przez algorytm zwalczania blokady pozostaje stały, nazwiemy etapem T_o.

Współczynnik WNE określony jest wzorem:

$$WNE = \frac{\sum_{i=1}^{k} R_{oi} T_{oi}}{R_c T_c}$$

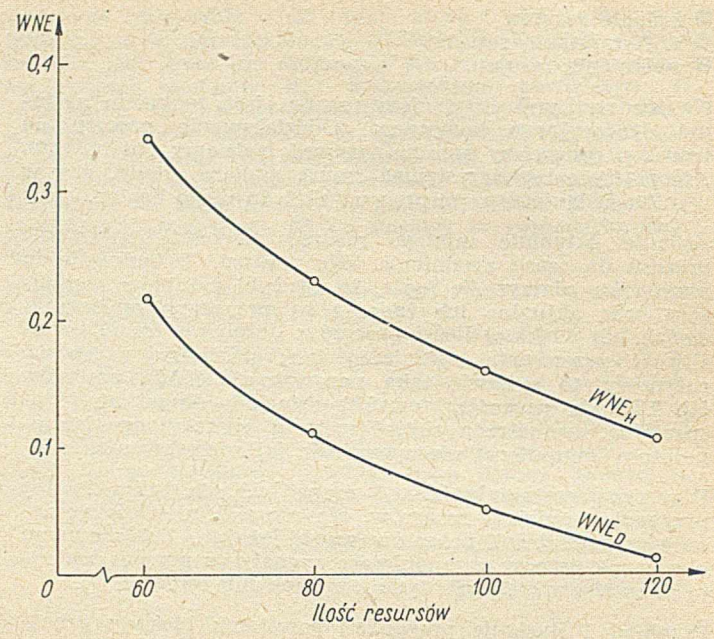
gdzie:

k — ilość etapów
T_{oi} — i-ty etap
R_{oi} — ilość resursów zajmowanych przez procesy opóźnione w etapie T_{oi}
R_c — całkowita ilość resursów w systemie komputerowym
T_c — całkowity czas realizacji wszystkich procesów

W Katedrze Maszyn Matematycznych WAT dokonano porównania wpływu dwóch algorytmów zwalczania blokad na efektywność wykorzystania resursów [7]. Porównano algorytm unikania Habermann'a [2] z algorytmem detekcji Holt'a [4]. Badania dokonano metodą symulacyjną. Zbudowano modele systemu rozdziału resursów, wykorzystując te algorytmy do zwalczania blokad. Modele zaprogramowane zostały w języku GPSS i uruchomione w komputerze ZAM-41. Dla obydwu modeli przyjęto identyczny strumień zgłoszeń żądań resursów przez procesy. Badano zmiany współczynnika nieefektywności WNE w funkcji całkowitej ilości resursów dostępnych w systemie komputerowym. Wyniki symulacji prezentują wykresy przedstawione na rys. 2.

Współczynnik nieefektywności wykorzystania resursów dla algorytmu Habermann'a oznaczono WNE_H, a dla algorytmu detekcji — WNE_D.

Na podstawie powyższych wykresów można stwierdzić, że algorytmy zwalczania blokad powodują obniżenie efektywności wykorzystania resursów w systemie oraz od typu algorytmu. W miarę wzrostu ilości resursów współczynniki nieefektywności maleją, a więc zmniejsza się szkodliwy wpływ algorytmów zwalczania blokad na efektywność wykorzystania resursów. Współczynnik WNE_D jest dla wszystkich przypadków ilości resursów w systemie mniejszy od współczynnika WNE_H. Stąd można wnioskować, że algorytm detekcji pozwala na efektywniejsze wykorzystanie resursów niż algorytm Habermann'a.



Rys. 2. Wykres zmian współczynników nieefektywności wykorzystania resursów

PODSUMOWANIE

Jednym z najistotniejszych problemów, który należy uwzględnić przy projektowaniu współczesnych systemów komputerowych jest ich efektywność. Z tego względu projektant systemu komputerowego powinien znać wpływ stosowanych rozwiązań na ten ważny parametr. Przykładem jest problem blokad.

Obecnie ocena wpływu algorytmów zwalczania blokad opiera się głównie na intuicji projektanta. W niniejszym artykule pokazano, że wpływ ten jest wyraźnie zróżnicowany rodzajem algorytmu zwalczania blokad i możliwy do oceny metodami symulacyjnymi. Pozwala to na porównywanie metod zwalczania blokad i wybór optymalnej dla konkretnego zastosowania. Istnieje więc potrzeba badań kompleksowych w celu pełnej oceny wpływu metod zwalczania blokad na efektywność wykorzystania resursów.

W przyszłych badaniach należy uwzględnić straty czasu procesora spowodowane wykonywaniem algorytmów zwalczania blokad. W szczególnych przypadkach może to być istotny czynnik wpływający na ocenę algorytmu.

LITERATURA

[1] Wybrane zagadnienia systemów operacyjnych, PWN, Warszawa, 1971

[2] Habermann, A. N.: Prevention of system deadlocks. Comm. ACM 12, 7/1969

[3] Hansen, P. B.: Operating system principles. Prentice Hall, 1973

[4] Holt, R. C.: On deadlock in computer systems. Technical report CSRG-6 Toronto 4/1971

[5] Horning, J. J. Randell, B.: Process structuring, Technical report CSRG-15, Toronto, 6/1972

[6] Pasula J.: Ochrona jakości bazy danych w systemach wielodostępnych. INFORMATYKA 12/1975

[77 Urbaniak L.: Algorytmy rozdziału resursów w systemie komputerowym bez blokad. WAT Warszawa 1975.

MACIEJ ZIARKOWSKI
ZETO ZOWAR
Warszawa

OSIRIS – pakiet programowy do badań statystycznych

GENEZA I DZIEDZINY ZASTOSOWANIA PAKIETU

Pakiet programów komputerowych o nazwie OSIRIS (Organized Set of Integrated Routines for Investigation with Statistics), powstał w wyniku współpracy Instytutu Socjologii z Centrum Obliczeniowym Uniwersytetu Michigan (USA).

Proces opracowywania, a następnie ciągłego udoskonalania tego pakietu zapoczątkowano na przełomie lat pięćdziesiątych i sześćdziesiątych, kiedy to do obliczeń statystycznych zastosowano komputery typu IBM-704, IBM-1401 i IBM-7090. Na maszynie serii IBM-360 pakiet OSIRIS po raz pierwszy uruchomiono w maju 1967 r.

W miarę narastania potrzeb szybkiego przetwarzania i analizowania masowych danych statystycznych (np.: opracowywania wyników wyborów), pakiet OSIRIS rozbudowywano o nowe funkcje, wzbogacając jego możliwości zastosowania. W ten sposób wersja pakietu z roku 1973 (znana i uruchomiona w ZETO ZOWAR jako OSIRIS-III) składa się z 60 różnych programów problemowych.

Do wykonania odpowiedniej analizy statystycznej, z maksymalnego zestawu sześćdziesięciu programów należy wykonać w określonej kolejności odpowiednią grupę kilku lub kilkunastu procedur.

Jakkolwiek pakiet OSIRIS projektowano i testowano przede wszystkim pod kątem analizy danych w badaniach socjologicznych, to jednak okazało się, że może on być również wykorzystywany w innych dziedzinach działalności gospodarczej i naukowej. Tak więc może być on wykorzystany dla potrzeb ewidencji personelnej, w pracach administracyjnych i laboratoryjnych, w archiwach itp. Wszechstronność zastosowania pakietu wynika z możliwości elastycznego doboru procedur jego modyfikacji.

ORGANIZACJA ZBIORÓW DANYCH

W naukach socjologicznych typowym i najprostszym źródłem informacji są badania ankietowe, gdzie na ustalony zestaw pytań udziela odpowiedzi określona grupa respondentów. Zestaw odpowiedzi udzielonych przez jednego respondenta stanowi jednostkę analizy, zwaną również obserwacją lub przypadkiem. Tak więc dane uzyskane w wyniku ankiety można przedstawić w postaci tabeli prostokątnej $\{a_{ij}\}$ w której wiersze reprezentują respondentów, natomiast kolumny zawierają pytania lub inaczej mówiąc, zmienne albo cechy. Zatem element tabeli oznaczony przez a_{ij} reprezentuje odpowiedź i tego respondenta na j-te pytanie tak, jak to pokazuje tabela.

Dane przedstawione w postaci tabelarycznej zazwyczaj nie stanowią regularnej tabeli prostokątnej, ponieważ nie wszyscy respondenci mogą udzielić sensownej odpowiedzi na dokładnie ten sam zestaw pytań. Dlatego też ogół respondentów będzie dzielił się na różne klasy, gdzie dla różnych klas będzie różny obszar odpowiedzi. Jednakże względy techniczne analizy takich danych narzucają potrzebę ujednolicenia struktury danych (wszystkie grupy respondentów powinny znaleźć się w jednorodnej klasie), co można uzyskać poprzez wprowadzenie pojęcia braku danych, albo też przez odpowiednie uzupełnienie tych klatek tabeli, które danego respondenta nie dotyczą. W ten sposób dane o strukturze nieprostokątnej uzupełnione są do struktury prostokątnej.

Zmienne / Obserwacje	1-sza zmienna	2-ga zmienna	...	j-ta zmienna	...	n-ta zmienna
n1-szy respondent	a_{11}					
2-gi respondent						
. . .						
i-ty respondent				a_{ij}		
. . .						
ndent -ty respo						

Bywają również takie badania, w których określona grupa respondentów udziela odpowiedzi na kilka różnych zestawów pytań i wówczas może być taka sytuacja, że na jeden zestaw będą udzielane jednakowe odpowiedzi, a na inny zestaw będą udzielane różne odpowiedzi. W tych sytuacjach mamy do czynienia z danymi o strukturze hierarchicznej. Przykładem tego typu danych mogą być informacje uzyskiwane w rezultacie badań nad gospodarstwem domowym, gdzie część danych będzie wspólna dla wszystkich członków rodziny, a inna część tych danych będzie dotyczyła tylko poszczególnych jej członków.

Reasumując powyższe uwagi na temat informacji źródłowych, jak również biorąc pod uwagę stronę techniczną przetwarzania tych informacji przy użyciu komputera, dane te gromadzone są w postaci odpowiednich zbiorów o różnym, ale ustalonym formacie zapisu rekordu, a mianowicie:

- zbiory sekwencyjne o stałym formacie rekordu uporządkowane według respondentów (przypadków)
- zbiory sekwencyjne zawierające różne typy rekordów zmiennej długości dla hierarchicznej struktury danych
- zbiory segmentowo-sekwencyjne o stałym formacie rekordu uporządkowane nie według respondentów a według zmiennych
- zbiory o stałym formacie rekordu zawierającym ustaloną liczbę n elementów macierzy kwadratowej o wymiarach $m \times m$, gdzie $m > n$
- zbiory o stałym formacie rekordu dla gromadzenia elementów macierzy prostokątnej.

Przygotowane na podstawie informacji źródłowych dane wejściowe (czytelne dla komputera) opisywane są za pomocą innego zbioru zwanego słownikiem. Słownik ten jest pomocniczym zbiorem rekordów identyfikujących w sposób unikalny każdą zmienną i jednocześnie opisujący format i położenie danych stanowiących zbiór wartości (argumentów) tych zmiennych. Posługując się nomenklaturą prostokątnej struktury danych (patrz tabela) j-ta kolumna tabeli oznacza zmienną, a rekord opisujący tę zmienną zostanie wprowadzony do słownika, natomiast i-ty wiersz oznacza argumenty zmiennych i w formie jednostkowego zapisu zostanie wprowadzony do zbioru wartości. Zatem

Mgr MACIEJ ZIARKOWSKI ukończył studia w 1965 r. na Wydziale Ekonomiki Produkcji Szkoły Głównej Planowania i Statystyki w Warszawie, a w 2 lata później — Studium Podyplomowe EPD SGPiS. W latach 1967—71 pracował w ETOB Warszawa jako programista i projektant systemów informatycznych. Obecnie — pracując w ZETO ZOWAR na stanowisku gł. analityka — specjalizuje się w zagadnieniach automatycznego wyszukiwania informacji oraz statystycznych systemów informatycznych.

dla wykonania określonej analizy statystycznej potrzebne są dwa zbiory informacji, a mianowicie zbiór zmiennych czyli słownik oraz skojarzony z nim zbiór wartości.

Zbiór słownikowy może być utworzony jedynie przez odpowiednie programy wchodzące w zestaw pakietu OSIRIS, natomiast zbiór wartości może być zakładany bądź w ramach tego pakietu bądź też poza pakietem OSIRIS, to znaczy przez inne systemy informatyczne. Ponieważ za pomocą słownika opisuje się zbiory wartości o różnej strukturze (prostokątna, hierarchiczna) w pakiecie OSIRIS rozróżnia się cztery typy słownika. Słowniki zawierają rekordy stałej długości — 80 znaków, a pierwszy z tych rekordów zawiera ogólne informacje opisujące sam słownik oraz informacje wyróżniające typ zbioru wartości opisywanego tym słownikiem. Następne rekordy przeznaczone są do opisania zmiennych; do opisania jednej zmiennej przeznaczony jest jeden rekord. Ogólny format różnych typów słownika podano poniżej.

Słownik typu — 1 — przeznaczony jest do opisania zbioru wartości o strukturze prostokątnej. Słownik typu — 1 może być utworzony jedynie za pomocą specjalnego programu (File Build) wchodzącego w zestaw pakietu i może on być przechowywany wyłącznie na nośniku pamięci magnetycznej (dysk lub taśma) ze względu na to, że zawiera on pola z wartościami w reprezentacji binarnej. Słownik ten zawiera następujące informacje:

- rekord opisujący słownik

zawier numer pierwszej i ostatniej zmiennej słownika

- rekord główny opisujący zmienną

zawiera numer zmiennej, format danych (alfabetyczne czy numeryczne), typ zmiennej (jedno-odpowiedziowa lub wieloodpowiedziowa), nazwa zmiennej (do 24 znaków alfanumerycznych), początek danych w rekordzie, długość pola z danymi itp.

- rekord pomocniczy opisujący zmienną

przeznaczony jest do rozszerzenia opisu zmiennej i w słowniku występuje opcjonalnie. W słowniku dla opisania jednej zmiennej można użyć szeregu rekordów pomocniczych różnych typów. Typ rekordu rozróżniany jest za pomocą jednoznakowego kodu (np.: Q, K, C. itd.).

Słownik typu — 2 — służy do opisywania tablic elementami, którymi mogą być różnego rodzaju mierniki statystyczne, wynikające ze współzależności dwu cech (np.: współczynnik korelacji).
Słownik typu — 2 zawsze poprzedza dane, które opisuje i w rzeczywistości stanowi on część zbioru wartości. Słownik ten zawiera rekordy trzech typów, a mianowicie:

- rekord opisujący słownik
- rekordy opisujące format karty zawierającej elementy tablicy
- rekordy zawierające etykiety dla zmiennych umieszczanych w wierszach i kolumnach.

Słownik typu — 3 — podobnie jak słownik typu — 1 przeznaczony jest do opisu danych o strukturze prostokątnej, różni się on jednak od słownika typu — 1 tym, że może być przechowywany na kartach dziurkowanych i w takiej formie może on być użyty jako zbiór wejściowy do programu obliczeniowego. Słownik typu — 3 zawiera rekordy o podobnej strukturze, jak słownik typu — 1.

Słownik typu — 4 — przeznaczony jest do opisania zbioru wartości, składającego się z różnych typów rekordów zmiennej długości. Słownik ten stosowany jest więc do opisu danych o strukturze hierarchicznej.

PROCES ZAKŁADANIA ZBIORÓW PODSTAWOWYCH (słownika i zbioru wartości)

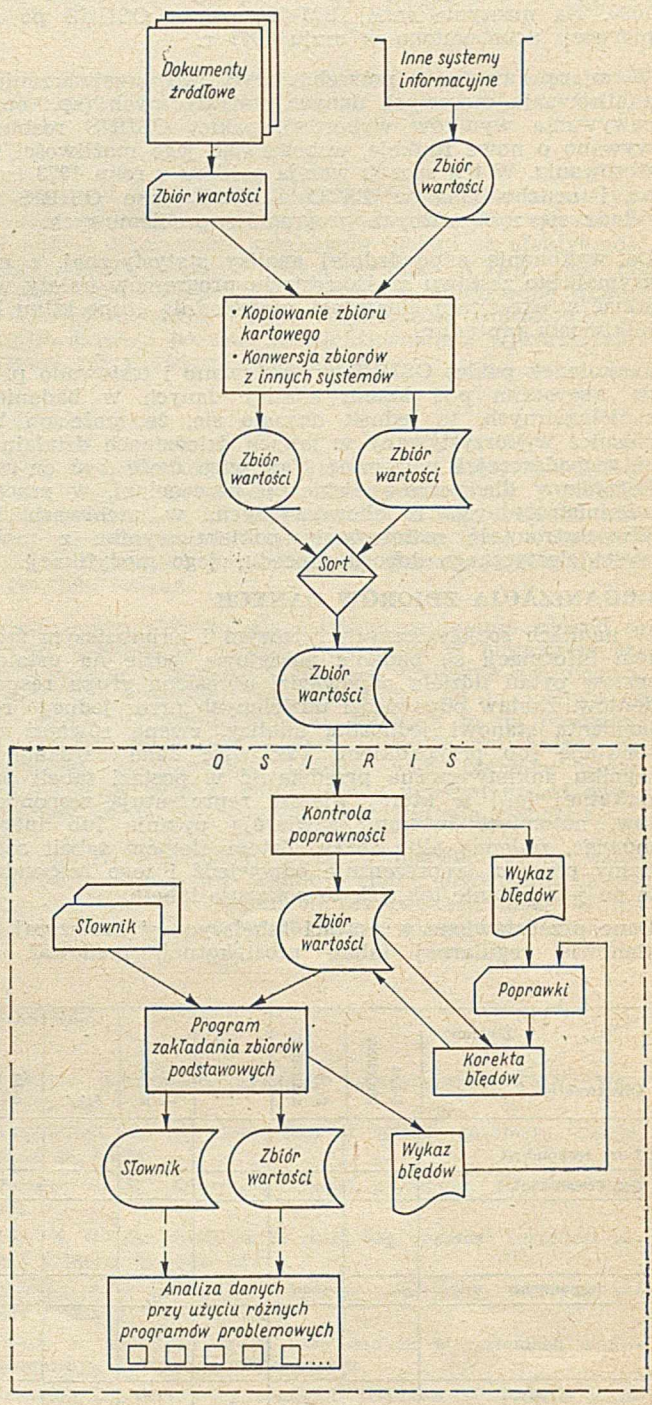
Jak już wcześniej wspomniano, elementem wyjściowym do przeprowadzenia określonej analizy statystycznej w oparciu o programy pakietu OSIRIS jest utworzenie dwóch zbiorów podstawowych, to znaczy słownika i skojarzonego z nim zbioru wartości. Utworzenie tych zbiorów odbywa się automatycznie za pomocą jednego z programów pakietu na podstawie informacji wejściowych przygotowywanych przez użytkownika. Informacje wejściowe do utworzenia słownika przygotowywane są na 80 kolumnowych kartach dziurkowanych w jednej z dwóch form, a mianowicie:
— w formie o ustalonej strukturze karty (fixed form dictionary)
— przy użyciu specjalnych parametrów kluczowych (key form dictionary).

W przeciwieństwie do zbioru słownikowego, który może być utworzony jedynie w ramach omawianego pakietu, zbiór wartości można uzyskać poprzez wykonanie jednego z programów pakietu OSIRIS, względnie można go przejąć jako produkt wyjściowy z innego systemu informatycznego. Maksymalna długość rekordu w zbiorze wartości może wynosić 4000 znaków alfanumerycznych.

Oznacza to, że do opisania jednej obserwacji (respondenta) można użyć 4 tysiące zmiennych (cech) zakładając, że wartość zmiennej będzie jednoznakowa. Jednoznakowy kod kategorii występuje najczęściej wówczas, gdy mamy do czynienia z odpowiedziami typu „tak" lub „nie".

W pakiecie OSIRIS dopuszczalna długość kodu kategorii może przyjmować maksymalnie 9 znaków.

Znając długość poszczególnych pól zawierających kody kategorii oraz ich rozmieszczenie w ramach rekordu zbioru wartości, można przystąpić do kodowania informacji wejściowych do utworzenia słownika, a następnie wykonać proces zakładania zbiorów podstawowych, który schematycznie przedstawiono na rysunku.



Proces zakładania zbiorów podstawowych

Podobnie, jak w oprogramowaniu większości systemów informatycznych, także i w pakiecie OSIRIS — w celu wykonania określonego elementu całego zadania wykonuje się określoną sekwencję czynności, które razem stanowią pewien moduł przetwarzania. Na moduł zakładania zbiorów podstawowych składają się następujące operacje:
1. kopiowanie informacji wejściowych z nośnika kartowego na nośnik magnetyczny z ewentualną konwersją danych
2. uporządkowanie zbiorów danych
3. wykonanie formalnej i częściowo merytorycznej kontroli danych w zbiorze wartości oraz ich skorygowanie
4. utworzenie zbioru słownikowego i zbioru wartości.

Uzyskane w ten sposób dwa podstawowe zbiory stanowią dane wejściowe do dalszego przetwarzania, polegającego na przeprowadzaniu różnych analiz statystycznych przy użyciu jednego lub większej ilości programów problemowych wchodzących w zestaw pakietu OSIRIS.

TECHNIKA KORZYSTANIA Z PAKIETU

Pakiet OSIRIS realizowany jest w trybie partiowym na maszynach cyfrowych serii IBM-360 i IBM-370 pod nadzorem systemu operacyjnego OS. Największe z programów pakietu w zakresie zapotrzebowania na miejsce nie przekraczają 106 k bajtów pamięci operacyjnej. Ponadto pakiet wymaga co najmniej 3 000 ścieżek pamięci roboczej na dyskach magnetycznych. Do wywoływania programów i przypisywania im odpowiednich zbiorów informacji wykorzystywana jest specjalna procedura, składająca się ze zdań języka sterującego wykonywaniem problemów — JCL (Job Control Lanquage).

Programy problemowe pakietu w czasie wykonywania określonego zadania korzystają ze standardowego zestawu kart (zadań) sterujących, za pomocą których do programu wprowadza się odpowiednie parametry. W zależności od rodzaju parametru dostarczonego programowi, na wyjściu tego programu uzyskuje się żądany wynik przetwarzania.

W jednym zdaniu sterującym można umieścić od kilku do kilkunastu parametrów. Spośród całego zestawu kart sterujących (rozróżnia się 10 typów takich kart) do najważniejszych należą:

- karta selekcji globalnej i lokalnej

służy do uzyskiwania podzbiorów wartości.
Przykład:

a. INCLUDE V2 = 23 AND V5 = 6

b. EXCLUDE V10 = 02-05, 09-12 OR VI = 1 — 3

Zdanie (a) oznacza, że do obliczeń mają być włączone tylko te rekordy ze zbioru wartości, z których wartość zmiennej nr 2 równa się 23 i jednocześnie zmienna nr 5 = 6.
Zdanie (b) oznacza, że z obliczeń mają być wyeliminowane te dane, w których zmienna nr 10 przybiera wartość od 2 do 5 lub 9 — 12, lub jeśli zmienna nr 1 przybiera wartość 1 — 3.

- karta parametrów głównych

służy do wprowadzania do programu informacji sterujących wykonywaniem określonych funkcji tego programu. Specyfikowanie parametrów odbywa się za pomocą przypisywania określonym słowom kluczowym ustalonych wartości. Jeśli na karcie parametrów głównych nie umieścimy żadnego z parametrów, wówczas przyjmowane są przez program na zasadzie domniemania te parametry, które dla danego przebiegu są typowe i występują najczęściej. Mówiąc inaczej, do przebiegu przyjmowany jest stały standardowy zestaw parametrów, a tylko wówczas, gdy wykonanie pewnych funkcji programu wydaje się zbędne, można ten standardowy zestaw odpowiednio modyfikować.

Przykład:

PRINT — NODICT

W powyższym przykładzie napis PRINT oznacza słowo kluczowe, a napis NODICT stanowi jeden z parametrów tego słowa, przy czym semantyczne znaczenie parametru NODICT jest takie, że nie chcemy drukować słownika (no dictionary).

- karta ze zdaniami rekodowania

Służy do wykorzystania zasad transformacji danych wejściowych w programach analizy danych. Proces rekodowania polega na uzyskiwaniu nowych zmiennych w wyniku wykonania określonych operacji arytmetycznych i logicznych na zmiennych zdefiniowanych w słowniku. Wyniki rekodowania wykorzystuje się w programach na ogół jednorazowo, chociaż przez użycie niektórych z programów uzyskane rezultaty obliczeń zachować można na stałe przez wprowadzenie ich do słownika i wykorzystywać je w analizach późniejszych. Zasady rekodowania stanowią swego rodzaju system interpretacji języka do transformacji danych wejściowych i w pakiecie OSIRIS stanowią dość obszerne zagadnienie, dlatego też ze względu na charakter niniejszego opracowania nie będą tutaj dokładnie omówione.

OPIS FUNKCJI PAKIETU

Pakiet programów OSIRIS został pomyślany jako elastyczne narzędzie pracy dla ośrodków i instytucji badawczych w dziedzinie nauk społeczno-politycznych. Dlatego też mówiąc o rodzaju i zakresie wykonywanych przez ten pakiet funkcji można by ogólnie stwierdzić, że za pomocą tego pakietu można w zasadzie wykonać wszystkie te obliczenia, które mogą okazać się niezbędne do przeprowadzenia określonego badania, wymagającego posłużenia się współczesnymi metodami statystycznymi. Tak więc w rezultacie wykonywania swych funkcji, pakiet realizuje następujące operacje:

1. w zakresie prezentacji danych statystycznych

- generuje szeregi rodzielcze dla populacji jednowymiarowych
- zestawia tablice korelacji dla populacji wielowymiarowych
- wykonuje punktowy wykres rozsiewu zależności w kartezjańskim układzie współrzędnych

2. w zakresie estymacji (szacowania) parametrów rozkładów cech

- dla populacji jednowymiarowych oblicza: średnią arytmetyczną, wartość modalną, medianę, wariancję, odchylenie standardowe, współczynnik skośności, kwantyle oraz kurtozę
- dla populacji wielowymiarowych oblicza: tekst t dla średnich, statystykę chi — kwadrat $|\chi^2|$, współczynnik kondyngencyjności, V Cramera, współczynnik tau (Goodmana — Kruskalla i Kendalla), współczynniki gamma, współczynniki lambda, odchylenie standardowe, odchylenie normalne, wariancje oraz Rho Spearmana.

3. w zakresie weryfikacji hipotez statystycznych

- testuje różne hipotezy dla populacji p-wymiarowych
- wykonuje analizę wariancji, analizę regresji oraz analizę czynnikową.

4. inne zestawienia i mierniki statystyczne.

Uzyskiwane w rezultacie przetwarzania wyniki wyprowadzone są w formie zbiorów danych pośrednich, względnie w postaci zestawień końcowych. Dane pośrednie stanowią zbiory wejściowe do innego programu lub grupy programów.

Przykładem danych pośrednich może być tablica współczynników korelacji. Wyniki obliczeń końcowych uzyskiwane są głównie w formie zredagowanych tablic jedno lub dwuwymiarowych. Tablice jednowymiarowe zawierają rozkłady liczebności jednej cechy, natomiast tablice dwuwymiarowe podają rozkłady warunkowe dla dwóch zmiennych, jak również rozkłady brzegowe dla tych cech. Liczebności podawane są w liczbach bezwzględnych, jak również istnieje możliwość opcjonalnego obliczania indywidualnych i kumulowanych rozkładów procentowych w wierszach i kolumnach. Oprócz tablic, w zależności od rodzaju wykonywanej analizy statystycznej oraz rodzaju użytego do tego tego celu programu, uzyskuje się również wyniki w innej formie, jak np.: listowanie danych w odpowiednim układzie.

Z punktu widzenia wykonywanych funkcji w systemie, wszystkie programy wchodzące w zestaw pakietu OSIRIS można podzielić na dwie grupy, a mianowicie: programy organizacyjno-manipulacyjne oraz programy analizy danych. Programy grupy pierwszej służą do wykonywania takich operacji jak: zakładanie zbioru słownika i zbioru wartości, tworzenie nowych zmiennych (rozszerzenie słownika), listowanie danych, korekty danych, agregacja danych itp. Za pomocą programów grupy drugiej wykonuje się operacje analizy danych, które polegają na obliczaniu różnych syntetycznych mierników statystycznych omówionych powyżej.

OCENA PAKIETU

Podejmując próbę ogólnej oceny wyżej opisanego pakietu należałoby powiedzieć, że pozytywnym rozwiązaniom towarzyszą również pewne braki.

CZYNNIKI POZYTYWNE:

- pakiet wdrożony i eksploatowany od wielu lat w różnych krajach (niezawodność działania)
- możliwość wykorzystania różnych nośników informacji
- możliwość współpracy pakietu z innymi systemami informatycznymi
- dość szeroki zakres dziedzin zastosowania pakietu
- unifikacja i standaryzacja sterowania wykonywaniem zadań
- możliwość rozszerzania pakietu o nowe procedury.

CZYNNIKI NEGATYWNE:

- standaryzacja w sensie usztywnienia formatu drukowanych wyników niektórych obliczeń
- możliwość wykonywania programów jedynie w trybie partiowym.

Reasumując powyższe uwagi dotyczące oceny pakietu OSIRIS, autor niniejszego artykułu przyznaje, że jest to ocena dość ogólna i subiektywna. Na pełniejszą i pogłębioną ocenę przydatności tego pakietu może złożyć się opinia szerszego grona jego użytkowników, poparta doświadczeniem uzyskanym w wyniku zastosowania do rozwiązywania różnych problemów merytorycznych.

BERNARD BUŚKO
Wojskowa Akademia Techniczna
Warszawa

Ustalanie procedur kontrolno-weryfikacyjnych w procesie przetwarzania danych

ZDEFINIOWANIE PROCEDURY KONTROLNO-WERYFIKACYJNEJ

Prawidłowo zorganizowany proces przetwarzania danych zakłada, że dane rejestrowane, przekazywane między urządzeniami lub komórkami organizacyjnymi, przetwarzane oraz emitowane poza system informatyczny (SI) mogą z określonym prawdopodobieństwem zawierać błędy. Dążąc do zmniejszenia tego prawdopodobieństwa lub wyeliminowania z procesu przetwarzania określonych rodzajów błędów w danych, przewiduje się określone metody wykrycia błędnej sytuacji, jej przeanalizowanie, określenie typu błędu, umiejscowienie go w strukturze danych, a wreszcie natychmiastowe skorygowanie błędu lub zasygnalizowanie błędnej sytuacji w sposób komunikatywny dla człowieka. Kompleks wymienionych tu czynności jest najczęściej łączony w jedną procedurę, zwaną procedurą kontrolno-weryfikacyjną (K-W). Procedura K-W może być samodzielnym elementem procesu przetwarzania albo też może być włączana do określonej innej procedury tego procesu. Jako samodzielna — może występować w operacjach kontroli dokumentów źródłowych, sprawdzenia poprawności przeniesienia danych na maszynowe nośniki danych (MND), a nawet mieć postać autonomicznego programu służącego np. do sprawdzenia poprawności danych zapisanych na MND.

Najczęściej jednak wchodzi ona do innych procedur procesu przetwarzania danych, np. zawarta jest w programie wczytywania danych źródłowych do pamięci zewnętrznej komputera, zakładania lub aktualizacji zbiorów itp.

PODSTAWOWE FUNKCJE PROCEDURY K-W

W zależności od rodzaju popełnionych we wcześniejszych operacjach błędów w danych, od jakości zarejestrowanych danych, od pojawiających się w sposób losowy odmiennych kombinacji rozkładu danych, od pomyłek w czynnościach operatorskich, można w czasie przetwarzania wyróżnić pewną ilość różniących się między sobą sytuacji błędnych.

Z reguły przedmiotem elementarnego przetwarzania nie są wyłącznie pojedyncze dane, lecz pewna ich sekwencja wynikająca ze struktury danych (pozycja w dokumencie, karta dziurkowana, rekord itp.). W tych okolicznościach za sytuacje błędne należy uznawać w szczególności te sekwencje danych, które:
— zawierają dane niezgodne z rzeczywistym pomiarem zdarzeń lub są zniekształcone przez czynniki natury technicznej
— są niekompletne lub zawierają nadmiar informacji z określonego punktu widzenia
— napływają w niewłaściwej kolejności
— są opracowywane w nieodpowiednich terminach
— zawierają wartości liczbowe nie mieszczące się w dopuszczalnych przedziałach
— mają nieodpowiednią formę, układ lub treść informacyjną
— powodują zakłócenia w przebiegu procesu przetwarzania danych lub wręcz uniemożliwiają realizację dalszych jego czynności
— stanowią niezamierzony wynik pracy projektantów, programistów lub operatorów uczestniczących w projektowaniu i eksploatacji SI.

Kolejnym zadaniem procedury K-W po wykryciu błędnej sytuacji jest ustalenie jej przyczyny, którą najczęściej może być błąd w danych, użycie niewłaściwej sekwencji danych (pochodzącej z innych zbiorów), niewłaściwe postępowanie operatora, a w szczególnych przypadkach błędy w programach, algorytmach przetwarzania, instrukcjach oraz chwilowe zakłócenia w pracy urządzeń. Następnie przyczyna ta musi być dokładnie zdefiniowana, a jeżeli jest to błąd w danych lub ich sekwencji, to musi być ustalone umiejscowienie błędnej danej w strukturze. Po tych czynnościach procedura K-W winna określać tryb dalszego postępowania w zależności od rodzaju i charakteru błędu lub nieprawidłowości przetwarzania. W przypadkach, gdy istnieją algorytmy umożliwiające skorygowanie błędów w danych, następuje ich korekta (w tym i automatyczna). W przypadkach użycia niewłaściwych zbiorów, pomyłek w czynnościach operatorskich, powinno się wymuszać wznowienia danej czynności przetwarzania od nowa lub od określonego miejsca z żądaniem:

— użycia właściwych zbiorów
— właściwej reakcji operatora (np. poprawienia błędów w parametrach lub danych sterujących)
— skorygowania błędnych danych
— wykonania innych operacji korygujących.

Dla sytuacji błędnych wynikających z przyczyn technicznych, z zasady wznawia się przetwarzanie od nowa względnie od określonego miejsca w programie (np. podjęcie próby ponownego odczytu lub zapisu danych).

Do często spotykanych sytuacji błędnych należą te, w których przebieg przetwarzania danych, mimo błędów, jest kontynuowany dalej. W tym przypadku sekwencję danych z błędami wyłącza się z dalszego procesu przetwarzania.

Do sporadycznych należy zaliczyć te przypadki, gdy proces przetwarzania jest wstrzymany. Są to sytuacje, w których

brakuje istotnych dla SI danych, a także gdy błędy mogą zniekształcić sens informacji wynikowej lub zostaną one ujawnione w programach, algorytmach i instrukcjach.

W każdym przypadku, fakt ujawnienia sytuacji błędnej, niezależnie czy jest ona naprawialna oraz, czy proces przetwarzania danych jest kontynuowany lub przerywany, musi on być odpowiednio zasygnalizowany. Nie pretendując do pełnego wyczerpania zagadnienia, pod pojęciem sygnalizacji błędnej sytuacji należy rozumieć przedstawienie efektu zastosowanej procedury kontrolno-weryfikacyjnej w komunikatywnej postaci, która:

— w sposób możliwie jednoznaczny i kompletny opisuje sytuację uznaną za błędną lub niewłaściwą
— podaje informacje uzupełniające, umożliwiając przeprowadzenie wyczerpującej analizy i weryfikacji sytuacji błędnej
— podaje decyzje podjęte już przez SI dla danej sytuacji (np. zawieszono w przetwarzaniu dane do wyjaśnienia lub uzupełnienia odpowiednimi danymi, nie uwzględniono w przetwarzaniu lub usunięto ze zbiorów SI itp.).

CHARAKTERYSTYKA PROCEDUR K-W

Procedury kontrolno-weryfikacyjne w zależności od stawianych im zadań mogą być bardzo proste lub złożone.

Procedura prosta może sprowadzać się do wykorzystania jednej metody kontroli lub weryfikacji w czynności kontrolno-weryfikacyjnej. Zastosowanie prostej procedury może występować w przypadku wykrywania jednego typu błędów w danych (np. przy kontroli czy są to dane numeryczne), sprawdzania kompletności zbioru lub dokumentów, sprawdzania prawidłowości użycia właściwych zbiorów wejściowych, kontrolowania prawidłowości użytych parametrów itp.

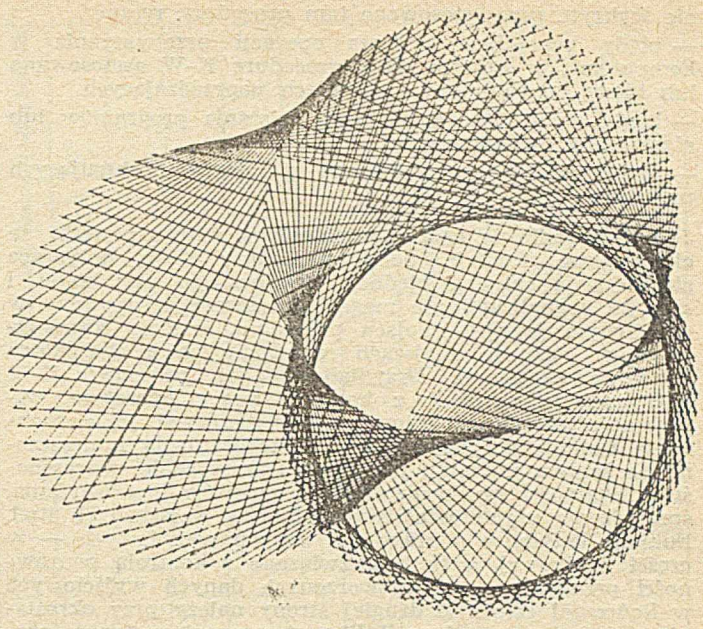
Zadanie wykrycia błędów należących do różnych grup rodzajowych przy jednoczesnej kontroli kompletności danych, prawidłowości realizacji czynności operatorskich oraz innych czynności kontrolno-weryfikacyjnych w jednym nieprzerwanym przebiegu przetwarzania należy do procedur złożonych.

W zależności od stosowanych technik przetwarzania, funkcje procedur K-W w poszczególnych operacjach mogą być realizowane ręcznie lub automatycznie. Procedury ręczne mają zastosowanie głównie przy pomiarze i rejestracji danych, przenoszeniu danych na MND, spedycji dokumentów źródłowych lub MND do ośrodków przetwarzania danych, kompletacji dokumentów lub MND do przetwarzania, weryfikacji i dystrybucji do użytkowników informacji wynikowych. Są one realizowane przez powołane do tego celu komórki.

Procedury K-W realizowane w sposób automatyczny są w większości oparte o podprogramy komputerowe i mogą być wspomagane środkami technicznymi (np. kontrola parzystości, sprawdzanie geometrii rozkładu otworów w karcie dziurkowanej, tablice kodowe dopuszczalnych znaków itp.). Stosowane są one we wszystkich operacjach realizowanych za pomocą komputera.

Spośród przedkomputerowych technik przetwarzania danych (technika ręczna, środki małej i średniej mechanizacji, maszyny licząco-analityczne) jedynie procedury ręczne — w wyniku nagromadzenia wiekowych już doświadczeń — są powszechnie znane i w zasadzie sformalizowane. Natomiast procedury programowe nie są jeszcze wszechstronnie przebadane pod względem ich możliwości, warunków stosowania, skuteczności w działaniu i tym samym nie znajdują one należytego odbicia przy projektowaniu oprogramowania SI. Z przeglądowej analizy opisywanych systemów informatycznych wynika, że najczęstszym ich zastosowaniem jest kontrola poprawności wydziurkowanych danych na MND (a w tym i poprawności rejestracji), kompletności MND składających się na jeden zapis, zgodności z wzorcami (np. występowanie identyfikatora w zbiorze danymi katalogowymi, normatywami), przynależności danych do danego SI, sprawdzanie prostych związków logicznych i arytmetycznych.

Mimo wyszczególnienia szerokiego wachlarza realizowanych funkcji odnosi się wrażenie, że nie wykorzystuje się wszystkich możliwości kontrolno-weryfikacyjnych, a ponadto same procedury K-W sprowadzają się w zasadzie do stwierdzenia faktu wykrycia błędu i określenia, jakkolwiek nie we wszystkich przypadkach, jego charakteru. Wiele zastrzeżeń mogą budzić również sposoby rozwiązania takich funkcji procedury K-W, jak analiza błędnej sytuacji, umiejscowienie błędu, a także sposoby sygnalizacji.



Na taki obraz składa się niewątpliwie wiele czynników. Wynika on z tego, że informatyka jest dziedziną stosunkowo młodą, brak jest doświadczeń i ich uogólnień, brak właściwego ustosunkowania się do problemu wiarygodności danych i jego niedocenienie przy projektowaniu i wdrażaniu SI, jak również i z tego, że „...systemy oprogramowania są najbardziej chyba złożonymi tworami świadomej ludzkiej działalności" [1]. Przyczyn można szukać także pośród podstawowych wad dotychczas stosowanych metod projektowania SI, takich jak:
— „lekceważenie" zagadnień zabezpieczenia niezbędnej funkcjonalnej i technicznej niezawodności systemu
— zbytni optymizm w stosunku do wiarygodności danych wejściowych i wyników maszynowych
— nie przestrzeganie procedur pracy na zbiorach danych
— ukierunkowanie zabezpieczenia organizacyjnego tylko na ustabilizowane warunki funkcjonowania systemu.[2])

W porównaniu z procedurami ręcznymi, procedury programowe są bardzo skutecznym narzędziem w wykrywaniu zarówno błędów w danych, jak i innych nieprawidłowości procesu przetwarzania oraz pozwalają opracować i udostępnić właściwe merytorycznie podstawy do przeprowadzenia korekty błędów w danych i ponownego włączenia ich do procesu przetwarzania. W ten sposób stworzone zostają warunki do podwyższenia poziomu wiarygodności informacji wyjściowych w porównaniu z poziomem informacji wejściowych. Natomiast to, czy warunki te zostaną wykorzystane właściwie, zależy przede wszystkim od człowieka, analizującego błędy i podejmującego określone decyzje odnośnie do sposobu korekty. Podstawowe procedury ręczne mają znacznie mniejszy zakres możliwości. Głównym ich zadaniem jest zapewnienie właściwej jakości operacji wykonywanych w procesie przetwarzania ręcznie. Obejmują one wykrycie i skorygowanie błędów popełnianych w trakcie wykonywania tych czynności. Jeżeli w wyjątkowych przypadkach uzyskuje się podwyższenie poziomu wiarygodności danych, to jest ono nieznaczne, nieprzekraczające w zasadzie wartości jednego rzędu.

Z rozważań tych wynika doniosła rola, jaką w SI mają spełniać programowe procedury K-W. Muszą one wykryć i ewentualnie skorygować:
— błędy popełnione na etapie przetwarzania przedkomputerowego (od pomiaru danych do przygotowania MND) i

---

[1]) W. M. Turski. Projektowanie oprogramowania systemów liczących. W zbiorze Projektowanie maszyn i systemów cyfrowych. PWN. Warszawa, 1972 str. 126

[2]) Osnowy sowremiennoj sistemotechniki. Mir. Moskwa, 1975 str. 7

nie wykryte przez stosowane tam procedury ręczne
— błędy popełnione podczas operacji przetwarzania w komputerze, w którym daną procedurę K-W zastosowano lub błędy popełnione w oepracjach poprzedzających
— błędy wynikające z błędnego działania programów lub algorytmów przetwarzania
— nieprawidłowości w działaniu operatorów obsługujących programy.

Jest to zadanie na ogół skomplikowane z tego powodu, że elementarne funkcje danej procedury K-W, dotyczące stwierdzenia sytuacji błędnej, określenia w niej błędu i sposobu sygnalizacji nie mogą być zawsze zgrupowane w określonym jednym miejscu programu, lecz są rozmieszczone w różnych miejscach wynikających z charakteru działania programu. Przykładowo dla procedury K-W związanej jednocześnie z kontrolą pracy operatora, poprawności danych wejściowych i wyjściowych, jej elementarne funkcje dotyczące pracy operatora będą umieszczone na samym początku programu (sprawdzenie poprawności i legalności wprowadzonych parametrów przetwarzania, sprawdzenie użycia właściwych zbiorów wejściowych itp.). Funkcje dotyczące poprawności danych wejściowych — w części środkowej, a funkcje związane z kontrolą poprawności przetwarzania i wytworzonych danych wyjściowych w końcowej części. Z drugiej strony należy przy określaniu zadań dla procedury K-W umieszczanej w danym programie uwzględniać fakt, że zarówno programy poprzedzające daną czynność, jak i rozpatrywany program mogą zawierać błędy i to w niemałej liczbie [3]). Nakazuje to przewidywać elementarne funkcje kontrolne do wykrycia tego rodzaju błędów, szczególnie w okresie wdrażania SI do eksploatacji, kiedy wykrywa się dużą liczbę błędów w programach i algorytmach (rzędu 60—80% w stosunku do wszystkich elementów SI) [4]).

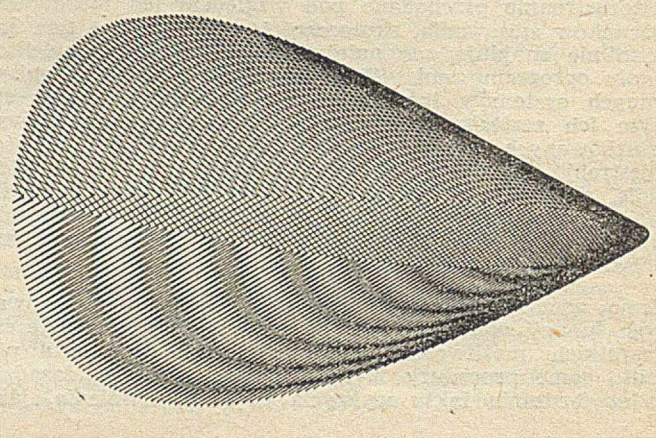
Mamy więc do czynienia z pewnym nadmiarem procedur kontrolno-weryfikacyjnych, które w czasie właściwej eksploatacji są zasadniczo bezużyteczne. Można by je nazwać w pewnym sensie „martwymi", w odróżnieniu od procedur aktywnych, wykrywających sytuacje błędne związane z bieżącym przetwarzaniem danych. Jeżeli w stosunku do aktywnych procedur K-W wymagania mogą być zróżnicowane w zależności od zakładanego poziomu wiarygodności, kompletności i terminowości danych lub od charakteru i skutków błędów, to w odniesieniu do procedur „martwych" wymagania takie muszą być w zasadzie jednakowe i bardzo wysokie. Należy bowiem zawsze pamiętać o konieczności zapewnienia wysokiej niezawodności działania SI. Następstwa ujawnienia błędów w algorytmach i programach w czasie eksploatacji są poważne i zwykle nie można ich usunąć w drodze korekty lub powtórzeń określonych przebiegów procesu przetwarzania. Skorygować tak przekłamane dane można dopiero po wykonaniu korekty oprogramowania i powtórzenia przetwarzania od czynności poprzedzającej fakt wykrycia błędnej sytuacji, lub w przypadkach ekonomicznie uzasadnionych doraźnie zaprojektowanym oprogramowaniem (niezależnie od usuwania przyczyn). Należy się jednak liczyć z tym, że w czasie niezbędnym do usuwania przyczyn powstania i korekty błędów w danych, użytkownik może być pozbawiony dostarczanych przez system informacji.

Charakterystyka procedur K-W byłaby niekompletna, gdyby nie uwzględnić dodatkowej możliwości kontroli zarówno kompletności danych, jak i prawidłowości realizacji operacji procesu przetwarzania. Możliwość ta, nie mieszcząca się w ramach zadań procedur K-W polega na tym, że po każdym działaniu programu jest lub powinien być emitowany arkusz przetwarzania [5]), zawierający nie tylko dane charakteru ewidencyjno-statystycznego i sterującego, lecz także dane charakteryzujące ilościowo i jakościowo dane przetworzone, wczytane przez program lub zapisane do nowych zbiorów. Konfrontacja dwu lub kilku arkuszy sąsiadujących ze sobą operacji, pozwala stwierdzić, czy użyto właściwe zbiory, czy liczba danych wytworzona przez program P_n została uwzględniona w przetwarzaniu przez program P_{n+1} itp. Można w ten sposób stwierdzić brak uchybień w organizacji przebiegu przetwarzania oraz mieć pewność, że wszystkie dane zostały przetworzone zgodnie z założonym algorytmem. W szczególnych sytuacjach dane wytwarzane na potrzeby arkusza przetwarzania mogą być wykorzystane również i przez procedury K-W.

CZYNNIKI WPŁYWAJĄCE NA JAKOŚĆ PROCEDUR K-W

Znaczenie, jakie będą miały procedury K-W w zakresie zabezpieczenia właściwego poziomu wiarygodności, kompletności i terminowości danych zależy przede wszystkim od sposobu podejścia do ich zaprojektowania. Projektowanie to powinno się rozpoczynać już na etapie założeń, przez określenie generalnych ustaleń dla systemu kontroli wiarygodności danych, poprzez etap opracowania projektu technicznego (sprecyzowanie założeń na procedury K-W i sposób ich rozmieszczenia, ogólne wymagania do wszystkich programów i instrukcji eksploatacyjnych), a na projektowaniu programów kończąc. Do opracowania szczegółowych założeń procedur K-W danymi wyjściowymi powinny być:
— wymagany poziom wiarygodności danych dla SI dla wymagania w zakresie niezawodności
— charakterystyka źródeł powstawania i jakości tych danych
— jakość pracy urządzeń przewidywanych do wykorzystania w SI oraz charakterystyka wymaganych cech osobowych personelu obsługi i projektowania SI
— wyniki analizy pozwalającej zlokalizować źródła powstawania błędów, określić charakter i ewentualne skutki tych błędów, dokonać ich klasyfikacji i ustalić tryb postępowania korygującego
— ogólne wymagania dotyczące sposobu korekty danych błędnych przez same procedury K-W, zasady sygnalizacji błędnych sytuacji oraz ewentualne punkty przewidziane do włączania skorygowanych danych do procesu przetwarzania.

Szczegółowe założenia na procedury K-W określonego programu, powinny być opracowane nie później niż założenia programu oraz powinny stanowić część integralną tych założeń. Przy opracowaniu tych założeń, oprócz uwzględnienia wymagań określonych w projekcie technicznym, należy:
— przeanalizować dokładnie założenia programu w celu znalezienia elementów kontrolno-weryfikacyjnych koniecznych do uwzględnienia w procedurze K-W (np. możliwości oprogramowania w zakresie operowania zbiorami — otwarcie, zamknięcie zbioru; etykiety początku, końca zbioru)
— przeanalizować możliwość wystąpienia i wykorzystania wyników funkcji kontrolnych komputera (środkami technicznymi), jego oprogramowania standardowego (system operacyjny, makroinstrukcje itp.), języków programowania oraz sposobu sygnalizacji i przekazywania sterowania do programu
— dokładność analizy charakteru ujawnianych sytuacji błędnych, określić możliwości skorygowania i wznowienia realizacji programu



[3]) W. M. Turski w cytowanej uprzednio pracy (str. 130) podaje, że działające złożone systemy oprogramowania zawierają około jednego błędu na 2—3 tys. instrukcji.

[4]) Por. P. G. Gaganow, W. W. Lipajew, W. A. Wołkow, W. N. Proswiryn — Analiz oszibok w słożnych programmach sistem uprawlenija obiektami. Uprawlajuszczije sistemy i maszіny, 1973 Nr 3 str. 34—35.

[5]) Szczegółową charakterystykę arkusza przetwarzannia czytelnik może poznać w artykule B. Buśko, J. Śliwieński: Funkcje arkusza przetwarzania w procesie wdrażania i eksploatacji systemów informatycznych. Wojskowy Przegląd Organizacji i Informatyki. 1975 Nr 4 str. 72—81.

TABELA

Przyczyny powodujące ujawnienie sytuacji błędnej / Czynności procesu przetwarzania danych	Pomiar danych	Rejestracja (kontrola)	Przekazywanie (spedycja) danych	Przenoszenie danych na MND (kontrola)	Wczytanie danych z częściową kontrolą	Wczytanie danych z pełną kontrolą	Kontrola danych	Sortowanie	Dobieranie	Łączenie	Rozdzielanie	Aktualizacja zbiorów	Przetwarzanie (obliczanie)	Opracowanie danych do wydruku	Wydruk danych z opracowaniem	Wydruk danych bez opracowania	Weryfikacja wydruków
1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18
Niewłaściwa wartość danej lub symbolu (przekłamania znaku, przestawienia itp.)	×	×		×	×	×	×					×	×	×	×	×	×
Dokument nieczytelny, niewłaściwie wypełniony Niewłaściwa symbolizacja		×		×													×
a) dokumentu		×	×	×													×
b) maszynowego nośnika danych				×	×	×	×										
c) identyfikatora SI		×	×	×	×	×	×										
Brak lub zagubienie a) dokumentu		×	×	×													×
b) maszynowego nośnika danych			×	×	×	×	×										
c) określonej danej	×	×		×	×	×	×					×	×	×	×	×	×
d) pozycji w dokumencie		×		×	×	×	×		×			×					
e) rekordu					×	×	×	×	×	×	×	×	×	×	×	×	
f) zbioru danych					×	×	×		×	×	×	×	×	×	×	×	×
Nadmiar lub zwielokrotnienie a) dokumentów		×	×	×													×
b) pozycji w dokumencie		×		×	×	×	×	×				×					
c) maszynowych nośników danych					×	×	×	×				×					
d) rekordów								×	×	×		×	×	×	×	×	
Niewłaściwa struktura danych a) w dokumencie		×		×	×	×	×	×				×					×
b) na maszynowym nośniku danych					×	×	×					×					
c) w rekordzie								×	×	×	×	×	×	×	×	×	
d) brak uporządkowania zbioru									×	×	×	×	×	×	×	×	
Przeterminowany lub niewłaściwy a) dokument		×	×	×													×
b) maszynowy nośnik danych					×	×	×					×	×				
c) rekord lub zbiór danych							×	×	×	×	×	×	×	×	×	×	
Niewłaściwy charakter znaku	×	×		×	×	×	×										
Wartość danej a) nie przyjmuje zadanej wielkości		×		×	×	×	×					×		×	×	×	
b) nie mieści się w przedziale					×	×	×					×					
c) nie jest jedną z zadanego zbioru					×	×	×					×		×	×	×	
Niezgodne lub niespełnione a) suma kontrolna					×	×	×										
b) liczba kontrolna w symbolu					×	×	×										
c) zależność arytmetyczna					×	×	×					×	×				
d) zależność logiczna					×	×	×					×	×				
Niedopuszczalne działania a) próba założenia nowego rekordu przy istniejącym										×		×					
b) brak rekordu aktualizowanego												×					
c) rekord aktualizujący nie spełnia warunków formalnych i merytorycznych												×					
d) sprzeczność w danych dla wykonania operacji maszynowych					×	×	×					×	×	×	×	×	
e) sprzeczność w danych do wykonania założonego algorytmu												×	×	×	×	×	
Błędy operatorskie (dane sterujące, parametry, niewłaściwe zbiory itp.)					×	×	×	×	×	×	×	×	×	×	×	×	
Błędy techniczne i oprogramowania standardowego z przekazaniem sterowania					×	×	×	×	×	×	×	×	×	×	×	×	

— ustalić możliwie prawdopodobne sytuacje błędne, jakie mogą wystąpić w programie wskutek błędnych danych, błędnego przetwarzania, błędów operatorskich, błędów w tym samym lub poprzedzającym go programie itp.

Właściwie zaprojektowane procedury K-W powinny w zasadzie stanowić dość szczelny filtr dla błędów popełnionych wcześniej i błędów operatorskich oraz umożliwić skuteczną analizę błędów, efektywną ich korektę i ponowne włączenie danych do procesu przetwarzania.

Traktując powyższy artykuł jako przyczynek do określe-

nia znaczenia procedur K-W w problemie zabezpieczenia wiarygodności danych należy stwierdzić, że nie zawiera on wystarczających podstaw metodyczno-merytorycznych do ich projektowania.
Uwarunkowane jest to:
— dużym zróżnicowaniem organizacji procesu przetwarzania danych w komputerze, zwłaszcza dla jego początkowego stadium, tj. zakładania i aktualizacji zbiorów, gdzie obecność procedur K-W jest niezbędna i w istotny sposób wpływa na poziom wiarygodności informacji na wyjściu SI
— trudnością jednoznacznego sklasyfikowania procedur K-W według wykrywanych sytuacji błędnych oraz jednoznacznego przyporządkowania ich do poszczególnych operacji przetwarzania danych (np. błąd w symbolu indeksowym można wykryć przy wczytywaniu danych do pamięci komputera, w czasie aktualizacji, a także dopiero przy wydruku lub kontroli wyników przetwarzania)
— trudnością jednoznacznego umiejscowienia identycznej procedury K-W (np. procedurę kontroli uporządkowania danych)
— koniecznością dublowania procedur K-W wynikającą z założenia, że poprzednie operacje przetwarzania mogą nie tylko nie wykryć wszystkich błędów, ale nawet wprowadzić nowe.

O stopniu złożoności tego problemu świadczy zawartość tabeli, w której przedstawiono próbę pokazania możliwości ujawnienia sytuacji błędnych w określonych funkcjach procesu przetwarzania danych.

STANISŁAWA KORYTOWSKA
Biblioteka Narodowa
Ośrodek Przetwarzania Danych
Warszawa

TEXTPAC — uniwersalny pakiet programów do przetwarzania danych tekstowych

Pakiet programów TEXTPAC został opracowany w 1966 r. w USA dla potrzeb konkretnego systemu gromadzenia i wyszukiwania informacji. Pakiet ten składa się z 31 programów przetwarzania danych tekstowych (patrz tabela).

Przetwarzanie to obejmuje podstawowe rodzaje opracowania tekstów informatycznych i stąd głównymi użytkownikami tego pakietu są biblioteki i ośrodki inte.

TEXTPAC umożliwia:
- tworzenie i aktualizowanie bazy danych tekstowych
- wyszukiwanie z posiadanej bazy informacji w ujęciu bieżącym (selektywnym) oraz retrospektywnym
- generowanie zestawień wprowadzonych tekstów w postaci biuletynów oraz opartych na nich indeksów.

Wykorzystanie pakietu TEXTPAC do wyszukiwania w ujęciu bieżącym było już opisane w INFORMATYCE [1].

Dla zainteresowanych wykorzystaniem tego pakietu w Polsce warto dodać, że wchodzi on w zakres oprogramowania standardowego komputerów serii IBM · 360/370 i dlatego jego eksploatacja nie jest związana z ponoszeniem stałych opłat abonenckich na rzecz tej firmy, jak to jest np. w przypadku pakietu STAIRS [2], który jest tzw. pakietem licencyjnym.

Typy informacji wejściowych i wyjściowych

Źródłem danych wejściowych są teksty odpowiednich dokumentów (opisy bibliograficzne publikacji, literatury firmowej, norm, patentów, prac naukowo-badawczych itp.).

Dane zasadniczo wprowadza się na kartach dziurkowanych, jakkolwiek można je wprowadzać również bezpośrednio na dysk lub taśmę magnetyczną. Teksty przenosi się na nośniki maszynowe bez zmian, dodając do nich następujące dodatkowe informacje kontrolne:

- 12-znakowy identyfikator (numer systemowy) dokumentu
- 3-znakowy kod paragrafu (pole kontrolne wydruku).

Paragrafy dotyczą takich elementów opisu dokumentu, jak: autor, tytuł, instytucja, analiza dokumentacyjna, hasła klasyfikacji przedmiotowej itp.

Stąd zapis na jednej karcie dziurkowanej musi mieć formę przedstawioną na rys .1.

Przygotowane w taki sposób dane przenoszone są następnie na taśmę magnetyczną (program TRC 202). W czasie przenoszenia następuje przeformatyzowanie danych oraz

Rys. 1

opisanie każdego występującego w dokumencie [1]) słowa pod kątem jego lokalizacji kontekstowej, długości itp.

Użytkownikowi praktycznie pozostawiono swobodę wyboru liczby elementów danych i ich struktury, tzn. podziału opisu dokumentu na wspomniane paragrafy. Ma on jednak obowiązek stosowania jednolitej formy elementów opisu wprowadzanych do poszczególnych paragrafów, a określony paragraf musi zawierać we wszystkich dokumentach ten sam typ informacji. Jest to szczególnie ważne dla paragrafów o kodach 090, 200, 300, 600, ponieważ wg nich opracowywane i drukowane są indeksy. Wiąże

[1]) Słowo w znaczeniu wyrazu języka naturalnego.

Mgr inż. STANISŁAWA KORYTOWSKA ukończyła studia na Wydziale Elektroniki Politechniki Warszawskiej w 1965 r. Od 2 lat pracuje jako starszy projektant systemów przetwarzania danych i zajmuje się m.in. adaptacją standardowego oprogramowania IBM do przetwarzania informacji tekstowych, najpierw w IINTE, a obecnie w Bibliotece Narodowej.

TABELA 1. Zestawienie programów pakietu TEXPAC

Lp.	Symbol programu	Nazwa programu	Pełnione funkcje
1.	TRC 202	EDIT	przekształca dane wejściowe na format wymagany przez system
2.	TRC 260	360 CONDENSED TEXT EDIT	przetwarza tekst skondensowany stanowiący wynik programu TRC 210
3.	TRC 250	EDIT MERGE	łączy zapisy dwóch zbiorów EDIT TEXT (głównego i ze zmianami) w jeden zbiór uporządkowany sekwencyjnie
4.	TRC 203	EDIT PRINT	drukuje rekordy tekstu (wyjście z TRC 202 lub 260) albo wykryte niezgodności (wyjście z TRC 207)
5.	TRC 208	SEARCH FOR MISSING ITEMS	sprawdza kompletność danych tekstu wejściowego
6.	TRC 210	EDIT CONVERT	przetwarza zbiór EDIT TEXT na jeden lub więcej następujących zbiorów: 1. skondensowany tekst 360 2. rekordy wyszukiwawcze 3. rekordy częstotliwości występowania słów (WORD FREQUENCY)
7.	TRC 052	OMAHA TO DICTIONARY	generuje rekord dla każdego słowa ze zbiorów danych, które występuje 5 lub więcej razy
8.	TRC 204	DICTIONARY UPDATE	aktualizuje (lub tworzy) słownik w porządku alfabetycznym oraz umożliwia listowanie
9.	TRC 205	LENGHT (ALPHA SORT	sortuje rekordy słownika wg długości słowa
10.	TRC 206	DICTIONARY BLOCKING	grupuje posortowane rekordy słownika w bloki o długości do 1000 bajtów
11.	TRC 211	WORD FREQUENCY SORT	sortuje zbiór wyjściowy z programu TRC 210 standardowy program systemu operacyjnego OS/360
12.	TRC 212	WORD FREQUENCY OMAHA	przygotowuje listę występowania poszczególnych słów oraz liczbę dokumentów zawierających każde z tych słów
13.	TRC 207	DICTIONARY COMPARE	sprawdza słowa tekstu wejściowego przez ich porównanie ze słownikiem operacyjnym (wyjście z TRC 206)
14.	TRC 022	BULLETIN PRINT	drukuje biuletyn oraz przygotowuje indeksy
15.	TRC 020	INDEX SORT	sortuje indeksy
16.	TRC 021	INDEKS PRINT	drukuje posortowane indeksy
17.	TRC 001	CIS PROFILE UPDATE	tworzy zbiór profilu lub go aktualizuje
18.	TRC 002	CIS PROFILE DIAGNOSTIC	sprawdza poprawność formatu i logiki pytań
19.	TRC 010	CIS MEMORY LOAD	przetwarza poprawne profile na tablice wewnętrzne
20.	TRC 011	CIS SEARCH ANSWER	tworzy zbiór zawierający nazwiska (nazwy instytucji) i adresy związane z odpowiednimi profilami
21.	TRC 012	CIS ANSWER INVERSION	porządkuje zbiór odpowiedzi i łączy go ze zbiorem nazw i adresów
22.	TRC 013	CIS DISK LOAD	powoduje powstanie zbioru odpowiedzi
23.	TRC 221	RETRO QUESTION DIAGNOSTIC	sprawdza poprawność formatu i logiki pytań
24.	TRC 222	RETRO MEMORY LOAD	przetwarza poprawne pytania na tablice wewnętrzne
25.	TRC 223	RETRO SEARCH	tworzy dwie kartoteki, z których jedna zawiera numery dokumentów i pytań łącznie z danymi statystycznymi; natomiast druga-numery pytań oraz rekordy skondensowanego tekstu dokumentów stanowiących odpowiedzi
26.	TRC 251	RETROSPECTIVE MERGE	przygotowuje zbiory wyszukiwania retrospektywnego przez powiązanie zapisów tekstu skondensowanego i zapisów wyszukiwawczych
27.	TRC 252	RETROSPECTIVE MASTER MERGE	łączy przygotowane programem TRC 251 rekordy ze starymi rekordami wyszukiwania retrospektywnego
28.	TRC 227	RETROSPECTIVE TEXT EXPANSION	tworzy listę numerów pytań, dla których zostały znalezione odpowiedzi. Zapisuje tekst dokumentu będącego odpowiedzią tyle razy, ile wystąpił na w/w liście
29.	TRC 228	RETROSPECTIVE TEXT SORT	sortuje zbiór wyjściowy z programu TRC 227 wg numerów pytań
30.	TRC 229	RETROSPECTIVE	drukuje pytania i wyszukane odpowiedzi
31.	TRC 226	RETROSPECTIVE STATISTICAL	drukuje informacje statystyczne o wykorzystaniu Systemu

się to niejednokrotnie z potrzebą znormalizowania skrótów odpowiednich elementów opisu (np. jednolite skróty nazw języka, kraju, instytucji itp.). Konieczność wprowadzenia skrótów wynika z narzuconego przez system formatu zapisu danych wejściowych. Paragrafy indeksowe są polami stałej długości i wprowadzone do nich informacje nie mogą przekroczyć długości 69 znaków (1 wiersz). Pozostałe paragrafy tworzą pola zmiennej długości o maksymalnej długości do 54 wierszy (w jednym wierszu maksymalnie 16 słów).

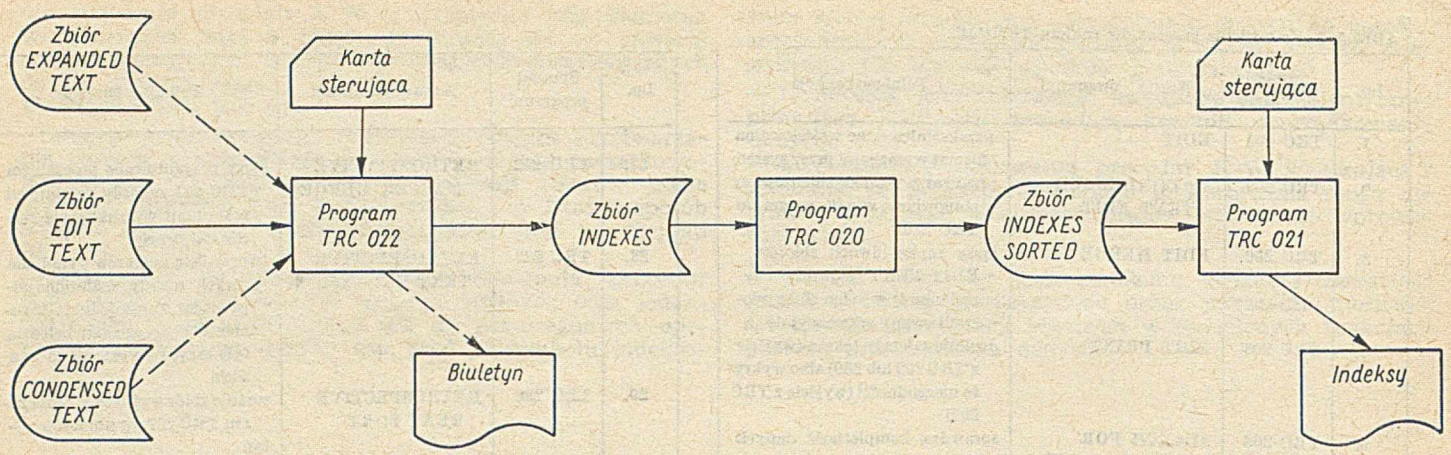
Ograniczenie długości paragrafów indeksowych udało się rozszerzyć w wyniku opracowanej w ZETO ZOWAR modyfikacji programów standardowych. Rozszerza to znacznie możliwości wykorzystania pakietu. Format opisu dokumentów wejściowych w pakiecie TEXTPAC charakteryzuje się więc dużą prostotą oraz jest wygodny do kontrolowania wprowadzanej informacji.

Przykładem zestawu elementów opisu dokumentu wejściowego może być opis podany w cytowanym artykule [1] dla informacji chemicznej na taśmach Chemical Abstracts Condensates.

TEXTPAC rozporządza podobnie elastycznymi możliwościami określania treści dokumentów wynikowych. Można uzyskać następujące informacje wynikowe:
— cały opis dokumentu
— oddzielne pola danych (poszczególne elementy opisu)
— biuletyn oraz do 5 różnych indeksów
— odpowiedzi na zadane pytania (profile).
Biuletyn będący wydrukiem zawartości opisów wprowadzonych do bazy danych jest uporządkowany sekwencyjnie wg identyfikatorów opisu dokumentów. Równolegle z biuletynem drukowane są indeksy. W programie tworzenia biuletynu przewidziano wiele możliwości, np.:
— tworzenie biuletynu z restartem, tzn. rozpoczynanie druku od określonego identyfikatora
— drukowanie pełnych opisów dokumentów albo określonej ich części, tzn. określonego paragrafu, który podaje się na karcie parametrycznej wydruku.
Również w przypadku drukowania indeksów pakiet pozostawia dowolność w postaci alternatyw drukowania albo wszystkich indeksów, albo też wyłącznie określone w karcie parametrycznej.
Schemat tworzenia biuletynu i indeksów przedstawia rys. 2.

AKTUALIZACJA DANYCH

TEXTPAC dysponuje bardzo bogatym aparatem aktualizacji bazy danych. Istnieje możliwość wprowadzania następujących zmian:
— usunięcia całego tekstu danej pozycji
— usunięcie tekstu poczynając od oznaczonego w karcie korekty paragrafu
— usunięcie jednego określonego paragrafu
— zamiana wiersza tekstu
— dodanie wiersza tekstu
— usunięcie wiersza tekstu
— zamiana jednego lub więcej słów tekstu
— usunięcie słowa
— dodanie słowa.



Rys. 2. Tworzenie biuletynu i indeksów

Zmiany te podaje się na kartach aktualizacji podczas wprowadzania programu TRC 202 (rys. 3).

System ma również silnie rozbudowaną część diagnostyczną do wykrywania przyczyn błędów.

OCHRONA DANYCH

System operacyjny każdego komputera przewiduje środki ochrony przechowywanych w pamięci zbiorów. Z reguły dane zabezpiecza się przed zniszczeniem na skutek awarii urządzeń i nieumiejętności obsługi.

Pakiet TEXTPAC posiada oprócz tego własne zabezpieczenia przed utratą danych oraz dostępem do nich osób niepowołanych. Polegają one na wprowadzeniu tzw. kodów tajności umieszczonych w paragrafie 080, które pozwalają zaliczyć oznaczone tym kodem zapisy do grupy zastrzeżonych lub tajnych. Manipulowanie zbiorami takich danych wymaga od użytkownika każdorazowego sformułowania i przekazania do systemu informacyjnego specjalnych poleceń.

ZBIORY DANYCH (KARTOTEKI) PAKIETU

TEXTPAC działa w oparciu o wiele zbiorów danych, z których najważniejszymi są:

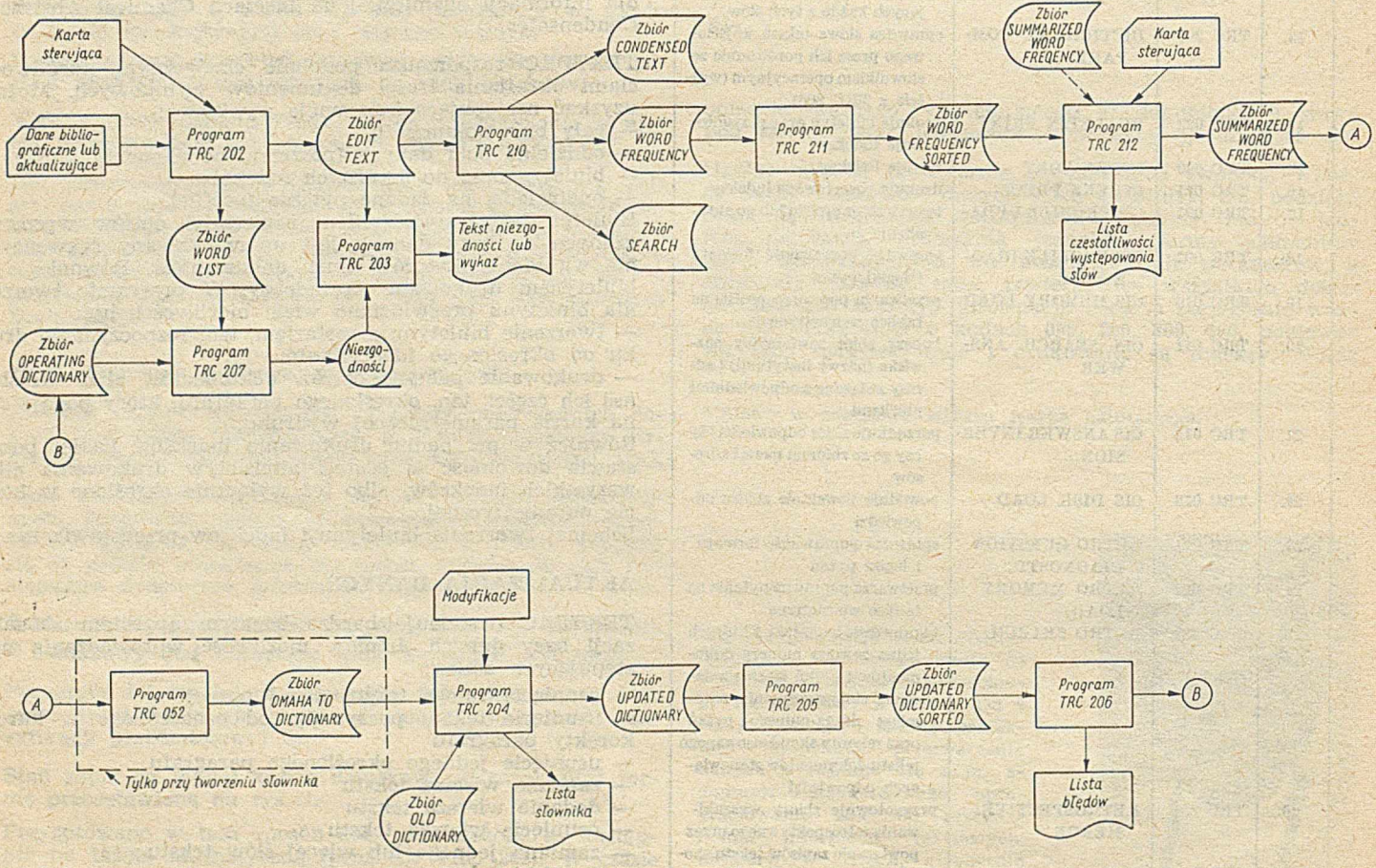
— kartoteka główna

— kartoteka słownikowa.

Kartoteka główna (EDIT TEXT) — powstaje w wyniku przekształcenia danych wejściowych na format wymagany przez system. Jest ona uporządkowana sekwencyjnie. Uporządkowanie następuje według 12-znakowego numeru identyfikacyjnego i 3-znakowego kodu pola kontrolnego. Rekordy tej kartoteki zawierają: numer dokumentu, dane bibliograficzne i inne elementy opisu treści dokumentu odpowiednio zestawione w paragrafach. Kompletność danych można sprawdzić za pomocą specjalnego programu. Z kartoteki EDIT TEXT za pomocą programu TRC 210 otrzymuje się trzy inne, bardzo ważne dla działania pakietu zbiory:

— kartotekę częstotliwości występowania słów (WORD FREQUENCY)

— kartotekę wyszukiwawczą (SEARCH)



Rys. 3. Tworzenie i aktualizacja bazy danych oraz słownika

— kartotekę tekstu skondensowanego (CONDENSED TEXT).

Kartoteka WORD FREQUENCY zawiera słowa z podaniem częstotliwości ich występowania w zarejestrowanych dokumentach.

Kartoteka SEARCH, a właściwie dwie kartoteki: RETRO SEARCH i CIS SEARCH, zawierają rekordy przystosowane odpowiednio do wyszukiwania retrospektywnego i selektywnego.

Rekordy kartoteki tekstu skondensowanego uporządkowane są według sekwencji informacji kontrolnych. Kartoteka ta łącznie z kartoteką główną oraz kartoteką EXPANDED TEXT stanowi podstawę działalności wydawniczej systemu TEXTPAC (patrz rys. 2).

Kartoteka tekstu rozszerzonego EXPANDED TEXT powstaje ze zbiorów uzyskanych w wyniku przetwarzania retrospektywnego i zawiera listę numerów pytań, dla których znaleziono odpowiedzi (posortowaną wg numerów w reprezentacji binarnej). Wyszukiwanie retrospektywne realizowane jest na podstawie kartoteki CONDENSED TEXT i RETRO SEARCH, które po połączeniu dają główną kartotekę wyszukiwawczą RETRO MASTER. Wyszukiwanie selektywne korzysta bezpośrednio z kartoteki CIS SEARCH.

Kartotekę słownikową (DICTIONARY) utworzyć można dwiema metodami:

1) przez wyperforowanie kart ze słowami, które mają wejść do słownika, alfabetyczne ich posortowanie oraz wprowadzenie do kartoteki programu aktualizacji słownika (TRC 204)

2) na podstawie zapisów wejściowych. Specjalny program (TRC 052) tworzy zapisy dla każdego słowa ze zbioru danych, które występuje 5 lub więcej razy.

Słownik nie może zawierać dwóch słów o identycznych pierwszych 20 znakach. Program aktualizacyjny (TRC 204) dodaje do rekordu informację o długości słowa. Rekordy uporządkowane są wg długości słowa (program TRC 205).

Rekordy posortowane grupuje się w bloki do 1000 bajtów (program TRC 206). Każdy blok zawiera słowa identycznej długości uporządkowane alfabetycznie.

Schemat tworzenia i aktualizacji bazy danych oraz słownika przedstawiony jest na rys. 3.

WYSZUKIWANIE INFORMACJI

Proces wyszukiwania polega, najogólniej mówiąc, na porównywaniu treściowych charakterystyk dokumentów z wymaganiami sprecyzowanymi w pytaniu użytkownika.

Pakiet TEXTPAC umożliwia wyszukiwanie selektywne (bieżące) oraz retrospektywne.

WYSZUKIWANIE SELEKTYWNE CIS

(Currennt Information Selection)

W wyszukiwaniu selektywnym operuje się tzw. profilami. Profil jest to stałe zapytanie użytkownika, przechowywane w pamięci, w specjalnej kartotece. Wyszukiwanie selektywne uruchamia się zazwyczaj przy każdej aktualizacji zbioru danych. Na wejściu programu wyszukiwawczego wprowadzany jest zbiór profili, które podlegają każdorazowo sprawdzeniu pod względem poprawności formatu i logiki. Równocześnie formuje się zbiór zawierający nazwiska (nazwy instytucji) i adresy związane ze zgłoszonymi profilami. Profile niepoprawne są ignorowane, natomiast poprawne zostają przetworzone na tablice wewnętrzne, które następnie porównuje się z rekordami zbiorów danych. Program wyszukiwawczy (TRC 011) gromadzi wybrane rekordy, które przekazywane są do fazy opracowania wydawniczego.

W wyniku wyszukiwania uzyskuje się zbiór, którego rekordy zawierają numer dokumentu i numer profilu.

Dokładny mechanizm wyszukiwania tego typu został opisany w powołanej na wstępie publikacji [1]. Zamieszczono tam również odpowiedni schemat działania.

WYSZUKIWANIE RETROSPEKTYWNE RS

(Retrospective Search)

W wyszukiwaniu retrospektywnym tworzony jest każdorazowo nowy zbiór pytań o objętości do 200 rekordów, posortowany według numerów pytań.

Każde pytanie podlega sprawdzeniu poprawności, przy czym błędnie sformułowane pytanie przekazuje się do poprawienia, natomiast prawidłowe przygotowuje się do przetworzenia (program TRC 222).

Właściwe wyszukiwanie (program TRC 223) polega na porównaniu poszczególnych pytań z rekordem utworzonego zbioru wyszukiwawczego retrospektywnego, który powstał w wyniku połączenia zbiorów tekstu skondensowanego i zapisów wyszukiwawczych. W wyniku wyszukiwania uzyskuje się dwa zbiory. Jeden zawiera numery dokumentów i pytań łącznie z danymi statystycznymi (jeżeli były one wymagane), drugi zaś — numery pytań wraz z rekordami skondensowanego tekstu dokumentów. Schemat wyszukiwania retrospektywnego jest analogiczny do schematu wyszukiwania selektywnego.

JĘZYK INFORMACYJNO-WYSZUKIWAWCZY

W systemie TEXTPAC rozróżniamy dwa rodzaje pytań wynikające z dwu metod wyszukiwania: profile w wyszukiwaniu selektywnym, zwane często pytaniami standardowymi oraz pytania w wyszukiwaniu retrospektywnym. Język wyszukiwawczy w obu rodzajach wyszukiwania jest jednakowy. Pytania mają postać wyrażeń logicznych. Pierwotnymi terminami tych wyrażeń są słowa lub ich rdzenie połączone operatorami logicznymi. Wykorzystywanych jest 6 operatorów, które spełniają następujące funkcje:

OR — odpowiada dysfunkcji-logiczne „lub"

AND — odpowiada koniunkcji-logiczne „i"

ADJACENT — skrót ADJ — grupuje słowa będące w bezpośredniej sekwencji

WITH — grupuje słowa występujące w ramach jednego zdania [2])

ABSOLUTE — skrót ABS — oznacza, że wyszukiwanie ma być zakończone pomyślnie z uwzględnieniem elementu pytania, przy którym występuje ten operator, bez względu na pozostałą logikę pytania

NOT — użyty przy określonym elemencie pytania oznacza, że występowanie tego elementu w dokumencie anuluje wszystkie pozostałe z wyjątkiem ABSOLUTE czynniki wyszukiwania.

Operatory ADJ i WITH są operatorami kontekstu zawężającymi działanie operatora AND. Pytania/profile mogą być uściślane bądź rozszerzane przez wykorzystanie:

1) maskowania słów (maskowanie końcówki po rdzeniu słowa)
— selektywnego
— bezwarunkowego

2) logiki sterowania
— CONTROL (wyszukiwanie ogranicza się do określonych pól)
— NOT-CONTROL (przeszukiwaniu nie podlegają określone pola)

3) kryteriów wyszukiwania — kryteria te określają liczbę aspektów uwzględnianych dla danego dokumentu w celu uzyskania odpowiedzi.

Wyszukiwanie odbywa się na poziomie ASPEKTOW (CONCEPTS — symbol zewnętrzny CON...) i POZIOMÓW LOGICZNYCH (LOGICAL LEVELS — symbol zewnętrzny A1, A2...). Pozwala to na dużą elastyczność w budowaniu pytań.

Zasady budowy pytań/profili zostały omówione w publikacji [1].

OGRANICZENIA NARZUCONE NA JĘZYK INFORMACYJNO-WYSZUKIWAWCZY

Ograniczenia te są dwojakiego rodzaju:

- dotyczące słów — są to ograniczenia długości słów, ich maskowania, liczby słów w jednym pytaniu/profilu oraz użyciu operatorów CONTROL I NOT-CONTROL.
- dotyczące poziomów logicznych i aspektów — są to ograniczenia typów łączników logicznych występujących w ramach jednego aspektu lub poziomu logicznego oraz ich liczby, jak również wielopoziomowego odwoływania się do jednego symbolu logicznego.

Ograniczenia te należy brać pod uwagę przy formułowaniu pytań/profili. Ich opanowanie przez użytkownika jest stosunkowo łatwe.

KONFIGURACJA KOMPUTERA

Pakiet TEXTPAC można uruchomić na różnych zestawach sprzętowych IBM obsługiwanych przez system operacyjny OS. Minimalna konfiguracja komputera serii IBM/360 zależy od:

---

2). Zdanie w znaczeniu używanym w języku naturalnym.

— wymagań narzuconych przez wygenerowaną wersję OS
— rozmiarów baz danych systemu informacyjnego.

TEXTPAC przystosowany jest do przetwarzania partiowego i opiera się na wykorzystaniu pamięci taśmowych. Standardowa jego wersja zakłada wykorzystanie 9 jednostek pamięci taśmowej oraz 1 jednostki pamięci dyskowej. W przypadku konfiguracji dyskowych wyposażenie minimalne wynosi 3 jednostki pamięci dyskowej (1 dysk systemowy plus 2 robocze) oraz 2 jednostki pamięci taśmowej.

Minimalny obszar pamięci operacyjnej pakietu TEXTPAC wynosi 256 k bajtów, z czego 128 k bajtów wystarcza na przechowanie programów problemowych. Programy napisane są w języku ASSEMBLER.

MOŻLIWOŚCI ZASTOSOWANIA PAKIETU

Pakiet TEXTPAC można adaptować do:
— informacji o bieżących nabytkach w dużych bibliotekach i ośrodkach informacji
— informacji specjalistycznych (np. informacji o pracach naukowo-badawczych, literaturze formowej, informacji dziedzinowej itp.) — przykładem jest zastosowanie tego pakietu przez Instytut Informacji Naukowo-Technicznej i Ekonomicznej do realizacji podsystemu informowania o pracach naukowo-badawczych
— selektywnego rozpowszechniania informacji tzw. SDJ (Selective Distribution Information) — przykładem jest system selektywnego wyszukiwania informacji chemicznej Biblioteki Głównej Politechniki Warszawskiej, opisany w publikacji [1].

Pakiet TEXTPAC można wykorzystać ponadto do współpracy z pakietem STAIRS. Pakiet STAIRS nie ma w swoim wyposażeniu programu przygotowującego dokument źródłowy. Może to być wykonane jednym z programów systemu TEXTPAC, a mianowicie — TRC 202. Format wyjściowy z tego programu jest akceptowany przez pakiet STAIRS. Możliwość współpracy tych dwóch pakietów poszerza więc krąg ich zastosowań.

UWAGI KOŃCOWE

Przed rozpoczęciem adaptacji pakietu TEXTPAC do konkretnych potrzeb użytkownika należy:
— ściśle ustalić formę zapisu określonego typu informacji w polach indeksowych (problem skrótów i interpunkcji) ze względu na prawidłowe zejście się w indeksach tego samego typu informacji
— ustalić zasady swobodnego indeksowania opracowywanego dokumentu, co ma wpływ na prawidłowość wyszukiwania.
— wykorzystać doświadczenia insytucji, które już system zaadaptowały.
Jak już wspomniano, w ZETO ZOWAR zmodyfikowano wiele standardowych programów pakietu TEXTPAC, rozszerzając przez to znacznie możliwości jego dostosowania do potrzeb różnych użytkowników.

LITERATURA:

[1] Wanda Pyrzyńska: System selektywnego wyszukiwania informacji chemicznej, INFORMATYKA nr 12/75
[2] Wojciech Kamiński, Izabela Rode-Babczenko: STAIRS system wyszukiwania informacji, INFORMATYKA nr 10/75

JÓZEF HOZER
GRZEGORZ KLONOWSKI
Politechnika Szczecińska

System informatyczny przedsiębiorstwa a metody ekonometryczne

Wzorem innych dziedzin życia gospodarczego mija również okres ekstensywnego rozwoju zastosowań informatyki. Należy zatem doskonalić metody budowy, wdrażania i eksploatacji systemów informatycznych. Jeden z kierunków usprawnienia rozwiązań w tym zakresie stanowią metody ekonometryczne. W trakcie budowy informatycznego systemu przedsiębiorstwa pojawiają się między innymi — problemy związane z:
— określeniem korelacji między wielkością materiału liczbowego do przetwarzania a podstawowymi parametrami działalności przedsiębiorstwa
— badaniem podatności zjawisk ekonomicznych w przedsiębiorstwie na automatyzację przetwarzania
— ustaleniem potrzeb informacyjnych niezbędnych dla zarządzania przedsiębiorstwem
— wskazaniem sposobu transformacji informacji wejściowych w celu uzyskania informacji przydatnych do celów zarządzania.

Zagadnienia powyższe, odnoszące się do poszczególnych etapów budowy systemu informatycznego przedsiębiorstwa, nie tracą nic ze swej aktualności również w fazie jego wdrażania i eksploatacji. Poważna rola tych zagadnień stwarza potrzebę znalezienia specjalnego podejścia w celu uzyskania w miarę możliwości najlepszych rozwiązań.

Mgr GRZEGORZ KLONOWSKI jest absolwentem Wydziału Inżynieryjno-Ekonomicznego Transportu Politechniki Szczecińskiej (1970) ze specjalnością organizacja przetwarzania danych. Od 1970 r. pracuje w Politechnice Szczecińskiej; obecnie jest starszym asystentem w Instytucie Rachunku Ekonomicznego. Ukończył kursy programowania i projektowania systemów informatycznych, biorąc następnie czynny udział w licznych pracach projektowo-wdrożeniowych. Jest autorem kilkunastu publikacji z dziedziny informatyki i rachunkowości.

Dr JÓZEF HOZER ukończył studia na Wydziale Inżynieryjno-Ekonomicznym Transportu Politechniki Szczecińskiej w 1969 r. ze specjalnością ekonometria. Od czasu ukończenia studiów pracuje w Politechnice Szczecińskiej, gdzie obecnie jest adiunktem w Zakładzie Ekonometrii i Badań Operacyjnych. Zajmuje się teorią regulacji w systemach ekonomicznych. Jest autorem kilkudziesięciu publikacji z zakresu ekonometrii i ekonomiki transportu samochodowego.

Nieodzowne jest więc stosowanie takich metod i narzędzi badawczych, które dają podstawę wyboru zmiennych ekonomicznych, stanowiących o potrzebie i modelu automatyzacji danego systemu zarządzania.

Wydaje się, że przy rozwiązywaniu zarysowanych tu zagadnień mogą znaleźć szerokie zastosowanie metody ekonometryczne. Jak dotąd w naszej teorii, jak i w praktyce wdrożeniowej brak konkretnych propozycji w tym zakresie.

Celem artykułu jest wskazanie na niektóre z wielu możliwych zastosowań metod ekonometrycznych w informatycznych systemach przedsiębiorstw. Wykorzystanie tych metod, już w pierwszym etapie projektowania daje z reguły trafną odpowiedź na pytanie, które spośród wielu aspektów procesu produkcyjnego przedsiębiorstwa mają decydujący wpływ na rozmiary i rodzaj materiału liczbowego, stanowiącego przedmiot późniejszego przetwarzania. Na podstawie historycznych, bieżących i planowanych wielkości wybranych zmiennych można wyznaczyć tendencję oraz oszacować przewidywane rozmiary zapotrzebowania na informacje. Ma to szczególnie duże znaczenie w odniesieniu do systemu informacyjnego, opartego na technice komputerowej [1]) z uwagi na trwający zwykle kilka lat proces projektowo-wdrożeniowy.

W tym miejscu wypada odnotować, że w najnowszej praktyce wdrożeniowej spotyka się pewne próby wykorzystania metod statystyki matematycznej [2]). Jednak w zdecydowanej większości przypadków ograniczają się one do analizy współczynników korelacji i badania linii regresji.

Obecny etap rozwoju zastosowań informatyki w naszym kraju charakteryzuje się przeniesieniem punktu ciężkości z wdrażania odcinkowych [3]) systemów informatycznych na integrację systemów już istniejących. Zabiegom integracyjnym, zmierzającym do utworzenia kompleksowego systemu informatycznego, towarzyszy zwykle poszerzenie zakresu prezentowanych informacji o informacje operatywne. Pojawia się przy tym ważny problem, jakie informacje objąć systemem, jakie zaś można pominąć. Odpowiedź na to pytanie dają metody ekonomiczne. Konstruując określone modele (ekonometryczne), stanowiące matematyczne odzwierciedlenie związków zachodzących między kwantyfikowanymi zjawiskami ekonomicznymi stwarza się warunki budowy i efektywnej eksploatacji systemu informatycznego.

W zależności od sposobu budowy i przeznaczenia model ekonometryczny, jako podstawowe narzędzie metod ekonometrycznych, może spełniać różne cele. I tak np. model przedsiębiorstwa zawiera w swoich założeniach co najmniej trzy elementy składowe, odnoszące się do działalności:

— produkcyjnej
— rynkowej
— finansowej.

Każdy z tych elementów jest do pewnego stopnia samodzielny i opisuje odrębne sfery działalności przedsiębiorstwa.

Problem analizy gospodarności odnosi się głównie do działalności produkcyjnej (np. racjonalności użycia czynników produkcji). Jednak w określonych warunkach można również mówić o gospodarności w zakresie działalności finansowej przedsiębiorstwa. Zasadniczą zmienną endogeniczną (kontrolowanym wskaźnikiem dyrektywnym) w części finansowej modelu ekonometrycznego jest rentowność. Rynkowa część modelu wskazuje sposób, w jaki przedsiębiorstwo jest powiązane z otoczeniem.

Taką trójczłonową budowę ma większość zbudowanych modeli ekonometrycznych przedsiębiorstw.

Wykorzystanie modelu w informatycznym systemie przedsiębiorstwa, a za jego pośrednictwem w systemie zarządzania tym przedsiębiorstwem, może być wielokierunkowe. Można go np. użyć do planowania, analizy skutków oddziaływania bodźców materialnego zainteresowania załogi (skuteczności instrumentów pobudzania) oraz kontroli realizacji zadań. Tak więc model ekonometryczny może znaleźć zastosowanie niemal we wszystkich funkcjach zarządzania. Zanim to jednak nastąpi muszą być spełnione odpowiednie warunki, rzutujące w istotny sposób na możliwość i stopień wykorzystania modeli ekonometrycznych w zarządzaniu. Otóż modele ekonometryczne przedsiębiorstwa, zbudowane w oparciu o nowoczesne metody ekonometryczne, zawierają w sobie zazwyczaj skomplikowane algorytmy obliczeniowe. Właściwa baza techniczna do przeprowadzenia tych obliczeń staje się jednym z podstawowych warunków praktycznego wykorzystania metod ekonometrycznych. Funkcję tę w sytuacji przedsiębiorstwa może z powodzeniem przejąć dopiero odpowiednio zaprojektowany i wdrożony system informatyczny.

Wzajemne uwarunkowanie funkcjonalności metod ekonometrycznych i systemów informatycznych świadczy o istnieniu między nimi sprzężeń zwrotnych. Sprzężenia te, jak dotąd rzadko dostrzegane są przez projektantów systemów informatycznych.

Np. wyróżniając w systemie informatycznym szczeble:
— ewidencji danych
— przetwarzania danych
— tworzenia informacji
— przetwarzania informacji
nietrudno zauważyć, że model ekonometryczny związany jest w określony sposób z każdym z tych szczebli.

I tak na etapie ewidencji danych, model ekonometryczny pomaga ustalić zakres i wydzielić dane, niezbędne dla charakterystyki określonych aspektów zjawisk gospodarczych.

Obok powszechnego w informatyce ankietowania kierownictwa, bardzo pomocny może okazać się proces ekonometrycznego modelowania danych, uznanych za podstawowe. Postać analityczna modelu ekonometrycznego daje w takim przypadku odpowiedź na pytanie, jakimi danymi należy dysponować w celu uzyskania zadanych przekrojów informacyjnych. Z drugiej strony przyczynia się do pełniejszego wykorzystania danych wejściowych, wskazując na możliwe do uzyskania nowe przekroje informacyjne. Takie zastosowanie modelu ekonometrycznego jest szczególnie użyteczne wtedy, gdy projektowanie systemu informatycznego odbywa się w warunkach nadmiaru informacji, co w praktyce występuje stosunkowo często.

Na podstawie postaci algebraicznej modelu ekonometrycznego można wyciągnąć też odpowiednie wnioski co do sposobu przetwarzania danych. W oparciu o model ekonometryczny możliwe jest sformułowanie syntetycznego ujęcia prezentowanych informacji dostosowanych do potrzeb wyższych szczebli zarządzania. Możliwość otrzymania odpowiednich informacji na temat struktury, poziomu agregacji oraz stopnia dokładności informacji znajduje swoje odbicie na wszystkich etapach prognozowania. Wydaje się jednak, że najciekawszy i najistotniejszy z punktu widzenia powiązań systemów informatycznych z metodami ekonometrycznymi, jest szczebel przetwarzania informacji [4]) Przed metodami ekonometrycznymi rysuje się tu możliwość szerokiego, praktycznego wykorzystania jako narzędzia wzbogacenia informacji — por. [4] s. 130. Zastosowanie metod ekonometrycznych może w istotny sposób przyczynić się do pełniejszego, a co za tym idzie efektywniejszego wykorzystania możliwości technicznych współczesnych komputerów.

Jednym z ważniejszych kierunków wykorzystania modeli ekonometrycznych w przetwarzaniu informacji jest niewątpliwie analiza wyników działalności przedsiębiorstw por. [1, 2]. Wkomponowany w system informatyczny przed-

---

[1]) System informacyjny funkcjonujący na bazie elektronicznych maszyn cyfrowych (komputerów) utożsamia się tu z systemem informatycznym i w tym znaczeniu określenia te występują zamiennie.

[2]) Metody wnioskowania ekonometrycznego są w istocie również oparte na metodach statystyki matematycznej, adaptowanych i rozwiniętych na użytek zastosowań ekonomicznych (głównie prognozowania ekonometrycznego)

[3]) Pojęcia systemu odcinkowego i systemu kompleksowego oraz relacje między nimi zachodzące były niejednokrotnie przedstawiane na łamach INFORMATYKI i stąd przytaczanie ich w tym miejscu uważa się za zbędne.

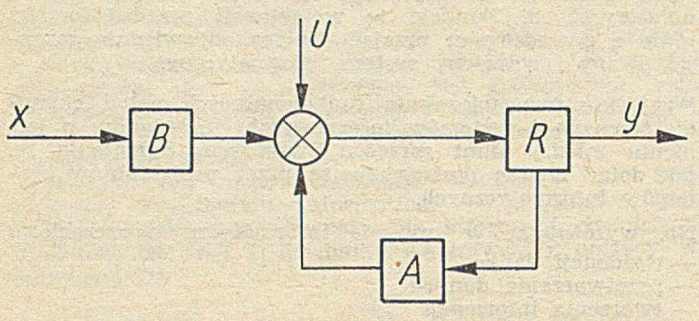
[4]) W odróżnieniu od szczebla przewarzania danych ograniczającego czynności do prostych operacji arytmetycznych charakterystycznych dla systemów informacyjnych ewidencyjno-statystycznych.

siębiorstwa model ekonometryczny może stanowić podstawę automatycznego systemu regulacji. Stan taki uzyskuje się, wmontowując w model ekonometryczny ogólnej postaci: [5]

$$Y = AY + BX + U$$

gdzie:

X — zmienne objaśniające
Y — zmienne objaśniane
A,B — parametry strukturalne (szacowane za pomocą maszyn matematycznych)
U — składnik losowy, mierzący efekty przypadkowe odpowiednie regulatory R wiążące wejścia z wyjściami informacyjnymi — por. [3] s. 132. Ilustrację graficzną przedstawionej wyżej zależności stanowi ogólny schemat układu (rys.)



Rys. Ogólny schemat układu

Zastępując zmienne Y zmiennymi ΔY można badać odchylenia od zadanych norm. Wymaga to precyzji w zakresie określenia parametrów systemu pobudzania, znajomości miejsc podejmowania decyzji i całego czeregu norm pracy i zdolności produkcyjnych przedsiębiorstwa. Za pomocą modelu ekonometrycznego (odpowiedniej postaci) można kontrolować odchylenia realizacji od przyjętej za podstawę wartości funkcji celu, zwracając szczególną uwagę na odchylenia wyjątkowe. Ponadto model taki może zawierać mechanizm kontroli odchyleń od wartości zmiennych kontrolowanych, wyznaczonych za pomocą różnych metod. Tak więc można mówić, o co najmniej trzech podstawowych rodzajach odchyleń:

$$U_t^{(1)} = Y_t - Y_t'$$

$$U_t^{(2)} = Y_t - \hat{Y}_t$$

$$U_t^{(3)} = Y_t - Y_t^x$$

gdzie:

Y_t — faktyczna realizacja zadań planowych
Y_t' — zadania planowe wyznaczone metodą tradycyjną
$\hat{Y}_t$ — zadania wyznaczone przez planistyczny model ekonometryczny
Y_{t_x} — plan wyznaczony przez kontrolny [6] model ekonometryczny

Każde z odchyleń (U_t) niesie inne informacje i w inny sposób powinno być wykorzystane w procesie sterowania przedsiębiorstwem.

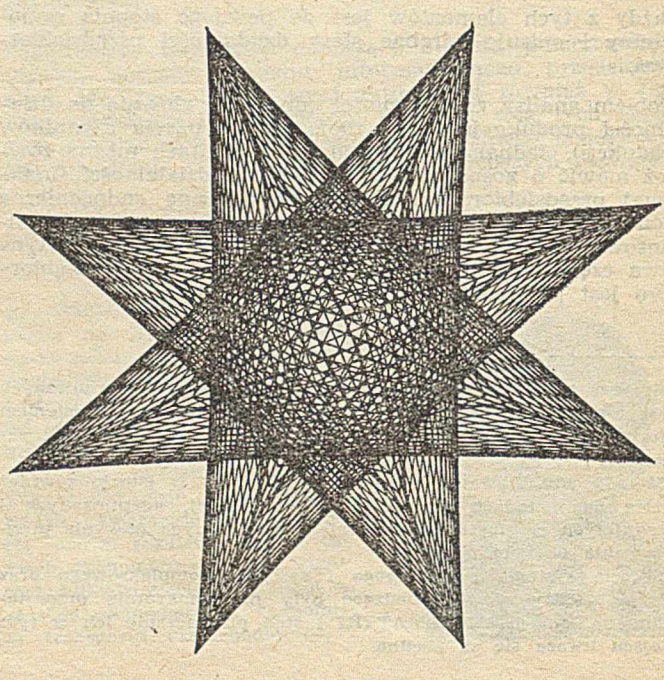
Nasuwa się spostrzeżenie, że w niektórych okresach, mimo uwzględnienia pełnego zbioru istotnych zmiennych objaśniających, mogą powstać nieprzewidziane odchylenia od planu na skutek ilościowych zmian w relacjach. W takich przypadkach model ekonometryczny może być dodatkowo wykorzystany dla analizy skutków zmiany parametrów strukturalnych modelu, czyli ilościowych relacji między badanymi zmiennymi. Tym samym model ekonometryczny staje się niejako automatycznie narzędziem symulacji. Symulacja, w warunkach funkcjonującego systemu informatycznego, daje możliwość uzyskania dodatkowych informacji w zakresie zachowania się kontrolowanego układu, którym może być wybrana dziedzina działalności przedsiębiorstwa. W szczególności wydaje się być ciekawa propozycja wykorzystania modelu ekonometrycznego do celów predykcji dyskryminacyjnej [7]. Jak już nadmieniono, w badaniach ekonometrycznych niezwykle ważną rolę odgrywa jakość informacji. Potwierdza się tym samym wspomniane uzależnienie praktyczności metod ekonometrycznych od sprawności systemu informatycznego. W przeciwnym przypadku stawia się pod znakiem zapytania przydatność ocen działalności przedsiębiorstw, m.in. z powodu zbyt dużego udziału w odchyleniach (U_t) zakłóceń o charakterze niedokładności informacyjnych.

Niezależnie od powyższego, model ekonometryczny zbudowany do celów analizy działalności gospodarczej przedsiębiorstw stawia przed systemem informatycznym wymagania odnośnie do dostarczenia informacji dotyczących subiektywnych zmiennych. Informacje te mają poważny wpływ na pogłębienie badań w zakresie rzeczywistych przyczyn powstawania odchyleń.

Należy wnioskować, że:
— istnieją przesłanki szerokiego stosowania metod ekonometrycznych w informatyce
— między metodami ekonometrycznymi i systemami informatycznymi istnieją zależności warunkujące praktyczną realizację jednych i drugich
— metody ekonometryczne mogą być z powodzeniem wykorzystane, zarówno w trakcie budowy systemu informatycznego, jak i w czasie jego eksploatacji
— zastosowanie metod ekonometrycznych w przetwarzaniu informacji przyczynia się do wzrostu efektywności systemów informatycznych.

LITERATURA

[1] Hozer J.: Próba ekonometrycznej analizy kosztów w przedsiębiorstwie transportu samochodowego. Przegląd Komunikacyjny nr 6/1970
[2] Hozer J.: Zastosowanie metod ekonometrycznych w analizie gospodarności przedsiębiorstw, Przegląd Statystyczny nr 2/1974
[3] Mynarski S.: Elementy teorii systemów i cybernetyki, PWN, Warszawa 1974
[4] Zieliński Z.: Wpływ zastosowań ekonometrii na budowę systemu informacji ekonomicznej w przedsiębiorstwie, w: Praktyczne sposoby organizowania systemu informacji ekonomicznej, Instytut Rachunku Ekonomicznego Politechniki Szczecińskiej, TNOiK, Szczecin-Kołobrzeg 1975.



[5] Pełny opis postaci wyjściowej ogólnego modelu ekonometrycznego zawiera praca [3] s. 132 i dalsze.
[6] Specjalny typ modeli ekonometrycznych do celów kontroli. Modelem takim jest na przykład model zbudowany do celów analizy gospodarności [2].
[7] Przez predykcję dyskryminacyjną rozumie się tu proces ustalania wartości zmiennych objaśniających (na wejściu systemu) dla założonych wartości zmiennych objaśnianych (informacji na wyjściu systemu informatycznego).

WŁADYSŁAW WORNALKIEWICZ
Biuro Projektowo-Technologiczne
PREDOM-PROJEKT
Wrocław

Propozycja rozwiązania podstawowej bazy danych kartotekowych przedsiębiorstwa z zastosowaniem systemu kartotek bazowych

UWAGI WSTĘPNE

Jednostki gospodarcze zgrupowane w Zjednoczeniu Przemysłu Zmechanizowanego Sprzętu Domowego „PREDOM", zostaną wyposażone, zgodnie z podjętymi wcześniej ustaleniami wynikającymi z koncepcji sieci obliczeniowej, m.in. w komputery JS EMC.

Koordynację i realizację prac w zakresie wykonania dokumentacji projektowej, programowej i eksploatacyjnej obiektowych systemów EPD powierzono Biuru Projektowo-Technologicznemu „PREDOM-PROJEKT" we Wrocławiu.

Zadaniem docelowym jest opracowanie i nadzór nad wdrożeniem wielotematycznych systemów kompleksowych, zorientowanych na podstawowe dziedziny działalności przedsiębiorstw podległych Zjednoczeniu „PREDOM". Spośród szerokiego wachlarza tematów działalności projektowej obejmującej zarówno opracowanie dokumentacji ośrodków przetwarzania danych jak i poszczególnych agend systemów kompleksowych, ograniczono się w niniejszym artykule do zaprezentowania i skomentowania zagadnienia zakładania i aktualizacji bazy danych kartotekowych przedsiębiorstwa na podstawie dokumentacji projektowo-programowej wykonanej dla Zakładów Metalowych „PREDOM-MESKO" w Skarżysku Kamiennej. Zgodnie z opracowaną koncepcją sieci obliczeniowej Zjednoczenia „PREDOM" przedsiębiorstwo to ma otrzymać komputer JS EMC typ R 30. Prace projektowe nad systemem kompleksowym rozpoczęły się przed trzema laty.

Założeniem generalnym było maksymalne wykorzystanie dostępnych w kraju gotowych pakietów oprogramowania, które w stosunkowo łatwy sposób można by przystosować do warunków i potrzeb użytkownika komputera R 30.

W trakcie prac nad systemem zaszła potrzeba wyprzedzenia opracowania dokumentacji w stosunku do terminu zainstalowania komputera.

Przyjęto koncepcję systemu kompleksowego EPD, opartego na opracowanym docelowym modelu zarządzania przedsiębiorstwem.

Zarys ogólnej struktury systemu kompleksowego przedstawiono na rys. 1, podając przykładowo szczegółową specyfikację funkcjonalnych jednostek przetwarzania wchodzących w skład podsystemu „Planowanie i rozliczanie produkcji podstawowej".

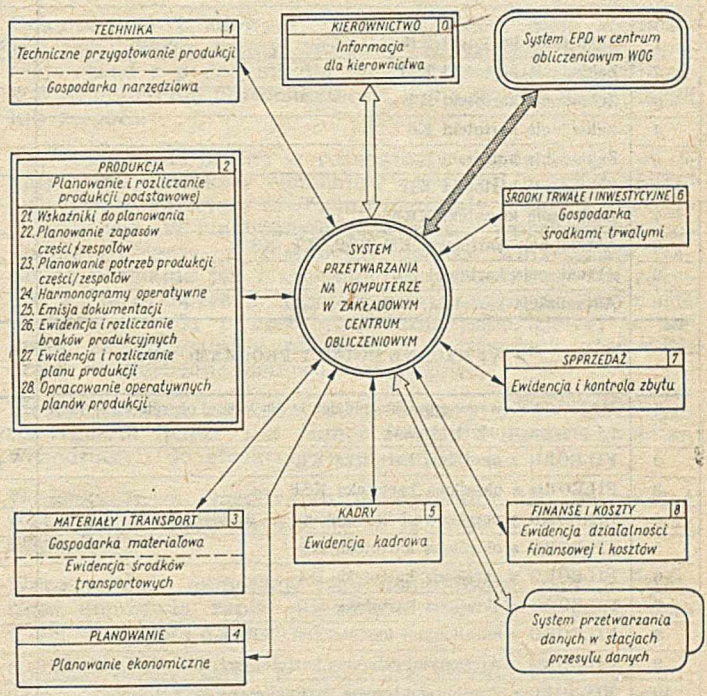
OGÓLNA KONCEPCJA UTWORZENIA BAZY DANYCH

Po opracowaniu założenia systemu kompleksowego, uwagę skoncentrowano na rozwiązaniu budowy bazy danych kartotekowych przedsiębiorstwa, a zwłaszcza wyłonieniu w niej tzw. zbiorów „centralnych" systemu kompleksowego oraz zbiorów „prywatnych" poszczególnych podsystemów. Zbiorami centralnymi uznano kartoteki podstawowych danych z zakresu technicznego przygotowania produkcji (TPP).

Wychodząc z założenia, że komputery JS EMC są odpowiednikiem komputerów serii IBM 360, uznano że również standardowe pakiety oprogramowania użytkowego będą zbliżone w swych rozwiązaniach [4]. W konsekwencji tego faktu zdecydowano się więc na wykorzystanie do realizacji ww. tematu pakietu organizacji kartotek dyskowych — BOMP oraz na uruchomienie programów zakładania i aktualizacji kartotek bazy danych w oparciu o dane modelowe z przedsiębiorstwa na komputerze IBM 360/40 w pobliskim ośrodku FSC Starachowice.

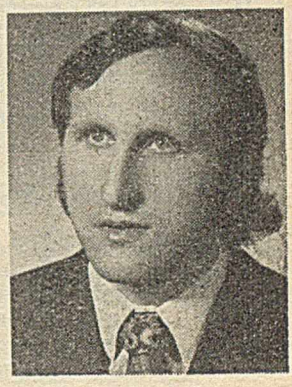
Generację modułów programowych: głównego, organizacji i wejścia/wyjścia przeprowadza się w miarę oprogramowywania poszczególnych kartotek z dysku dystrybucyjnego pakietu BOMP.

W projektowaniu podstawowej bazy danych kartotekowych przedsiębiorstwa wykorzystano doświadczenia krajowych użytkowników pakietu BOMP. Wprowadzono jednak szereg innowacji oraz przedstawiono nowe możliwości wykorzystania tego pakietu, zwłaszcza w zakresie budowy baz danych podsystemów gospodarki materiałowej, planowania produkcji podstawowej oraz gospodarki narzędziowej.



Ogólna struktura systemu kompleksowego EPD przedsiębiorstwa PREDOM-MESKO

Należy nadmienić, że baza danych przedsiębiorstwa podlega stopniowej rozbudowie i uzupełniana jest o nowe kartoteki dyskowe w miarę rozwoju systemu EPD. Traf-



Mgr inż. WŁADYSŁAW WORNALKIEWICZ ukończył w 1969 r. Wydział Mechaniczny oraz Wydział Informatyki i Zarządzania (1973 r.) Politechniki Wrocławskiej. Obecnie pracuje w Biurze Projektowo-Technologicznym Przemysłu Precyzyjnego we Wrocławiu. Zajmuje się problematyką związaną z wykorzystaniem pakietu Bill of Material Procesor (BOMP).

ność decyzji wykorzystania pakietu organizacji kartotek BOMP potwierdził fakt, że System Kartotek Bazowych [1, 2], który stanowić będzie oprogramowanie Jednolitego Systemu w ogólnej charakterystyce, zgodny jest z wykorzystywanym pakietem BOMP.

Powyższy przykład podejścia sugeruje, że również inni przyszli użytkownicy komputerów JS EMC chcąc przyspieszyć opracowanie dokumentacji projektowo-programowej (w sytuacji gdy brak jeszcze w kraju szerszego dostępu do tego typu sprzętu), mogą rozpatrzyć możliwości wcześniejszego wygenerowania modułów programowych z pakietu BOMP i przetestowania własnych programów na komputerach IBM.

Z chwilą zainstalowania we własnym ośrodku obliczeniowym sprzętu JS EMC, jego użytkownicy stosunkowo szybko zmieniając między innymi deklarację urządzeń w modułach programowych i kartach sterujących, będą mogli przenieść swoje zbiory i programy na komputery tego typu. Ogólna charakterystyka systemu organizacji kartotek BOMP, a ostatnio również Systemu Kartotek Bazowych [1] jest znana w kręgu użytkowników komputerów IBM i JS EMC, dlatego też ograniczę się tylko do podkreślenia pewnych oryginalnych cech rozwiązania w porównaniu do standardowego rozwiązania pakietu BOMP. Zostanie to przedstawione w układzie kolejnych kartotek bazy danych objętych obecną dokumentacją projektową. Kartoteki te są wzajemnie połączone. Połączenia zapisów zarówno wewnątrz kartotek, jak i między kartotekami stanowią łańcuchy składników lub zastosowania utworzone z pól adresowych. Dokumentację projektową opracowaną dla każdej z kartotek oddzielnie, a dotyczącą zakładania i utrzymywania kartotek bazy danych, uzupełniono o założenia w zakresie otrzymania wydawnictw na żądanie, uzyskując w ten sposób podstawę do opracowania dziedzinowych banków danych.

Fakt ten oczywiście nie przesądza, że opracowana dokumentacja może być stosunkowo łatwo adaptowana dla potrzeb innych jednostek gospodarczych zjednoczenia „PREDOM", które otrzymują komputery JS EMC. Pozostanie w pierwszym rzędzie problem uwzględnienia specyficznych elementów bazy indeksowej oraz normatywnej danego przedsiębiorstwa.

Zapotrzebowanie miejsca na dyskach magnetycznych dla poszczególnych kartotek obliczono na minikomputerze WANG 2200 B korzystając z programu własnego optymalizacji wykorzystania dysku o pojemności 30 MB.

Na podstawie dotychczasowej praktyki sugeruje się kolejność zakładania i aktualizacji kartotek oraz dobór modułów programowych zgodnie ze schematem podanym w tabeli.

TABELA. Kolejność zakładania i aktulizacji kartotek podstawowej bazy danych przedsiębiorstwa z propozycją doboru modułów programowych

Lp.	FUNKCJE PROGRAMU UŻYTKOWNIKA	DOBÓR MODUŁÓW PROGRAMOWYCH									
		1	2	3	4	5	6	7	8	9	10
1	Zakładanie kartoteki KR	x									
2	Zakładanie kartoteki KSP		x								
3	Zakładanie kartoteki KT			x							
4	Zakładanie kartoteki KS				x						
5	Zakładanie kartoteki KZ					x					
6	Zakładanie kartoteki KD						x				
7	Zakładanie kartoteki KRZ							x			
8	Aktualizacja kartoteki: KR KSP, KT, KS								x		
9	Aktualizacja kartoteki KZ									x	
10	Aktualizacja kartoteki: KR, KD, KRZ										x
	WYMAGANE MODUŁY PROGRAMOWE:										
1	MAINLINE (wygenerowany oddzielnie dla każdej określonej funkcji programu użytkownika)	x	x	x	x	x	x	x	x	x	x
2	FILEORG z określoną kartoteką KR	x									
3	FILEORG z określoną kartoteką KSP		x								
4	FILEORG z określonymi kartotekami: KR, KSP, i KT			x							
5	FILEORG z określoną kartoteką KS				x						
6	FILEORG z określoną kartoteką KZ					x				x	
7	FILEORG z określoną kartoteką KD						x				
8	FILEORG z określonymi kartotekami: KR, KD i KRZ							x			
9	FILEORG z określonymi czterema kartotekami: KR, KSP, KT, KS								x		
10	FILEORG z określonymi trzema kartotekami KR, KD, KRZ,										x
11	IOROOTPHAS z określonymi kartotekami: KR, KSP, KT			x							
12	IOROOTPHAS z określoną kartoteką KR				x						
13	IOROOTPHAS z określonymi trzema kartotekami: KR, KD, KRZ							x			x
14	IOROOTPHAS z określonymi czterema kartotekami: KR, KSP, KT KS								x		
15	IOROOTPHAS z określoną kartoteką KZ									x	
16	IORPOCESS (generowany dla określonej funkcji programu użytkownika			x	x			x	x	x	x
17	IOOPENCLOS z określonymi kartotekami: KR, KSP, KT			x							
18	IODPENCLOS z określoną kartoteką KR				x						
19	IOROOTPHAS z określonymi trzema kartotekami: KR, KD, KRZ							x			x
20	IOROOTPHAS z określonymi czterema kartotekami: KR, KSP, KT, KS								x		
21	IOROOTPHAS z określoną kartoteką KZ									x	

gdzie: KR — kartoteka rodzajowa
KSP — kartoteka stanowisk pracy
KT — kartoteka technologiczna
KS — kartoteka strukturalna
KZ — kartoteka zapotrzebowań
KD — kartoteka dostawców
KRZ — kartoteka realizacji zamówień
MAINLINE — moduł główny
FILEORG — moduł organizacji kartoteki (kartotek)
IOROOTPHAS — moduł we/wy (sekcja podstawowa)
IOPROCES — moduł we/wy (faza przetwarzania)
IOOPENCLOS — moduł we/wy (faza otwarcia/zamknięcia)

KONCEPCJA KARTOTEK

Kartoteka rodzajowa jest kartoteką główną tzw. macierzystą. Zawiera pełny zbiór indeksu materiałowego przedsiębiorstwa, opracowanego w oparciu o SWW. W skład tej kartoteki wchodzą również wszystkie elementy indeksu konstrukcyjnego (części, podzespoły, zespoły, wyroby produkcji własnej)

Oprócz elementów wynikających z rozwiązania standardowego pakietu BOMP kartotekę rodzajową uzupełniono o pomoce warsztatowe specjalne.

Budowa zarówno numerycznych symboli indeksu konstrukcyjnego, jak i pomocy warsztatowych wynika ze stosowanej w przedsiębiorstwie specyfiki oznaczeń tych elementów. Dokument źródłowy „Karta indeksu konstrukcyjnego" można zastąpić dokumentem „Karta rysunku", po uprzednim przystosowaniu obecnie stosowanego formularza do perforowania danych. Umożliwi to stopniowe przejęcie przez system EPD problemu związanego z gospodarką dokumentacją konstrukcyjną.

Kartoteka stanowisk pracy odbiega znacznie od rozwiązania standardowego. Stanowić będzie zbiór zagregowanych stanowisk pracy produkcyjnych i pozaprodukcyjnych oznaczonych symbolami od 0 do 6. Pojęcie agregacji odnosi się do liczebności zbioru tych stanowisk [3]. Przyjęto następujące poziomy oraz symbole agregacji:

0 — pojedyncze stanowisko pracy
1 — grupa jednorodnych stanowisk pracy
2 — gniazdo lub linia produkcyjna
3 — oddział
4 — wydział, dział
5 — zakład, służba
6 — przedsiębiorstwo.

Układ zapisów o różnej agregacji w kartotece stanowisk pracy sugeruje, że można będzie dokonać powiązań adresowych odpowiadających obrazowi struktury organizacyjnej przedsiębiorstwa. W tym celu zabezpieczono w zapisach kartoteki pola wielkości planowanych dla poszczególnych miesięcznych okresów planistycznych, przeznaczone do zarejestrowania dysponowanego oraz efektywnego funduszu czasu pracy stanowisk odpowiadających agregacji 0 do 4 lub 1 do 4.

Dokumentami źródłowymi do założenia kartoteki stanowisk pracy są:

— karta pojedynczych stanowisk pracy (oprócz ręcznych) tzw. „Karta maszyn i urządzeń" dla stanowisk o stopniu agregacji 0

— karta grup stanowisk technologicznie zamiennych tzw. „Karta grup jednorodnych stanowisk pracy" obejmująca stanowiska o agregacji 1

— karta pozostałych stanowisk zarówno produkcyjnych jak i pozaprodukcyjnych o nazwie „Karta jednostek organizacyjnych".

Symbol indeksu stanowiska pracy tworzą dwa pola: nr grupy stanowisk pracy, (wydział, gniazdo, grupa stanowisk pracy) oraz nr inwentarzowy, przy czym zapis symbolu zależny jest od stopnia agregacji stanowiska.

Kartoteka technologiczna jest zbiorem łańcuchowym umożliwiającym powiązanie adresowe kartotek głównych: rodzajowej i stanowisk pracy. Kartoteka ta obejmuje zbiór operacji technologicznych (obróbczych, montażowych, kontrolnych itp.) wynikających z kart technologicznych.

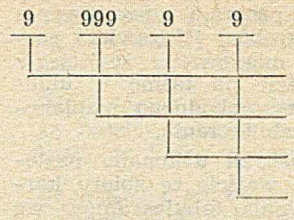
W rozwiązaniu standardowym jako pola identyfikujące poszczególne zapisy w kartotece podaje się:

— nr zespołu lub części (dla nawiązania adresu do kartoteki rodzajowej)

— nr kolejnej operacji (dla ułożenia sekwencji zapisów operacji w kartotece technologicznej)

— nr grupy stanowisk pracy (dla nawiązania adresu do kartoteki stanowisk pracy).

W prezentowanym rozwiązaniu, uwzględniającym specyficzne warunki przedsiębiorstwa „PREDOM-MESKO" w zakresie zapisu w systemie tradycyjnym informacji opisujących normę czasową (zwłaszcza dla operacji wielobsadowej) wprowadzono dodatkowe pola kontrolne stanowiące rozszerzenie zapisu operacji poprzez podanie dodatkowych informacji o następującej strukturze:



Pewne części wyrobów realizowane w cyklu produkcyjnym wykonywane są z przyczyn organizacyjnych wg kilku kolejnych kart technologicznych rozpoczynających się od tego samego numeru kolejnej operacji. Zaszła więc potrzeba podania wyróżnika kolejnej karty technologicznej dla określenia kolejności poszczególnych operacji.

W zakresie charakteru operacji dokonano podziału na:

0 — operacje wspólne (dla produkcji podstawowej oraz produkcji części zamiennych)
1 — operacje występujące wyłącznie w odniesieniu do produkcji części zamiennych np. pakowanie
2 — operacje występujące tylko w produkcji podstawowej.

W przypadku adaptacji podanego rozwiązania dla potrzeb innego przedsiębiorstwa można dokonać innego podziału operacji.

Operacje technologiczne realizowane są na konkretnej grupie jednorodnych stanowisk pracy o różnej obsadzie osobowej. Dla rozpatrywanego przedsiębiorstwa założono, że obsada jednego stanowiska nie przekracza 5 osób.

Dokumentem źródłowym wprowadzającym dane źródłowe jest tzw. „karta normy czasowej" stanowiąca wykaz operacjo-obsad z podanymi dla nich informacjami umożliwiającymi obliczenie pracochłonności jednostkowej danej części lub zespołu.

Kartoteka strukturalna w rozwiązaniu standardowym obejmuje zapisy struktur elementów produkcji własnej występujących w kartotece części lub materiały o określonej normie zużycia. W omawianym rozwiązaniu wprowadzono pozostałe elementy, a mianowicie pomoce warsztatowe (zarówno normalne jak i specjalne) podając normy ich zużycia w danej operacji. Podejście to umożliwia otrzymanie wydawnictw dla potrzeb gospodarki narzędziowej w zakresie zaopatrzenia w narzędzia i wyposażenie stanowisk pracy w niezbędne pomoce warsztatowe.

Identyfikatorem kartoteki strukturalnej w standardowym rozwiązaniu BOMP jest numer zespołu i numer składnika wchodzącego do struktury tego zespołu.

W omawianym rozwiązaniu identyfikator ten ma postać rozszerzoną, obejmującą numerację zespołu, operacji oraz składnika.

Wprowadzenie numeracji jako dodatkowego pola kontrolnego umożliwia zapis składników wg ich występowania w kartach technologicznych.

Ponadto każdy zapis w kartotece strukturalnej ma podany symbol alternatywnej operacji oraz wyróżnik przynależności składnika do struktury konstrukcyjnej. Badając ten wyróżnik można więc będzie otrzymać na żądanie tabulogram obejmujący pełny wykaz części lub zespołów określonych przez konstruktora wyrobu.

W zakresie wyszukiwania informacji z kartoteki strukturalnej opracowano programy własne w języku ASSEMBLER, wykorzystujące makroinstrukcje BOMP oraz oparte na logice pakietu standardowych programów wyszukiwania informacji dotyczącej struktury wyrobów [5].

Programy te realizować mogą 3 typy rozwinięć i zwinięć.

Rozwinięcia:

— wg jednego poziomu (dające wykazy pozycji produkcji własnej, materiałów i pomocy warsztatowych specjalnych wchodzących w skład wyrobu)
— stopniowane (z podaniem poziomu występowania poszczególnych składników w rozwijanym wyrobie)
— sumaryczne (z podaniem sumarycznej ilości danego składnika w rozwijanym wyrobie).

Programy zwinięć mają analogiczne nazwy lecz posługują się łańcuchem zastosowania określonego składnika.

Warto podkreślić, że standardowy program wyszukiwania informacji dotyczącej struktury wyrobów [5] zawiera szereg niedogodnych ograniczeń, a mianowicie: wejście z określoną ilością kart żądań, wyjście na taśmę w układzie rozplanowania wydruku. brak swobodnego rozplanowania nagłówka i linii wierszy tabulogramu.

W opracowanych programach własnych dokonano następujących udogodnień: alternatywne wejście ze zbioru kartowego lub taśmowego, rozplanowanie tabulogramów wg życzenia użytkownika, wyjście na zbiór taśmowy.

Zwrócić należy uwagę, że program rozwinięcia sumarycznego posługuje się tablicą poziomów w pamięci operacyjnej oraz tzw. polem akumulacyjnym dla rozwinięć (przy partii zamówionej) $n = 1$ w kartotece rodzajowej. W przypadku rozwinięcia potrzeb zamówionej partii wyrobów należy w kartotece rodzajowej uwzględnić drugie pole akumulacyjne (dla $n > 1$) wydruku jednostkowej ilości składników na wyrób oraz ilość składników na partię zamówienia.

Opracowanie własnych programów rozwinięć i zwinięć umożliwia łatwe dostosowanie ich do potrzeb innych przedsiębiorstw przemysłowych dla odmiennych potrzeb w zakresie rozplanowania wydruków.

Opracowana dokumentacja ww. czterech kartotek, które stanowią bazę danych TPP produkcji podstawowej, była źródłem pomysłu budowy podstawowej bazy danych kartotekowych dla podsystemu „Gospodarka materiałowa" (GM).

W koncepcji tej zastosowano oryginalne rozwiązanie, a mianowicie wykorzystano wolne połączenia adresowe w kartotece rodzajowej w zapisach pozycji materiałowych do założenia kartoteki realizacji zamówień, które łączy analogicznie jak kartoteka technologiczna kartotekę rodzajową z kartoteką dostawców.

Procedura zakładania i aktualizacji kartoteki dostawców jest zbliżona do procedur występujących w kartotece stanowisk pracy.

Dla ułatwienia odnotowania stanów magazynowych narastająco w ciągu roku dokonano w oparciu o program własny również połączenia adresowego kartoteki stanów magazynowych z kartoteką rodzajową.

Kartoteka stanów magazynowych aktualizowana jest na podstawie dokumentów obrotu materiałowego rozliczanych w cyklu miesięcznym. Kartoteki bazy danych GM można zakładać, aktualizować, a w sytuacji przepełnienia obszarów kartotek dyskowych reorganizować korzystając z istniejących modułów BOMP.

Przedstawione rozwiązanie bazy danych GM sugeruje, że można je adaptować również dla innych przedsiębiorstw przemysłowych, które korzystać będą z komputerów JS EMC. Agenda planowania produkcji podstawowej (PPP) opiera się na kartotekach TPP i GM oraz na zbiorze danych planistycznych utworzonym na podstawie identyfikatorów z kartoteki rodzajowej i technologicznej. Zbiór ten o nazwie „Kartoteka zapotrzebowań" zostanie wg założeń projektowych połączony z kartoteką rodzajową programem własnym. W tym celu w kartotece rodzajowej przewidziano pole adresowe umożliwiające połączenie obu wymienionych kartotek. Kartoteka zapotrzebowań generowana jest wg procedury kartoteki stanowisk pracy.

Wykorzystując w kartotece rodzajowej oraz kartotece stanowisk pracy wolne pola adresowe dla zapisów pomocy warsztatowych specjalnych sugeruje się utworzenie czterokartotekowej bazy danych gospodarki narzędziowej przez powiązanie kartotek głównych następującymi zbiorami łańcuchowymi:

— kartoteką technologiczną pomocy specjalnych

— kartoteką strukturalną pomocy specjalnych (wykonywanych przez dane przedsiębiorstwo).

* * *

Powyższe informacje nie podają rozwiązania pełnej podstawowej bazy danych przedsiębiorstwa. Zamierzeniem autora było tylko wskazanie możliwości w zakresie wykorzystania modułów programowych pakietu BOMP dla przyszłych użytkowników komputerów JS EMC oraz Systemu Kartotek Bazowych.

LITERATURA:

[1] Grzemowski J.: Oprogramowanie Jednolitego Systemu. System Kartotek Bazowych. Informatyka nr 11/74, str. 34—36

[2] System organizacji kartotek „BOMP". Podręcznik programowania

[3] Włoczewski J.: Zastosowanie elektronicznej techniki obliczeniowej w planowaniu i ewidencji produkcji. Stosowane nowe pojęcia i określenia. WNT Warszawa 1972, str. 27—28

[4] Jersow A.: Rozwój komputeryzacji w ZSRR INFORMATYKA 3/75, str. 8

[5] System IBM/360 — Program wyszukiwania informacji dotyczącej struktury wyrobów (360-ME-07x). Instrukcja dla programisty

[6] Dokumentacja projektowo-programowa biura „PREDOM-PROJEKT".

ANDRZEJ J. K. PLEWICKI
Uniwersytet Warszawski

Wprowadzenie do języka programowania APL (4)

APL jest systemem składającym się z języka programowania i komend systemowych umożliwiających użytkownikowi pracę w systemie oraz globalne operacje na danych i funkcjach zdefiniowanych. W poprzednich artykułach omówiliśmy sam język, teraz zajmiemy się komendami systemowymi i pewnymi szczegółami technicznymi pracy w APL.

Ogólnie są dwie możliwości pracy w systemie APL: na terminalu podłączonym do komputera IBM/360 lub IBM/370 lub na minikomputerze IBM 5100. Ponieważ IBM 5100 nie był jeszcze w Polsce prezentowany ograniczymy się do opisu pracy w APL na terminalu podłączonym z jednostką centralną dużego komputera. Istnieje kilka typów terminali, które mogą pracować w APL: 1050, 2740-1, 2741, CMC-72, 3767 oraz 3270. Istotne jest, że wszystkie te terminale mogą być połączone z jednostką centralną za pomocą zwykłej linii telefonicznej. Na przykład w październiku ubiegłego roku pokaz pracy APL na terminalu 2741 odbył się w Katowicach, podczas gdy jednostka centralna znajdowała się w Instytucie Doskonalenia Kadr Kierowniczych w Warszawie. Najczęściej używanym typem terminala jest właśnie 2741 i dlatego wszelkie konieczne szczegóły techniczne będą się odnosiły do tego typu urządzenia. Terminal 2741, z punktu widzenia użytkownika, jest elektryczną maszyną do pisania z papierem nawiniętym na wałek w sposób ciągły. Ilość znaków w linii wynosi 130. Klawiatura tego terminala do stosowania języka APL wygląda następująco:

Ponieważ APL pracuje w podziale czasu, to użytkownik ma wrażenie, że pracuje sam. Każdy użytkownik jest identyfikowany przez system poprzez swój własny numer *(SIGN-ON)*. Numer identyfikacyjny użytkownika jest mu przyznawany przez operatora systemu pracującego na terminalu uprzywilejowanym *(MASTER TERMINAL)*, znajdującym się zazwyczaj w pobliżu jednostki centralnej.

APL posiada możliwość zabezpieczenia użytkownika przed użyciem jego numeru przez innego użytkownika. Takim zabezpieczeniem jest hasło *(PASSWORD)* podawane przy włączaniu się do systemu.

Po włączeniu do systemu użytkownik otrzymuje do swojej dyspozycji aktywny obszar pracy *(ACTIVE WORKSPACE)*, to znaczy pewien obszar w pamięci operacyjnej komputera. W chwili włączenia aktywny obszar nazywa się *CLEAR WS* i nie zawiera żadnej zmiennej i funkcji.

Oprócz tego użytkownik ma swoją bibliotekę *(LIBRARY)*, w której może przechowywać pewną, zadaną przez operatora systemu, liczbę obszarów pracy. Cały obszar można załadować *(LOAD)* z biblioteki do obszaru aktywnego, co pozwala uruchamiać zawarte w nim funkcje. Załadowanie obszaru z biblioteki powoduje, że poprzednia zawartość obszaru aktywnego zostaje z niego wymazana. Kopia obszaru aktywnego może być przesłana do biblioteki użytkownika i tam przechowana *(SAVE)* po uprzednim nadaniu jej nazwy *(WSID)*. Nazwa obszaru może się pokrywać z nazwą zmiennej lub funkcji w nim zawartej. Pojedyncze zmienne lub funkcje mogą być kopiowane *(COPY)* z obszaru w bibliotece. Odwrotnie, użytkownik może wymazać *(DROP)* każdy obszar ze swojej biblioteki lub wymazać *(ERASE)* funkcje i zmienne z obszaru aktywnego.

Ponieważ czasami potrzebne są tylko pewne grupy funkcji i zmiennych, w APL można je łączyć w grupy *(GROUP)* nadając całej grupie wspólną nazwę. Nazwa grupy może być nazwą jednego z obiektów w grupie. Inną zaletą grupy jest możliwość jej całego skopiowania, podczas gdy w przeciwnym wypadku należy kopiować każdy obiekt osobno.

Oprócz bibliotek „prywatnych" dostępnych tylko dla ich właścicieli istnieją jeszcze biblioteki ogólnodostępne o numerach od 1 do 999. Każdy użytkownik może załadować dowolną przestrzeń z biblioteki ogólnodostępnej lub skopiować dowolny obiekt. Biblioteki ogólnodostępne zawierają gotowe rozwiązania najczęściej spotykanych problemów. W jednym systemie wszystkie obszary pracy mają tę samą wielkość, określoną w czasie generowania systemu. Standardowa wielkość obszarów pracy wynosi 72 k bajtów (72×2^{10}). Maksymalna wielkość wynosi 16 M bajtów (16×2^{20}) co zadowala chyba najbardziej nawet wybrednych użytkowników.

Za pomocą funkcji systemowej użytkownik może się w każdej chwili dowiedzieć, ile miejsca zostało mu jeszcze w obszarze aktywnym. Symbolem funkcji systemowej jest nałożony na siebie znak funkcji kodowania i dekodowania, a wymieniona funkcja ma numer 22.

```
      ⎕22
73728
```

Każdy obszar pracy ma swoje parametry, które zachowują się przy wszelkich operacjach dokonywanych na tym obszarze. Są to: szerokość wydruku *(WIDTH)*, indeks początkowy *(ORIGIN)*, liczba cyfr znaczących drukowanych w wyniku *(DIGITS)* oraz liczba nazw *(SYMBOLS)*.

Szerokość wydruku może być zmieniana w zakresie od 30 do 130 znaków. Indeks początkowy może wynosić 0 lub 1 i jest on określony jako pierwsza liczba naturalna. Jak wiadomo 0 jest uważane lub nie jest uważane za liczbę naturalną. APL pozwala na wyznawanie obydwu poglądów.

Liczba cyfr znaczących w wydruku może być zmieniana w zakresie od 1 do 16.

Liczba nazw oznacza maksymalną liczbę nazw, jaka może być używana w danym obszarze. Jest to istotne, gdyż każda nazwa zajmuje w obszarze pewną ilość miejsc pamięci. Minimalna liczba nazw wynosi 26. Zmiany liczby nazw można dokonać tylko w czystym obszarze aktywnym *(CLEAR WS)*.

CLEAR WS ma następujące parametry: szerokość wydruku 120, indeks początkowy 1, liczba cyfr 10 i liczba nazw 256.

Użytkownicy znajdujący się przy różnych terminalach mogą między sobą przesyłać wiadomości *(MESSAGE)* dzięki temu, że każdy terminal ma swój adres *(PORT)*.

Ze względu na to, że czasami następuje zatrzymanie wykonania funkcji wprowadzony został tak zwany wskaźnik stanu *(STATE INDICATOR)*, który w wyniku daje chronologiczną listę zatrzymanych funkcji z wymienieniem numeru wiersza, w którym funkcja została zatrzymana, przy czym na szczycie tej listy zostaje wymieniona funkcja, która została zatrzymana jako ostatnia. Gwiazdka użyta obok nazwy funkcji oznacza, że funkcja ta jest używana przez inną funkcję, oczywiście wymienioną w następnej pozycji listy funkcji zatrzymanych.

Na przykład

```
H[7] *
G[2]
F[3]
```

oznacza, że zatrzymana została funkcja H w wierszu numer 7 oraz, że funkcja ta jest użyta przez funkcję G w wierszu numer 2, a z kolei funkcja G jest użyta przez funkcję F w wierszu numer 3. Wskaźnik stanu można wyczyścić za pomocą strzałki poziomej w prawo, przy czym jedna taka strzałka usuwa z listy funkcji zatrzymanych tylko tę funkcję, która znajduje się na jej szczycie. Definicję funkcji zatrzymanej można otworzyć tylko wtedy, gdy jest ona ostatnią zatrzymaną funkcją.

Omówimy teraz poprawianie błędów. W czasie pracy na terminalu, gdy chcemy wprowadzić treść linii do komputera przyciskamy klucz oznaczony na klawiaturze jako RETURN. Wówczas klawiatura zostaje zablokowana aż do czasu przyjścia odpowiedzi z komputera. Po wydrukowaniu odpowiedzi klawiatura zostaje odblokowana i możemy ponownie pisać to, co chcemy wprowadzić do wykonania. Jeżeli podczas pisania popełnimy błąd i nie wprowadziliśmy jeszcze tego wiersza do komputera (nie przycisnęliśmy klucza RETURN), to mamy jeszcze szansę na poprawienie tego błędu. W tym celu kluczem BACKSPACE cofamy element piszący (w terminalu 2741 jest to głowica pisząca) do znaku, który chcemy poprawić i przyciskamy klucz ATTN. System, w odpowiedzi wydrukuje pod tym znakiem symbol koniunkcji i przesunie papier o jeden wiersz do góry. Wszystko to co napisaliśmy po wskazanym przez nas znaku, łącznie z tym znakiem, zostaje usunięte z pamięci terminala, natomiast to, co teraz napiszemy, zostaje zapamiętane jako dalsza część tego wiersza.

Na przykład poprawienie błędu przy dzieleniu przez zero może wyglądać następująco

```
      3÷0+2
        ∧
      1+3
0.75
```

Jeżeli nie poprawimy błędu, to albo otrzymamy wyniki, które nie mają dla nas sensu, albo system wypisze nam rodzaj popełnionego błędu *(ERROR MESSAGE)* i wskaże jego miejsce po wydrukowaniu błędnego wyrażenia. Podamy teraz i omówimy najczęściej spotykane rodzaje błędów.

RESEND	— błąd linii telefonicznej przesyłającej dane. Należy wówczas jeszcze raz przesłać ostatni wiersz
CHARACTER ERROR	— użycie znaku nie należącego do alfabetu APL
VALUE ERROR	— użycie nazwy zmiennej, która nie ma nadanej wartości
DOMAIN ERROR	— użycie argumentu nie należącego do dziedziny funkcji podstawowej

SYNTAX ERROR — błąd składni: dwie zmienne bez funkcji między nimi, funkcja bez argumentu, brak lub nadmiar nawiasu, niewłaściwe użycie funkcji zdefiniowanej

RANK ERROR — niezgodność wymiarów argumentów

LENGTH ERROR — niezgodność wymiarów macierzy

DEPTH ERROR — użycie pętli nieskończonej

LABEL ERROR — błędne użycie dwukropka w definicji funkcji

WS FULL — za mały obszar pracy

DEFINITION ERROR — niewłaściwe otwarcie funkcji, próba otwarcia funkcji zatrzymanej, czekającej na wykonanie innej funkcji

SYSTEM ERROR — błąd jednostki centralnej. Po takim błędzie zawartość obszaru aktywnego zostaje wymazana, zagubiona i użytkownik otrzymuje obszar *CLEAR WS*.

Podamy teraz i omówimy komendy systemowe. Obok każdej komendy podajemy cyfry oznaczające numer odpowiedzi drukowanej przez system w przypadku nie wykonania komendy *(TROUBLE REPORT)*.

Podane hasła mogą być pominięte w zależności od użytkownika. Wszystkie komendy systemowe zaczynają się od zamkniętego nawiasu okrągłego.

Odpowiedzi systemu

1 *NUMBER NOT IN SYSTEM*
2 *INCORRECT SIGN-ON*
3 *ALREADY SIGNED-ON*
4 *NUMBER IN USE*
5 *NUMBER LOCKED OUT*
6 *NOT WITH OPEN DEFINITION*
7 *WS NOT FOUND*
8 *WS LOCKED*
9 *OBJECT NOT FOUND*
10 *NOT GROUPED, NAME IN USE*
11 *WS FULL*
12 *NOT SAVED, WS QUOTA USED UP*
13 *NOT SAVED, THIS WS IS...*
14 *IMPROPER LIBRARY REFERENCE*
15 *MESSAGE LOST*
16 *INCORRECT COMMAND*

Komendy systemowe

)2438: *PASSWORD* 1 2 3 4 5 16

Powoduje włączenie użytkownika o numerze 2438 do systemu. W odpowiedzi system drukuje numer terminala, godzinę, datę, nazwisko użytkownika oraz rodzaj systemu *(APL/360, APLSV, APL/CMS)*.

)OFF: PASSWORD 16

Powoduje wyłączenie użytkownika z systemu. Podobne hasło staje się nowym hasłem. W odpowiedzi system drukuje numer terminala, godzinę, datę, inicjały użytkownika, czas połączenia i łączny czas dotychczasowych połączeń oraz czas jednostki centralnej zużyty w czasie połączenia i łączny czas tej jednostki.

)PORT 16

Drukuje listę aktualnie działających terminali wraz z inicjałami użytkowników.

)MSGN PORT TEKST 15 16

Powoduje przesłanie wiadomości *TEKST* do użytkownika pracującego przy terminalu o numerze *PORT*. Klawiatura terminala zostaje zablokowana do momentu dojścia wiadomości do adresata o czym system zawiadamia drukując słowo *SENT*. Wiadomość może być dowolnym ciągiem znaków występujących w alfabecie APL.

)OPRN TEKST 15 16

Powoduje przesłanie wiadomości *TEKST* do operatora systemu na takich samych warunkach, jak przesłanie wiadomości do innego użytkownika.

)WSID 16

Powoduje wydrukowanie nazwy obszaru aktywnego.

)WSID NAZWA 16

Powoduje nadanie nazwy *NAZWA* obszarowi aktywnemu. W odpowiedzi system drukuje *WAS NAME*, gdzie *NAME* jest poprzednią nazwą tego obszaru.

)ORIGIN N 16

W wyniku indeks początkowy otrzymuje wartość *N*. W odpowiedzi system drukuje *WAS M*, gdzie *M* jest poprzednią wartością indeksu początkowego.

)WIDTH N 16

Powoduje zmianę szerokości wydruku na *N*. W odpowiedzi system drukuje *WAS M*, gdzie *M* jest poprzednią wartością szerokości wydruku.

)DIGITS N 16

Powoduje zmianę cyfr znaczących w wydruku na *N*. W odpowiedzi system drukuje *WAS M*, gdzie *M* jest poprzednią wartością liczby cyfr znaczących.

)SYMBOLS N 16

Powoduje zmianę liczby nazw na *N*. W odpowiedzi system drukuje *WAS M*, gdzie *M* jest poprzednią wartością liczby nazw.

)LIB 14 16

Powoduje wydrukowanie nazw przestrzeni znajdujących się aktualnie w bibliotece użytkownika.

)LIB NUMER 14 16

Powoduje wydrukowanie nazw przestrzeni znajdujących się w bibliotece ogólnodostępnej o numerze *NUMER*.

)LOAD NAZWA: PASSWORD 7 8 16

Powoduje wymazanie zawartości obszaru aktywnego i załadowanie do niego zawartości obszaru o nazwie *NAZWA*, znajdującej się w bibliotece użytkownika. W przypadku, gdy obszar ten był przesłany do biblioteki z hasłem, należy je podać.

)LOAD NUMER NAZWA 7 8 16

Powoduje załadowanie obszaru o nazwie *NAZWA* z biblioteki ogólnodostępnej o numerze *NUMER* do obszaru aktywnego użytkownika na takich samych zasadach, jak powyżej. W obydwu przypadkach załadowania obszaru do obszaru aktywnego system w odpowiedzi drukuje *SAVED GODZINA DATA*, gdzie *GODZINA* i *DATA* określają ostatnie przechowanie obszaru (włożenie do biblioteki).

)FNS 16

Drukuje nazwy funkcji zdefiniowanych znajdujących się w obszarze aktywnym.

)VARS 16

Drukuje nazwy zmiennych zawartych w obszarze aktywnym.

)GRPS 16

Drukuje nazwy grup znajdujących się w obszarze aktywnym.

)GRP NAZWA 16

Drukuje nazwy zmiennych i funkcji wchodzących w skład grupy o nazwie *NAZWA*.

)SI 16

Drukuje listę zatrzymanych funkcji wraz z numerami wierszy, w których zostało zatrzymane ich wykonanie.

)SIV 16

Przykład pracy w APL

```
      ⍳5
0 1 2 3 4
      ∇ Z←BIN N
[1]   LA:Z←(Z,0)+0,Z
[2]   →LA×N≥⍴Z
[3]   ∇
      BIN 3
VALUE ERROR
BIN[1] LA:Z←(Z,0)+0,Z
                  ^
      Z←1
      →1
1 3 3 1
      BIN 4
VALUE ERROR
BIN[1] LA:Z←(Z,0)+0,Z
                  ^
      ∇ BIN[.1]Z←1∇
      )SI
BIN[1]
      →1
1 4 6 4 1
      )WSID ABC
WAS CLEAR WS
      )SAVE
ABC    12.23.34 04/02/76
      )OFF
002  12.24.07 04/02/76 PLE
CONNECTED     0.15.34  TO DATE   57.13.41
CPU TIME      0.00.12  TO DATE   13.09.03


      )2438
INCORRECT SIGN-ON
      )2438:ANDRZEJ
002) 12.08.00 04/02/76 PLEWICKI


  A P L \ 3 6 0
      )LIB
RESEND
      )LIB
ANDRZEJ
JERZY
HSW
      ⍳22
77876
      )WSID
CLEAR WS
      )LOAD HSW
SAVED  14.30.46 12/29/75
      )VARS
      )FNS
ALF     ALMIN   DIST    POLFIT
      ∇ POLFIT[□]∇
      ∇ Z←POLFIT N
[1]   Z←Y⌹X∘.*0,⍳N
      ∇
      )CLEAR
CLEAR WS
      )ORIGIN 0
WAS 1
```

Drukuje listę zatrzymanych funkcji i nazwy zmiennych lokalnych związanych z daną funkcją.

)ERASE NAZWY 16

Powoduje wymazanie z obszaru aktywnego zmiennych, funkcji i grup wymienionych jako *NAZWY*.

)CLEAR 16

Powoduje wymazanie wszystkich obiektów z obszaru aktywnego. W wyniku tej operacji obszar aktywny staje się obszarem *CLEAR WS*.

)COPY NUMER NAZWA OBIEKT 6 7 8 9 11 16

Powoduje skopiowanie do obszaru aktywnego obiektu o nazwie *OBIEKT* z obszaru *NAZWA* znajdującego się w bibliotece ogólnodostępnej o numerze *NUMER*. Komenda kopiowana pozwala na jednorazowe skopiowanie tylko jednego obiektu, zmiennej, funkcji lub grupy. Opuszczenie nazwy obiektu powoduje skopiowanie zawartości całego obszaru. Opuszczenie numeru biblioteki powoduje skopiowanie z biblioteki użytkownika. W przypadku, gdy nazwa obiektu kopiowanego pokrywa się z nazwą obiektu znajdującego się w obszarze aktywnym, to po skopiowaniu otrzymamy w obszarze aktywnym obiekt, który był kopiowany. Oznacza to, że zmienne funkcje i grupy o tych samych nazwach co zmienne funkcje i grupy kopiowane zostają zastąpione przez te ostatnie.

)PCOPY NUMER NAZWA OBIEKT 6 7 8 9 11 16

Działa tak samo jak komenda *COPY*, z tym, że zostają skopiowane tylko te obiekty, których nazwy nie pokrywają się z nazwami obiektów w obszarze aktywnym.

)GROUP NAZWA NAZWY 10 16

Powoduje nadanie nazwy *NAZWA* grupie obiektów wymienionych jako *NAZWY*. Nazwa grupy może być nazwą jednego z obiektów grupy.

)SAVE NAZWA: PASSWORD 6 12 13 16

Powoduje przesłanie do biblioteki użytkownika kopii obszaru aktywnego i przechowanie go tam pod nazwą *NAZWA*. Opuszczenie nazwy powoduje przechowanie obszaru pod aktualną nazwą obszaru aktywnego.

)DROP NAZWA 7 16

Powoduje wymazanie z biblioteki użytkownika obszaru o nazwie *NAZWA*.

Podamy teraz przykład pracy w APL od włączenia się do systemu do wyłączenia się.

Na tym kończymy wprowadzenie do systemu APL. W ostatnich latach zostały opracowane i uruchomione dwie nowe wersje tego języka: APLSV i APL/CMS. Pozwalają one na użycie APL prawie wszędzie tam, gdzie zachodzi potrzeba użycia komputera. Opis tych wersji APL i ich zastosowań postaramy się podać w następnych artykułach. Na zakończenie chciałbym przeprosić wszystkich czytelników za tak suchy, zwięzły opis APL, ale była to jedyna droga do zmieszczenia się w tej objętości.

LITERATURA

[1] Berry P .C. APL/360 Primer. IBM.
[2] Falkoff A. D., Iverson K. E. APL/360 User's Manual. IBM.
[3] Iverson K. E. APL Language. IBM.
[4] Iverson K. E. A Programming Language. J. Wiley.

CZYTELNIKU!

Pamiętaj o konieczności zapewnienia sobie prenumeraty na 1977 rok

Szczegółowe warunki zostaną podane w następnym numerze INFORMATYKI

Detekcja i lokalizacja defektów w sieciach logicznych*)

Doświadczenie kilkunastu lat eksploatacji maszyn cyfrowych wykazało, że niemożliwe jest w praktyce osiągnięcie pełnej niezawodności zarówno sprzętu, jak i oprogramowania.

Pojawiły się więc próby rozwiązań sprzętowych i programowych zmierzające do umożliwienia prowadzenia w systemie cyfrowym poprawnych obliczeń mimo istnienia w nim defektów. Całość przedsięwzięć zmierzających do tego celu zarówno tradycyjnych (zapobieganie powstawania błędów) jak i niekonwencjonalnych (ignorowanie szkodliwych skutków błędów) zdaje się tworzyć ostatnio osobną, zintegrowaną w pewnym stopniu gałąź wiedzy komputerowej zwaną Fault Tolerant Computing — „liczenie mimo uszkodzeń". Przegląd stanu wiedzy w dwóch działach tej nowej dziedziny, w skrócie zwanej FTC — a mianowicie zapobiegania powstawaniu błędów wskutek defektów sprzętu i zapobieganie powstawaniu błędów przy produkcji oprogramowania zaprezentowany w ramach dorocznej, piątej już konferencji „FTC-5" w Paryżu (czerwiec 1975) przedstawiony zostanie w tym i następnym numerach naszego pisma.

Wraz z pojawieniem się pierwszych komputerów personel, który je obsługiwał stanął przed problemem napraw obwodów logicznych, a zatem detekcji i lokalizacji uszkodzeń. W epoce tej przeważały techniki czysto manualne i nieco rzemieślnicze.

Wskutek wprowadzenia technologii mikroelektronicznej złożoność i gęstość sieci logicznych w istotny sposób wzrosła. Obecnie na płytce obwodu drukowanego mieści się z łatwością kilkaset podstawowych funktorów logicznych (bramek i przerzutników), modularne konstrukcje równie jak zredukowane wymiary geometryczne powodują, że tradycyjne metody napraw są mało praktyczne (jeśli nie niemożliwe), z pewnością zaś nieekonomiczne.

Z drugiej strony ilość końcówek wejściowych i wyjściowych oraz elementów „pamięci" zawartych w systemach logicznych o ograniczonych nawet rozmiarach, zwiększyła znacznie ilość kombinacji, które trzeba zbadać, aby stwierdzić poprawność funkcjonowania wszystkich elementów składowych systemu.

Rozwiązanie problemu testowania — „łamigłówka wyższego stopnia" — możliwa jest dzięki automatyzacji zarówno przygotowywania sekwencji testów, jak i ich praktycznego stosowania.

Pierwsze prace dotyczące takiej automatyzacji pojawiły się w przybliżeniu w latach pięćdziesiątych. Od chwili zasygnalizowania problemu bibliografia, którą trudno by nazwać wyczerpującą osiągnęła nie mniej niż 2000 tytułów. Pomimo tak licznych prac nie mamy odpowiedzi na dwie podstawowe kwestie:

— czy automatyczna generacja sekwencji testowych jest problemem rozwiązalnym?
— jeśli nie, jakie są rezultaty teoretyczne, użyteczne w praktyce w zastosowaniach przemysłowych?

Spróbujmy odpowiedzieć na te pytania dokonując przeglądu literatury i analizy praktyki stosowanej w przemyśle.

DEFINICJE

Wobec braku ustalonej terminologii używane będą w miarę możności niżej zdefiniowane terminy:

defekt: fizyczna niesprawność prowadząca do trwałego lub przemijającego złego działania

defekty nierozróżnialne: defekty, których skutki są identyczne

defekt niewykrywalny: defekt, którego skutek nie wpływa na działanie sieci

wejścia i wyjścia pierwotne: wejściowe i wyjściowe końcówki sieci dostępne z zewnątrz

test: ustawienie wartości logicznych na wejściach pierwotnych

sekwencja testowa: szereg takich ustawień

długość sekwencji: ilość testów, które się na nią składają

sieć redundancyjna: sieć, której pewne defekty są niewykrywalne.

PROBLEMY

Wyznaczenie testu sieci logicznej wiąże się z kilkoma problemami różnej wagi, zależnej od stopnia tych problemów.

Pierwszy z nich to problem detekcji; jest on istotny wtedy, gdy wystarcza jedynie informacja o istnieniu lub nieistnieniu defektu, to znaczy, gdy chcemy się upewnić, że sieć jest sprawna.

Drugi problem dotyczy **lokalizacji lub diagnostyki** defektów; w przypadku stwierdzonego złego działania chcemy wiedzieć, który element składowy sieci jest tego przyczyną.

Test sieci ma zatem do spełnienia trzy zadania:

— weryfikacja poprawnego działania
— lokalizacja przyczyny błędu
— weryfikacja poprawnej eliminacji defektu po naprawie.

Dwa pierwsze z wymienionych wyżej problemów sformułowane są w sposób precyzyjny w licznych opracowaniach zarówno dla sieci kombinacyjnych jak i sekwencyjnych.

BŁĘDY i DEFEKTY

Zanim rozpatrzymy problemy detekcji i lokalizacji, celowe jest wyjaśnienie założeń przyjmowanych w stosunku do powstałych defektów. Założenia te dotyczą:

natury błędów: rozróżnia się zwykle błędy logiczne i błędy propagacji.

Błędy logiczne odpowiadają modyfikacji funkcji sieci, podczas gdy błędy zwane błędami propagacji odpowiadają zmianom czasu propagacji powodowanym przez deformacje impulsów lub przez impulsy szkodliwe

skutków defektów: rozważyć należy cztery kategorie defektów:

a) defekty modyfikujące funkcję logiczną pojedynczego elementu. W szczególnym, bardzo istotnym w praktyce przypadku chodzi tu o defekty zwane zazwyczaj sklejeniami z 0 lub 1. Trzeba podkreślić, że określenie „defekt sklejenia" końcówki z 0 lub 1 niekoniecznie musi oznaczać, że ta ostatnia ma potencjał wysoki lub niski. Należy poprostu oczekiwać, że element wewnętrzny, z którym końcówka jest połączona, będzie się zachowywał tak, jak gdyby odpowiednie wejście lub wyjście było efektywnie zwarte z zerem lub jedynką, niezależnie od funkcji logicznej.

Trzeba tu rozróżnić między innymi defekty zwane „ekwipotencjalnymi" i defekty „końcówek wejściowych". Pierwszy przypadek odpowiada defektom dotyczącym wyjścia elementu, których skutki rozprzestrzeniają się na wszystkie rozgałęzienia o tym samym potencjale.

Drugi przypadek odpowiada defektom dotyczącym końcówek wejściowych elementu.

Aby uzupełnić założenie o defektach typu sklejeń wspomnijmy, że istnieje wiele rodzajów defektów, które nie mogą być traktowane jako proste sklejenia

b) defekty, które zmieniając funkcję logiczną elementu uszkodzonego wpływają na działanie elementów, z którymi jest on połączony

c) defekty, które nie zakłócają działania elementu uszkodzonego, ale zakłócają działanie elementów z nim połączonych

d) defekty wpływające na wiele elementów; najbardziej reprezentatywnym przykładem tego typu są zwarcia przewodów. Skutek zwarcia dwóch końcówek zależy od ich miejsca w sieci i od własności elektrycznych obwodu związanego z każdą z funkcji logicznych; innymi słowy od technologii.

*) J.-C. Rault — La detection et la localisation des défauts dans les circuits logiques.

Tak więc skutków różnych typów defektów nie da się na ogół określić drogą prześledzenia tylko schematu logicznego sieci; trzeba jednocześnie brać pod uwagę takie czynniki jak topologia i zastosowana technologia. Z uwagi tej wynika, że opracowanie programu automatycznej generacji sekwencji testowych wymaga uwzględnienia związków z innymi programami wspomaganego komputerem projektowania sieci i wytwarzania obwodu.

- **czasu trwania defektów:** defekty mogą być trwałe lub przemijające
- **ilość defektów:** czyni się zwykle dwa założenia:

defekt jednokrotny: jednocześnie rozważany jest tylko jeden defekt

defekt wielokrotny: w sieci wystąpić może wiele obecnych jednocześnie defektów.

Zazwyczaj czyni się założenie o istnieniu jedynie defektów jednokrotnych przyjmując implicite, że prawdopodobieństwo niewykrycia defektu wielokrotnego przy użyciu sekwencji testowej, uzyskanej przy założeniu istnienia defektu jednokrotnego, jest dostatecznie małe.

Hipoteza ta nie zawsze jest prawdziwa, zwłaszcza w przypadku sieci redundancyjnych, dla których konieczne jest rozpatrywanie defektów wielokrotnych

- **względnej częstotliwości defektów:** doświadczenie wskazuje, że pewne defekty zdarzają się częściej od innych. Rozważyć trzeba wpływ na potencjalne defekty wielu czynników, których wagę określają studia niezawodnościowe zastosowanej technologii, niestety, bardzo mało jest danych statystycznych dla tego celu.

W konkluzji stwierdzamy, że większość algorytmów używanych dziś przy generacji sekwencji testowych i symucji sieci logicznych zawierających defekty opiera się na dwóch założeniach upraszczających:

- rozpatrywane defekty są uszkodzeniami typu „sklejenia", rzadziej zwarcia
- testowana sieć zawiera defekt pojedynczy.

Dla ścisłości zauważmy, że założenia te są zwykle dopuszczalne, choć prowadzą do pewnych braków praktycznych, trzeba jednak podkreślić, że ich ewentualna nieodpowiedzialność do zastosowanej technologii może zdezawuować rezultaty otrzymane w systemie automatycznej generacji sekwencji testowych.

DEFEKTY RÓWNOWAŻNE

Wiadomo, że ilość defektów możliwych w sieci rośnie bardzo szybko wraz z ilością połączeń. Tak więc, ze względów oszczędności zarówno czasu obliczeń, jak i obszaru pamięci i niezbędnej dla programu generacji sekwencji testowych, istotne jest uniknięcie wszelkiej redundancji w liście założonych defektów.

Redukcja ilości defektów przez grupowanie ich w klasy równoważności jest podejściem najczęściej stosowanym. Przypadek sieci kombinacyjnych, obecnie już klasyczny, znajduje zadowalające rozwiązania w zakresie defektów ograniczających się do sklejeń.

W przypadku sieci sekwencyjnych napotykamy problemy koncepcyjne i praktyczne, które ciągle pozostają nierozwiązane.

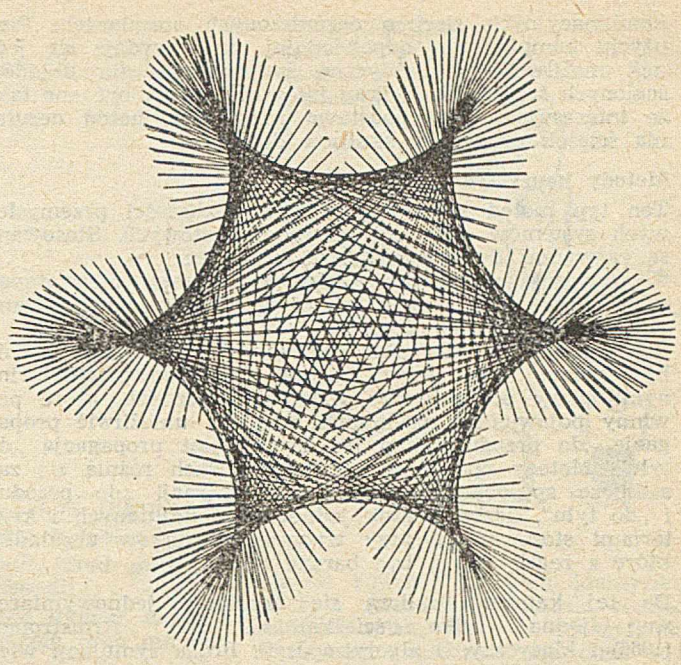
W praktyce, dla obwodów drukowanych zawierających układy scalone ograniczamy się do rozpatrywania równoważnych defektów na poziomie połączeń między kostkami układów i nielicznych defektów na poziomie bramek wewnątrz tych układów.

ZASADA GENERACJI SEKWENCJI TESTOWYCH

Należy stwierdzić, że generacja sekwencji testowych opiera się na dwóch głównych zasadach deterministycznej i probabilistycznej prowadzących do licznych metod, mieszanych w mniejszym lub większym stopniu.

Zasada deterministyczna (stosować można trzy podejścia):

- **analiza:** zestaw kombinacji podawany jest na wejścia sieci, następnie określa się drogę symulacji zachowania sieci i defekty wykrywania przez tę sekwencję. Metoda ta wymaga posiadania dobrego programu symulacji defektów, ponieważ ilość iteracji przy poszukiwaniu testu efektywnego na ogół zwiększa czas obliczeń ponad dopuszczalną wartość
- **synteza:** dla danego defektu określa się jedną lub kilka kombinacji detekcyjnych śledząc działanie i (lub) wewnętrzną strukturę sieci. Tutaj obliczenia są prostsze niż w przypadku analizy.



Niestety algorytmy efektywnej syntezy dającej się zastosować do wszystkich typów sieci ciągle jeszcze nie są precyzyjne

- **analiza i synteza:** obie poprzednie metody można mieszać. W pierwszym etapie sekwencja wykrywająca dany defekt określana jest drogą syntezy; drugi etap to określenie drogą analizy wszystkich defektów wykrywanych przez tę sekwencję. Ten proces jest powtarzany dopóty, dopóki nie otrzymamy sekwencji testowej dla wszystkich potencjalnych defektów, których istnienie założyliśmy.

Zasada probabilistyczna (wynikają z niej dwie następujące procedury):

- pierwsza, oparta na wspomnianej wyżej metodzie analizy, składa się w zasadzie z poszukiwania drogą symulacji defektów wykrywanych przez testy, które jednak nie są ustalone z góry, jak w zasadzie deterministycznej, lecz dostarczane przez generator sekwencji przypadkowych; innymi słowy działamy metodą Monte Carlo, w której analiza odpowiada programowi symulacji sieci logicznej zawierającej defekty
- druga polega na podawaniu na wejścia sieci, w chwili efektywnego testu wybranych losowo sekwencji testowych. W tym przypadku nie chodzi już o poszukiwanie szczególnych kombinacji testowych związanych z listą danych defektów, lecz po prostu o obliczanie minimalnej ilości kombinacji niezbędnych dla upewnienia się, że wszystkie defekty obecne w sieci będą wykryte z danym prawdopodobieństwem. Określenie tej minimalnej ilości nie wydaje się możliwe w praktyce.

METODY SYNTEZY

Metody syntezy podzielić można na dwie kategorie: metody zwane funkcjonalnymi, w których rozważa się jedynie funkcję sieci i metody zwane strukturalnymi, które obok funkcji uwzględniają topologię lub strukturę sieci.

Do pierwszej kategorii zalicza się metodę porównywania tablic prawdy, metodę uzupełnień oraz metodę „przecięć" używaną jednak tylko dla sieci kombinacyjnych i interesującą dziś jedynie ze względów historycznych. Trzeba także wymienić metodę prostych implikantów, znajdującą zastosowanie dla układów tranzystorowych MOS.

Kategoria metod strukturalnych zawiera w zasadzie metody określania sekwencji wejściowej zdolnej „uczulić" ścieżki prowadzące od miejsca defektu do pierwotnych wyjść sieci. Poszukiwanie tych sekwencji odbywać się może na dwa sposoby: algebraicznie lub heurystycznie.

Metody algebraiczne

Metody algebraiczne są z pewnością najstarsze i najliczniejsze; wspomnieć można prace Eldreda, Poage'a, metody normalnej postaci równoważnej, SPOOF i różnice boole'owskie.

Choć eleganckie z teoretycznego punktu widzenia, wszystkie te metody dają się zastosować efektywnie jedynie do

kombinacyjnych sieci o ograniczonych rozmiarach. Przy użyciu komputera o odpowiedniej mocy wydaje się jednak możliwe ich praktyczne zastosowanie dla układów scalonych średniej i wielkiej integracji, mogą być one także interesujące jako podstawa rozwijania metod uczulania ścieżek propagacji skutków defektu.

Metody heurystyczne

Ten typ metod stosowany jest w większości przemysłowych systemów generacji sekwencji testowych. Stosowane są zazwyczaj dwie podstawowe operacje:

- propagacja skutków defektu w kierunku pierwotnych wyjść sieci wzdłuż jednej lub wielu ścieżek; ta faza zwana jest propagacją „do przodu"
- propagacja w kierunku pierwotnych wejść sieci wyników ustaleń dokonanych podczas fazy „do przodu"; innymi słowy poszukiwanie wartości logicznych, które powinny pojawić się na liniach sieci, by umożliwić propagację „do przodu". Etap ten zwany jest propagacją „do tyłu". Metody oparte na tych podstawach różnią się zasadniczo sposobami wykonywania operacji „do przodu" i „do tyłu", liczbą ścieżek jednocześnie uczulanych i kryteriami stosowanymi przy wyborze warunków uczulania, które z reguły mogą być bardzo liczne.

Do tej kategorii zalicza się uczulanie jednowymiarowe (jedna tylko ścieżka) D. B. Armstronga (1966) i klasyczny D algorytm J. P. Rotha symulacji wielowymiarowej. Prace te są ważnym krokiem w rozwoju teorii syntezy sekwencji testowych. Chociaż stosują się one do sieci czysto kombinacyjnych, technika uczulania rozszerzona została na przypadki sieci sekwencyjnych synchronicznych, później asynchronicznych.
Ostatnio wprowadza się liczne usprawnienia tych metod: zwiększenie liczby zmiennych celem opisania wszystkich warunków uczulenia, uwzględnienie warunków wyścigów i hazardów oraz charakterystyk dynamicznych elementów, ocena a priori korzyści wynikających z określonego wyboru warunków uczulenia.

METODY ANALIZY

Wydaje się, że przeważają tu dwie metody: z jednej strony prosta symulacja defektów dla danej sekwencji wejściowej, z drugiej — dedukcja na podstawie zachowania się „zdrowej" sieci listy defektów wykrywanych przez daną sekwencję wejściową; te dwie techniki związane są odpowiednio z symulacją równoległą i dedukcyjną.

W symulacji równoległej sieć „zdrowa" oraz N sieci otrzymywanych przez wprowadzenie explicite N pojedynczych defektów do sieci rozważanej są symulowane jednocześnie (N zależy od liczby bitów, którymi można operować w komputerze prowadzącym obliczenia). Technika ta jest obecnie najpopularniejsza.

W symulacji dedukcyjnej symulowana jest jedynie sieć nie zawierająca defektów dla każdej kombinacji wejściowej, a otrzymywane rezultaty są po każdym kroku, a także w trakcie samego przebiegu symulacyjnego wykorzystywane dla określenia zestawu defektów wykrywanych przez użytą sekwencję wejściową. Stosuje się także symulację sieci dla każdej konfiguracji defektów. Dla tego typu symulacji proponować można, jak się zdaje dwa typy algorytmów:
— algorytmy wynikające z zasady uczulania ścieżek propagacji (patrz J. P. Roth 1967), dobrze nadające się do sieci czysto kombinacyjnych, choć istnieją rozszerzenia na sieci sekwencyjne. Ich główną wadą jest niemożliwość prostego uwzględnienia warunków dynamicznych — najbardziej obiecujące ostatnio publikowane algorytmy określenia i propagacji listy defektów.
Zastosowania programów symulacyjnych typu dedukcyjnego są jeszcze zbyt świeże, aby pokusić się o kategoryczne porównanie przewag dwóch technik symulacyjnych. Jednakże ostatnie studia porównawcze zdają się wskazywać, że symulacja dedukcyjna jest bardziej efektywna dla „dużych" sieci „słabo" sekwencyjnych, natomiast symulacja równoległa dla „małych" sieci „silnie" sekwencyjnych. Z drugiej strony praktyczna implementacja symulatora dedukcyjnego jest oczywiście trudniejsza niż symulatora równoległego. Warto dodać, że te dwa typy symulatorów nie wykluczają się wzajemnie.

WNIOSKI

Analiza literatury związanej z tematem jasno wykazuje, że ogólny problem detekcji i lokalizacji defektów w sieciach logicznych jest daleki od rozwiązania. Ocenę te potwierdzają liczne przykłady sieci i systemów przemysłowych, w których stwierdza się obecność defektów wbrew wynikom teoretycznie kompletnych testów przygotowanych w oparciu o aktualnie stosowane metody generacji sekwencji testowych.
Wiele podstawowych problemów pozostaje ciągle nierozwiązanych np.:

- ułożenie zadowalającego algorytmu detekcji i lokalizacji defektów w sieciach sekwencyjnych o istotnych rozmiarach,
- zwiększenie gamy modelowanych defektów, co oznacza uwolnienie się od dwóch klasycznych założeń (defekty sklejeń i defekty pojedyncze) będących spadkiem po technologiach już zarzuconych (DTL, RTL). W szczególności istotna jest możliwość uwzględniania defektów wielokrotnych i defektów przemijających.

Większość prac dotyczy jedynie detekcji pojedynczych sklejeń;

- prace dotyczące lokalizacji defektów pojedynczych oraz detekcji i diagnostyki defektów wielokrotnych są o wiele mniej liczne
- automatyczne określanie pomocniczych punktów testowych dające stosunkowo większe ułatwienie testu niż utrudnienie sterowania
- unifikacja teorii testów i diagnostyki.

Wyodrębnić można kilka kierunków badań zmierzających do rozwiązania tych problemów:

- ulepszenie teorii sieci logicznych zawierających defekty; w tym kierunku idą techniki symulacji uszkodzonych sieci logicznych oraz techniki propagacji skutków defektów w sieciach zbudowanych ze złożonych modułów
- redukcja złożoności problemu detekcji i lokalizacji defektów przez wprowadzenie nowych koncepcyjnie technik dla sieci logicznych, poprzez wybór:

— konfiguracji najkorzystniejszej z punktu widzenia testów i diagnostyki,
— struktur łatwo testowalnych (obwody iteracyjne, układy samotestowalne, układy bez redundancji itp.).

Ostatni kierunek — poszukiwanie rozstrzygnięć problemów testowych, które nie dają rozwiązań rzeczywiście ekonomicznych, w sferze nowych podejść koncepcyjnych — wydaje się wielce obiecujący.

Tłumaczył i opracował: Jan Klimowicz

Z KRAJU

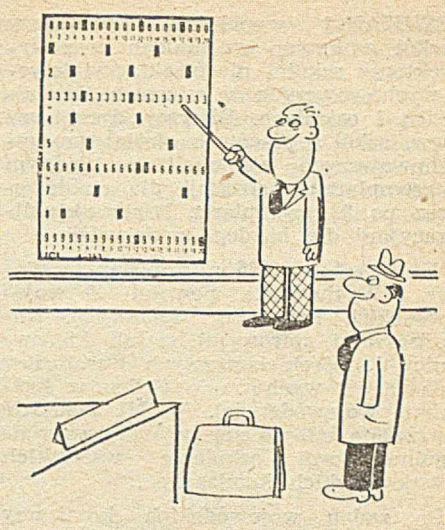
Informatycy na nartach

W tym roku narciarskie mistrzowstwa informatyków odbyły się już po raz czwarty; tym razem w dniach od 25 do 27 marca w Karpaczu. Rozegrano trzy konkurencje: slalom gigant, slalom specjalny oraz bankiet. Slalom gigant, traktowany jako sprawdzian indywidualny, wśród kobiet zakończył się zwycięstem Elżbiety Bromy-Wrzesień z warszawskiego IMM. Z mężczyzn najlepszy był Jacek Hałatek, przedstawiciel organizującego całą imprezę ZETO Jelenia Góra. Drużywono triumfowały panie z ZETO Katowice oraz panowie z MERA Wrocław.

Startowało 75 osób, w tym także reprezentanci Gdyni, Białegostoku, Krakowa, Kielc, Poznania, Lublina i Koszalina. Jak z tego wynika siła naszego informatycznego narciarstwa rozłożona jest dość równomiernie pomiędzy różne ośrodki.

Slalom specjalny, rozgrywany dla urozmaicenia zawodów systemem pucharowym wygrała drużyna ZETO Jelenia Góra. Na bankiecie sukcesy odnosili przedstawiciele miejscowych władz.
Informujemy o tym bezstronie mimo, że przedstawiciele naszego czasopisma nie otrzymali, niestety, zaproszenia na zawody. (mh)

EWIKAN 2
Elektroniczny podsystem ewidencji kadr nauczycielskich w Polsce



Ministerstwo Oświaty i Wychowania postanowiło w 1973 r. po raz pierwszy dokonać powszechnej ewidencji kadr nauczycielskich w Polsce. Ewidencji podlegało 450 000 osób zatrudnionych w szkołach podstawowych i średnich, placówkach oświatowo-wychowawczych i administracji szkolnej.

Opis podsystemu EWIKAN 1 oraz warunki organizacyjno-techniczne wynikające z przeprowadzonej ewidencji kadr nauczycielskich w 1973 r. zostały przedstawione w Informatyce nr 2/75.

Po zakończeniu spisu kadr nauczycielskich EWIKAN 1 przystąpiono do redakcyjnego opracowywania wyników w taką postać, aby były one czytelne dla szerokiej rzeszy pracowników oświatowo-wychowawczych. W rezultacie tych prac powstało osiemnaście zeszytów problemowych, które zostały rozprowadzone do wszystkich ogniw oświatowo-wychowawczych na szczeblu wojewódzkim i centralnym.

Zasady organizacyjne EWIKAN-2

W początkowych założeniach, EWIKAN 2 miał spełniać funkcję aktualizacyjną zbiorów EWIKAN 1. Jednak po głębszej analizie zasad organizacyjnych, struktury zbiorów oraz jakości zawartych w nich informacji zdecydowano, że należy dokonać powtórnej ewidencji nauczycieli celem utworzenia optymalnej kartoteki nadającej się do dalszych aktualizacji.

Tak więc przystąpiono w lutym ub. r. do opracowywania zmodyfikowanego podsystemu informatycznego pod nazwą EWIKAN-2.

Zasadniczej weryfikacji uległ kwestionariusz nauczyciela, w którym z 83 pytań zostawiono tylko 23. W grupie 23 pytań zawarto wszystkie niezbędne informacje, dotyczące nauczycieli pełno-zatrudnionych, niepełnozatrudnionych, zwolnionych od zajęć dydaktycznych i urlopowanych. Nowy arkusz nazwano „kartą aktualizacyjną danych osobowych nauczyciela" i jest czterokrotnie mniejszy od poprzedniego kwestionariusza. Ponadto jest on bardzo czytelny, uwzględniono w nim niezbędne pola do perforacji, a poszczególne pytania rozplanowano w taki sposób, aby z pierwszej strony można było uzyskać dwie, a z drugiej jedną kartę dziurkowaną. Tak więc z jednej karty aktualizacyjnej uzyskuje się 3 karty dziurkowane. Karta aktualizacyjna nauczyciela EWIKAN-2 zawiera następujące pytania:

1. Symbol aktualizacji (skąd przybył do szkoły i dokąd odszedł)
2. Dane identyfikujące (numer teczki, numer arkusza w teczce oraz numer dowodu osobistego)
3. Nazwisko i imię
4. Status (pełnozatrudniony, niepełnozatrudniony, zwolniony od zajęć dydaktycznych, urlopowany)
5. Numer szkoły (symbole: województwa, gminy, szkoły, wielkość miejscowości, resortu oraz typu szkoły)
6. Data urodzenia
7. Staż pracy pedagogicznej
8. Data zatrudnienia w obecnym miejscu pracy
9. Wykształcenie (poziom wykształcenia)
10. Kierunek ukończonych studiów nauczycielskich i wyższych
11. Szkoła wyższa, w której aktualnie studiuje, kierunek oraz rok studiów
12. Doskonalenie zawodowe (kursy)
13. Warunki mieszkaniowe (liczba izb, liczba osób, forma własności, charakter mieszkania)
14. Czas dojazdu i dojścia do pracy w jedną stronę
15. Czy mieszka w miejscowości będącej siedzibą pracy?
16. Charakter zatrudnienia (stanowisko i funkcja)
17. Uposażenie miesięczne
18. Praca dodatkowa
19. Realizowany przydział godzin dydaktycznych, w tym ponadwymiarowych
20. Zniżki godzin
21. Realizowany przydział godzin według przedmiotów i typów szkół
22. Podstawowe miejsce pracy nauczyciela niepełnozatrudnionego.
23. Rodzaj wymiaru zatrudnienia nauczyciela niepełnozatrudnionego

Do karty aktualizacyjnej opracowano instrukcję jej wypełnienia, która zawiera wszystkie niezbędne elementy wraz z pełnym wykazem stosowanych kodów i symboli.

Jednym z najważniejszych elementów całego podsystemu EWIKAN-2, a w szczególności karty aktualizacyjnej jest ustalenie identyfikatora osobowego każdego nauczyciela. Poza sprawą identyfikacji samego nauczyciela, istnieje również problem identyfikacji tzw. „teczek" [1] w czasie realizowanego spisu. System identyfikacji nauczycieli i „teczek" ma duże znaczenie przy kontroli poprawności zapisów podczas przeprowadzonej aktualizacji wymienionych danych, a także po jej zakończeniu.

Zastosowany w EWIKAN-2 system zdał w pełni [2] egzamin i może być również wykorzystany w realizacji całego szeregu innych systemów informatycznych.

Ustalono trzy stopnie identyfikacji:
a) seria i numer dowodu osobistego nauczyciela (seria 2 litery, numer 7 cyfr)

b) numer karty aktualizacyjnej w teczce (od 00 do 99 dla każdej teczki)
c) numer teczki (od 001 do 999 dla każdego komisarza spisowego).

Organizacja ewidencji nauczycieli (aktualizacja) EWIKAN-2

Po przygotowaniu podstawowych problemów wchodzących w zakres EWIKAN-2, np.:

— zakresu informacji niezbędnych w procesie aktualizacji
— wzorca karty aktualizacyjnej
— kodów i symboli stosowanych w aktualizacji
— metod identyfikacji i kontroli poprawności zapisów na karcie aktualizacyjnej
— instrukcji wypełniania karty aktualizacyjnej
— rozplanowania informacji na kartach perforowanych,

przystąpiono do przygotowania przeprowadzenia akcji aktualizacyjnej.

W pierwszej kolejności wydrukowano niezbędną ilość kart aktualizacyjnych (po dwie dla każdego nauczyciela + rezerwa).

Następnie wydrukowano odpowiednią ilość instrukcji wypełniania karty aktualizacyjnej. Zgodnie z przyjętymi zasadami, karty aktualizacyjne w

[1] „Teczka" — zbiór 100 ankiet ponumerowanych od 00 do 99 z nadanym symbolem gminy. „Teczki" zakładali komisarze spisowi.

[2] Tak pod względem organizacyjnym, jak i programowym.

EWIKAN-2 wypełniają dyrektorzy szkół gminnych we wsiach oraz dyrektorzy szkół i placówek oświatowo-wychowawczych w miastach, dzielnicach i osiedlach. Wszyscy dyrektorzy ww. szkół i placówek oświatowo-wychowawczych otrzymali po jednym egzemplarzu instrukcji dla siebie oraz po 2 egzemplarze karty aktualizacyjnej dla każdego nauczyciela.

We wrześniu 1975 r. minister Oświaty i Wychowania powołał 49 wojewódzkich komisarzy spisowych. Komisarzami spisowymi zostali kierownicy komórek kadrowych kuratoriów oświaty i wychowania (jeden w każdym województwie). W III dekadzie września odbyło się w Warszawie ogólnokrajowe szkolenie wszystkich wojewódzkich komisarzy .

Do zadań wojewódzkich komisarzy należało:

— przeprowadzenie szkolenia dyrektorów szkół i placówek oświatowo-wychowawczych na swoim terenie
— sprawdzenie czy wszystkie szkoły otrzymały niezbędną ilość kart aktualizacyjnych i instrukcji
— nadzór nad przeprowadzoną akcją aktualizacyjną, która odbyła się w dniach 1—15 października 1975 r.
— bezpośredni kontakt z Ministerstwem Oświaty w Warszawie oraz Zakładem Elektronicznej Techniki Obliczeniowej w Kielcach w sprawach dotyczących wypełniania kart aktualizacyjnych oraz udzielania niezbędnych informacji dyrektorom szkół
— odbiór w dniach 15—20 października 1975 r. kart aktualizacyjnych od dyrektorów szkół ze swojego terenu
— kompletowanie kart aktualizacyjnych w teczki oraz symbolizowanie ankiet i teczek w dniach 20—25 października
— przekazanie kompletu teczek (opisowych i zasymbolizowanych) ze swojego województwa do ośrodka obliczeniowego w terminie do 30 października
— uzyskanie pierwszego tabulogramu błędów z ośrodka obliczeniowego w terminie do 10 grudnia 1975 r.
— wykonanie pierwszej poprawy błędów i przekazanie jej wyników do ośrodka obliczeniowego w terminie do 20 grudnia
— uzyskanie drugiego tabulogramu błędów z ośrodka obliczeniowego w terminie do 30 grudnia 1975 r.
— wykonanie poprawy błędów i przekazanie jej wyników do ośrodka obliczeniowego w terminie do 5 stycznia 1976 r.

Tak więc komisarze spisowi mieli obowiązek wykonać cały szereg czynności od momentu rozpoczęcia spisu do chwili całkowitej poprawy błędów, które powstały na kartach aktualizacyjnych w czasie przeprowadzonej aktualizacji.

Należy podkreślić, że okres ten (3 miesiące) obejmował również perforację kart (1,4 mln) oraz całość wstępnego przetwarzania i kontroli danych.

Ministerstwo Oświaty i Wychowania nie posiada własnego zaplecza komputerowego i w związku z tym musi korzystać z ośrodków obliczeniowych usługowych zgrupowanych w Zjednoczeniu Informatyki. W tym celu zostało podpisane porozumienie między Ministerstwem Oświaty i Wychowania a Zjednoczeniem Informatyki. W wyniku nawiązanej współpracy Zjednoczenie Informatyki przyjęło na siebie następujące zadania:

— ustalenie głównego wykonawcy systemu, którym został Zakład Elektronicznej Techniki Obliczeniowej w Kielcach
— wykonanie dokumentacji projektowej systemu przez Zakład Elektronicznej Techniki Obliczeniowej w Kielcach
— oprogramowanie całego ssytemu oraz opracowanie dokumentacji programowej i eksploatacyjnej przez ZETO Kielce
— wykonanie perforacji kart maszynowych około 1,4 mln sztuk w około 20 ośrodkach obliczeniowych w całej Polsce w ciągu 1 miesiąca od daty dostarczenia kart aktualizacyjnych
— wczytanie kart maszynowych do pamięci komputerów, założenie zbiorów wojewódzkich, wydruk tabulogramów błędów, weryfikacja zbiorów, wydruk drugiego tabulogramu błędów, ponowna weryfikacja w 20 ośrodkach wojewódzkich na komputerach serii ODRA 1300.
— scalanie taśm magnetycznych (system opracowano na konfigurację taśmową komputera z 20 ośrodków obliczeniowych w jeden zbiór w układzie wojewódzkim i krajowym). Pracę tę realizował ośrodek ZETO Kielce
— wydruk wydawnictw końcowych w liczbie 60 tablic o łącznej objętości 14 000 stron również w ZETO Kielce.

Przyjęte przez Zjednoczenie Informatyki i podległe mu ośrodki obliczeniowe zobowiązania zostały w całości zrealizowane.

Zgodnie z omawianymi poprzednio zadaniami, komisarze spisowi w terminie do 5.01.76 r. przesłali wyniki drugiej korekty błędów.

Ośrodki obliczeniowe (20) zostały zobowiązane do ostatecznej weryfikacji zbiorów wojewódzkich i przekazania taśm magnetycznych głównemu wykonawcy tj. ZETO-Kielce w terminie do 15.01.76 r. W zakładzie tym rozpoczęto tworzenie centralnej kartoteki nauczycieli (również na taśmach magnetycznych) oraz produkcję 60 wydawnictw końcowych.

Zakończenie całości prac przewidziano na 30 marca 1976 r., a jej koszt nie powinien przekroczyć 6 mln złotych.

Precyzyjnie opracowany harmonogram realizacji całego przedsięwzięcia, ścisła współpraca Ministerstwa Oświaty z ZETO-Kielce oraz ośrodkami obliczeniowymi w całym kraju, wspólne poczucie odpowiedzialności ludzi wykonujących wszystkie elementy systemu — zdecydowały o powodzeniu całej akcji. Najwyższe słowa uznania należą się jednak głównemu projektantowi systemu oraz jego czterem współpracownikom (ZETO Kielce), którzy opracowali cały system niezwykle szybko, optymalnie i oryginalnie. Według opinii autorów systemu EWIKAN-2 oraz Ministerstwa Oświaty i Wychowania, czas realizacji następnych edycji EWIKAN może być skrócony do czterech miesięcy licząc od daty rozpoczęcia spisu aż do zakończenia produkcji wydawnictw końcowych. Również koszt jego przetwarzania będzie obniżony w następnych latach do 4 mln złotych z czego 2,4 mln to koszt perforacji danych na maszynowych nośnikach informacji.

Technologia przetwarzania podsystemu EWIKAN 2

Organizacja procesu projektowania i oprogramowania

W poprzednich punktach omówiono podstawowy dokument źródłowy tj. kartę aktualizacyjną danych osobowych nauczyciela, w oparciu o którą zaprojektowano maszynowy nośnik informacji — kartę dziurkowaną. Ponieważ opracowywanie karty aktualizacyjnej odbywało się równolegle z zaprojektowaniem maszynowych nośników informacji, dlatego też uwzględniono w obydwu zagadnieniach wymogi elektronicznej techniki obliczeniowej oraz potrzeby resortu oświaty i wychowania. Szczególną uwagę zwrócono na:

— prostotę i łatwość wypełniania kart aktualizacyjnych przez dyrektorów szkół
— łatwość perforacji informacji na kartach dziurkowanych

Jak już wspomniano z jednej karty aktualizacyjnej powstają trzy typy osiemdziesięciokolumnowych kart dziurkowanych.

Poza trzema typami kart opracowano jeszcze jeden typ nazwany swobodnym (TYP ϕ). W typie swobodnym perforuje się identycznie jak w poprzednich kartach — dane identyfikujące kartę — natomiast pozostałe informacje perforuje się dowolnie uzależniając ich treść tylko od numeru pytania z karty aktualizacyjnej nauczyciela. Wprowadzenie typu swobodnego ma na celu skrócenie czasu perforacji oraz zmniejszenie liczby kart perforowanych.

Po zaprojektowaniu kart dziurkowanych przystąpiono do opracowania dokumentacji projektu technicznego. Ze względu na wyposażenie wszystkich usługowych ośrodków obliczeniowych (20) realizujących system EWIKAN 2 w jednolity park komputerów serii ODRA 1300 (model ODRA 1304 z pamięcią taśmową typ PT-3 oraz model ODRA 1305), projekt techniczny uwarunkowany został tym sprzętem.

W czasie opracowywania projektu technicznego zaprojektowano wzory 60 wydawnictw końcowych (tabulogramów), które w całości zabezpieczają potrzeby informacyjne resortu oświaty i wychowania. Wszystkie zaprojektowane wydawnictwa podzielono na grupy problemowe, z których w późniejszym czasie zostanie opracowana seria wydawnicza do powszechnego użytku. Całość podsystemu została podzielona na trzy moduły, dla których wykonano następujące ilości programów:

- moduł I — 8 programów w języku PLAN
- moduł II — 3 programy w języku COBOL
- moduł III — 60 programów w języku COBOL.

Organizacja wewnętrzna podsystemu

Podział na moduły niezależnie od przesłanek ogólnych był konieczny w przypadku ogólnokrajowego zasięgu tak w kompletowaniu dokumentów źródłowych jak i w przetwarzaniu w różnych ośrodkach obliczeniowych. Tematyka modułów jest następująca:
I moduł — wczytywanie kart dziurkowanych, podporządkowanie kart, wydruk tabulogramów błędów, weryfikacja zbiorów zawartych na taśmach magnetycznych oraz tworzenie zaktualizowanych kartotek nauczycieli według województw. Prace te wykonują terenowe ośrodki obliczeniowe
II moduł — łączenie kartotek nauczycieli z poszczególnych województw w jedną kartotekę centralną. W module tym przewidziano działanie odwrotne, a mianowicie tworzenie kartotek wojewódzkich z jednej kartoteki centralnej. Wszystkie te prace powinny być wykonane w ośrodku obliczeniowym wiodącym
III — moduł — emisja wydawnictw wynikowych dla potrzeb resortu oświaty i wychowania. Prace te wykonuje się w ośrodku obliczeniowym wiodącym. W module III przewidziano również emisję wydawnictw dla wojewódzkich kuratoriów oświaty i wychowania. Czynności te można wykonać bądź w ośrodku wiodącym, bądź w ośrodkach obliczeniowych terenowych.

Uwagi końcowe

Organizacja wewnętrzna i zewnętrzna całego podsystemu EWIKAN 2 jest dość elastyczna i można ją doskonalić bez dokonywania zasadniczych zmian projektowych i programowych, przy minimalnych kosztach. Już obecnie po przeprowadzeniu całej akcji oraz wykonaniu olbiczeń, autorzy systemu widzą dalsze możliwości jego usprawnienia. Należy sądzić, że EWIKAN 3 będzie podsystemem całkowicie dopracowanym i zorganizowanym tak optymalnie jak tylko to jest możliwe na obecnym etapie nowoczesnego projektowania, programowania i eksploatacji dużych i w związku z tym szczególnie trudnych w realizacji systemów państwowych.

Obecny zbiór danych osobowych nauczycieli czyli tzw. kartoteka główna jest utworzona na taśmach magnetycznych. Życzeniem autorów systemu oraz resortu oświaty i wychowania jest przeniesienie zbioru na system dysków magnetycznych. Być może, że przy realizacji EWIKAN 3, sieć obliczeniowa Zjednoczenia Informatyki będzie dysponowała sprzętem komputerowym wyposażonym w dyski magnetyczne.

Następnym pragnieniem autorów EWIKAN jest tworzenie maszynowych nośników informacji na urządzeniach do klawiaturowej rejestracji danych na taśmach magnetycznych. Wreszcie ostatnim i chyba najbardziej efektywnym rozwiązaniem byłoby przeprojektowanie całego systemu na układ konwersacyjny. Oczywiście uzależnione jest to od posiadania niezbędnego sprzętu do teletransmisji tak na szczeblu centralnym jak i wojewódzkim.

Dopóki jednak nie zostaną zrealizowane powyższe zamierzenia, resort oświaty i wychowania będzie korzystał z wyników dostarczanych z obecnej edycji EWIKAN 2 i następnych. W chwili obecnej informacje zbierane i przetwarzane za pomocą opisanego systemu są jedynym analitycznym i syntetycznym źródłem danych o kadrze nauczycielskiej w Polsce. Dodatkową ich zaletą jest to, że zostały one zebrane równocześnie w całej Polsce i ujmują całą kadrę nauczycielską.

Jest to jedyne i niezmiernie cenne źródło danych potrzebne do badań naukowych nad stanem i rozwojem kadr nauczycielskich.

Opracowali:

mgr Bogumił Stachura
Ministerstwo Oświaty i Wychowania
mgr Stanisław Urbanek
ZETO Kielce

ZE ŚWIATA

Konferencja ICCAS 76 w Göteborgu

Jedną ze specjalistycznych konferencji naukowych organizowanych pod wspólnym protektorem IFIP (Międzynarodowa Federacja Przetwarzania Informacji) oraz IFAC (Międzynarodowa Federacja Automatycznego Sterowania) jest ICCAS (International Conference on Computer Applications in the Automation of Shipyard Operation and Ship Design — Międzynarodowa Konferencja na temat zastosowania komputerów do automatyzacji produkcji stoczniowej i projektowania statków).

Pierwsza konferencja ICCAS odbyła się w 1973 r. w Tokio i jej znaczny sukces spowodował decyzję o celowości kontynuowania następnych konferencji w tej dziedzinie. Funkcję zorganizowania następnej konferencji w 1976 roku powierzono Szwecji, która zajmuje aktualnie drugie miejsce na świecie w produkcji statków. Ze zrozumiałych względów konferencję zlokalizowano w największym szwedzkim ośrodku przemysłu stoczniowego Göteborgu.

Konferencja ICCAS 76 odbywa się w okresie od 9 (środa) do 11 (piątek) czerwca 1976. Dzień poprzedzający konferencję, tj. 8 czerwca przeznaczono na zwiedzanie przez uczestników czołowych zakładów przemysłowych na terenie miasta GÖTEBORG, drugiego co do wielkości ośrodka miejskiego Szwecji. Istnieje możliwość wyboru — zgodnie z głównymi zainteresowaniami uczestnika — spośród 9 propozycji, obejmujących obok zakładów przemysłu stoczniowego, również zarząd portu oraz zakłady przemysłu motoryzacyjnego (VOLVO), łożysk tocznych (SKF) i fotooptycznego (HASSELBLAD). Konferencja odbędzie się w Centrum Kongresowym położonym w śródmieściu Göteborg. Obrady będą odbywały się wyłącznie w języku angielskim. Do wygłoszenia zakwalifikowano łącznie 54 referaty pochodzące z 16 krajów. Tematykę referatów podzielono na 8 obradujących równolegle następujących sekcji tematycznych:

Sekcja Nr 1 problemy ogólne (9 referatów)

Sekcja Nr 2 szkolenie (3 referaty)

Sekcja Nr 3 projektowanie koncepcyjne (8 referatów)

Sekcja Nr 4 projektowanie i analiza (8 referatów)

Sekcja Nr 5 systemy informacyjne kierownictwa (8 referatów)

Sekcja Nr 6 systemy precyzujące produkcję (6 referatów)

Sekcja Nr 7 systemy technologii produkcji (6 referatów)

Sekcja Nr 8 produkcja wspomagana komputerem (6 referatów)

Na sesji plenarnej otwarcia zostanie wygłoszony referat główny na temat „Perspektywy automatyzacji i przetwarzania komputerowego w budownictwie okrętowym — operacje warsztatowe, produkcja statków oraz stosunki międzyludzkie" Tematem plenarnej sesji końcowej będzie dyskusja panelowa pt. „Stocznia, komputer i czynnik ludzki".

Warto odnotować, że konsekwencją doceniania pozycji światowej naszego przemysłu okrętowego jest fakt udziału przedstawicieli Polski, zarówno w Międzynarodowym Komitecie Organizacyjnym, jak i międzynarodowym komitecie programowym konferencji. Nieco skromniejszy jest udział Polaków wśród referentów. Wygłoszą oni łącznie 4 referaty : 1 w sekcji nr 3, 2 w sekcji nr 5 oraz 1 w sekcji nr 8.

Przy okazji należy zwrócić uwagę na rosnące opłaty uczestnictwa, które w przypadku omawianej konferencji wynoszą aż 950 koron szwedzkich (ok. 210 $ USA). Dlatego mimo bliskości położenia może to spowodować bardzo mały udział przedstawicieli naszego kraju wśród uczestników tej bardzo interesującej imprezy. (WK)

Informatyka narzędziem usprawniania organizacji i zarządzania gospodarką miejską województwa gdańskiego

Pierwsze wdrożenia systemów informatycznych w gospodarce miejskiej województwa gdańskiego przypadają na rok 1972, kiedy to w Wojewódzkim Przedsiębiorstwie Komunikacyjnym Gdańsk-Gdynia wprowadzono system gospodarki materiałowej oraz zorganizowano własną stację perforacji.

Dynamiczny rozwój zastosowań systemów informatycznych w zarządzaniu gospodarką komunalną i mieszkaniową w województwie gdańskim wiąże się ze zorganizowaniem w roku 1973 Terenowego Ośrodka Informatyki Wojewódzkiego Zjednoczenia Gospodarki Komunalnej i Mieszkaniowej. Ośrodkowi przydzielono zadanie opracowania, wdrażania i rozpowszechniania systemów informatycznych w gospodarce i administracji terenowej. Od początku zdawano sobie sprawę, że prawidłowa i szybka realizacja postawionych celów będzie uwarunkowana rozwojem kadry specjalistów w Terenowym Ośrodku Informatyki, przygotowaniem użytkowników systemów do stosowania informatyki w praktyce, a także — zważywszy na specyfikę województwa jako centrum makroregionu nadmorskiego — szybkie przygotowanie warunków do instalacji własnego komputera.

Już pod koniec 1973 roku stan zatrudnienia w Terenowym Ośrodku Informatyki wynosi 25 osób, głównie projektantów i programistów; w końcu 1974 roku — 61 osób, a w początkach 1976 r. — 87 osób. W roku 1974 zorganizowano własną stację perforacji, która obecnie jest wyposażona w 30 maszyn, w tym 90% to dziurkarki i sprawdzarki alfanumeryczne kart typu SOEMTRON.

Struktura kadrowa Ośrodka według stanu na początek roku 1976 przedstawia się następująco:

- analitycy systemów, projektanci i programiści 38 osób
- pracownicy stacji maszyn pomocniczych 37 osób
- administracja wraz z dyrekcją 8 osób
- obsługa 4 osoby

Realizując program rozwoju prac projektowych, TOI zabezpieczał sukcesywnie potrzeby w zakresie niezbędnej mocy obliczeniowej w różnych ośrodkach informatycznych, zlokalizowanych na terenie województwa gdańskiego. Dzięki temu było możliwe nie tylko opracowanie, lecz również wdrożenie szeregu programów i systemów. W okresie swojego istnienia, tj. 2,5 lat Terenowy Ośrodek opracował kilka własnych systemów, a w szczególności:

1. SYSTEM PROGRAMOWANIA I PLANOWANIA REMONTÓW BUDYNKÓW MIESZKALNYCH

Zakres systemu obejmuje programowanie i planowanie kosztów remontów według lokali obiektów i robót remontowych, a także typowanie budynków do remontów według przyjętych kryteriów. W toku opracowywania jest moduł systemu dotyczący kosztorysowania robót remontowych oraz planowania podstawowych środków produkcji.

W roku 1973 wykorzystano omawiany system do opracowania zbiorczego programu remontów budynków w skali całego województwa. W roku 1975 rozporzęto wdrażanie w dwóch przedsiębiorstwach drugiego modułu systemu dotyczącego planowania remontów w skali przedsiębiorstwa.

2. System rozliczeń czynszów obejmuje takie podstawowe problemy, jak:

- gospodarkę mieszkaniową
- ewidencję wpłat i zaległości
- wydruki rachunków za najem lokali
- sprawozdawczość z windykacji czynszów.

Zakres wdrożenia obejmuje 6 przedsiębiorstw gospodarki mieszkaniowej Trójmiasta i Elbląga. Aktualnie prowadzone są prace wdrożeniowe w Warszawie i w Łodzi.

System został uznany za system resortowy.

3. System sprawozdawczości i analiz oparty o własne sparametryzowane oprogramowanie umożliwia automatyzację dowolnej sprawozdawczości statystycznej i wewnętrznej oraz analizę danych w formie emisji różnorodnych wskaźników i zestawień w zależności od potrzeb użytkownika i opracowanego modelu danych wynikowych. Zakres wdrożenia obejmuje Wojewódzkie Zjednoczenie Gospodarki Komunalnej i Mieszkaniowej oraz Wojewódzką Komisję Planowania.

4. System ewidencji zatrudnienia wdrożony aktualnie w trzech przedsiębiorstwach.

5. System ewidencji płac znajdujący się na etapie wdrażania i obejmujący następujące podstawowe problemy:

- obliczanie zarobków i emisję list płac
- analizę płac
- ewidencję księgową robocizny z uwzględnieniem rozliczania kosztów płac
- sprawozdawczość płacową.

6. System ewidencji przedmiotów nietrwałych informujący o miejscu użytkowania przedmiotu nietrwałego i emitujący kartoteki wyposażenia pracowników, wdrożony w trzech przedsiębiorstwach.

W dążeniu do przyśpieszenia zastosowań informatyki w gospodarce komunalnej i mieszkaniowej przyjęto zasadę adaptacji względnie wdrożenia opracowanych przez inne ośrodki systemów informatycznych. Tak np. zaadaptowano system ewidencji gospodarki materiałowej eksploatowany w budownictwie województwa gdańskiego, poszerzając jego funkcje, zwłaszcza w zakresie prowadzenia centralnego indeksu materiałowego.

System wdrożono w dwudziestu przedsiębiorstwach gospodarki komunalnej i mieszkaniowej województw gdańskiego i elbląskiego.

Wdrożono także z powodzeniem w roku 1975 system finansowo—kosztowy, opracowany w Gdańskich Zakładach Nawozów Fosforowych, automatyzujący całą księgowość analityczną i syntetyczną oraz rozliczanie kosztów w przedsiębiorstwie.

Głównym zadaniem Terenowego Ośrodka Informatyki w roku bieżącym jest maksymalne rozpowszechnianie opracowanych i wdrożonych systemów informatycznych. Aktualny ograniczony potencjał obliczeniowy seci ZETO i innych ośrodków obliczeniowych na naszym terenie uniemożliwia jednakże pełną realizację programu wdrożeń i eksploatacji systemów informatycznych w gospodarce i administracji terenowej.

W latach 1973—1975 np. Terenowy Ośrodek Informatyki zmuszony był korzystać z mocy obliczeniowej aż 5 ośrodków obliczeniowych, co w sposób istotny utrudniało pracę wdrożeniowców i ją dezorganizowało.

Problem wzrostu efektywności zastosowań będzie rozwiązany już w roku bieżącym z chwilą zainstalowania w TOI własnego komputera typu ODRA 1305.

Dalszy rozwój Terenowego Ośrodka Informatyki związany jest ściśle z potrzebami w zakresie zastosowania informatyki do zarządzania w jednostkach gospodarczych i administracji terenowej województwa gdańskiego i województw sąsiadujących, zwłaszcza elbląskiego i słupskiego.

Obecnie ogólny potencjał mocy obliczeniowej EMC wykorzystywany na potrzeby gospodarki komunalnej i mieszkaniowej w województwie gdańskim i częściowo elbląskim wynosi ponad 250 godzin miesięcznie przy ograniczonym dostępie do komputera.

Możliwości rozpowszechniania systemów zabezpieczone odpowiednim potencjałem ludzkim mogłyby się zamknąć liczbą około 500 godzin pracy komputera miesięcznie pod koniec bieżącego roku, pod warunkiem dostarczenia odpowiedniej, skoncentrowanej w jednym miejscu, mocy obliczeniowej.

Szacuje się, że docelowe potrzeby mocy obliczeniowej do roku 1980, liczone w godzinach pracy komputera o pamięci 64 K w standardowym zestawie, dostarczanym przez producenta zamkną się liczbą około 2000 godzin miesięcznie. Do obsłużenia tak dużych potrzeb niezbędne będzie zwiększenie konfiguracji komputera, wyposażenie w urządzenia pamięci o szybkim dostępnie oraz zainstalowanie około roku 1979 drugiego komputera.

Zakłada się, że równolegle z zainstalowaniem w Terenowym Ośrodku Informatyki dużych zestawów komputerów, będzie się instalować bezpośrednio u niektórych użytkowników minikomputery produkcji krajowej.

Dotychczasowe doświadczenia z zastosowań informatyki w zarządzaniu gospodarką komunalną i mieszkaniową potwierdzają przewidywania w odniesieniu do efektywności wdrożeń.

Efekty z zastosowań informatyki obserwuje się zwłaszcza w zakresie usprawniania zarządzania w przesiębiorstwach. Można to zauważyć choćby na przykładzie przedsiębiorstwa gospodarki komunalnej i mieszkaniowej w Gdyni oraz Wojewódzkiego Przedsiębiorstwa Komunikacyjnego, gdzie zakres wdrożeń systemów informatycznych jest największy.

Równolegle z wdrażaniem tworzy się przede wszystkim sukcesywnie warunki umożliwiające m.in. spłaszczenie struktur organizacyjnych. Obserwuje się to na przykładzie wdrożenia systemu rozliczania czynszów, gdzie jednym z efektów jest likwidacja niektórych komórek organizacyjnych.

Zwiększony zakres informacji i pogłębiona analiza zjawisk gospodarczych w przedsiębiorstwie w wyniku wdrażania systemów informatycznych stwarzają warunki dla lepszego gospodarowania środkami produkcji.

Jednym z podstawowych utrudnień w zakresie zastosowania informatyki w zarządzaniu, z którymi na codzień mają do czynienia informatycy, jest w większości przypadków konieczność dostosowywania się do istniejących warunków i ograniczeń wewnętrznych i zewnętrznych zarządzania w jednostkach gospodarczych. Projektowanie radykalnych zmian w istniejących strukturach organizacyjnych, aczkolwiek zgodnych z teorią zarządzania, kończy się zwykle niepowodzeniem na etapie wdrażania.

Dlatego też w strategii wdrażania systemów informatycznych w gospodarce miejskiej w województwie gdańskim zakłada się sukcesywne przechodzenie od zmian ilościowych do zmian jakościowych o charakterze strukturalnym. Jest to jednakże proces długotrwały, wymagający od początku stosowania wyższych jakościowo metod projektowania oraz bieżącej konserwacji i aidaptacji wdrożonych i eksploatowanych systemów do zmieniających się warunków. Kolejnym utrudnieniem przy opracowywaniu systemów informatycznych w jednostkach gospodarki miejskiej i administracji terenowej jest brak tradycji i doświadczeń u użytkowników w zakresie informatyki oraz znajomości problemów branżowych u projektantów systemów informatycznych.

Pomimo istnienia przedstawionych wyżej problemów, których rozwiązanie warunkuje proces wdrażania informatyki w gospodarce miejskiej i administracji terenowej — dotychczasowe pozytywne doświadczenia informatyków województwa gdańskiego, zatrudnionych w Terenowym Ośrodku Informatyki w Gdańsku, pozwalają z ufnością patrzeć na proces wdrażania.

Jednym z ważnych elementów stymulujących rozwój zastosowań informatyki w gospodarce miejskiej i administracji terenowej jest duża różnorodność interesujących tematów do opracowania — różnorodność — przyciągająca ludzi o wysokich kwalifikacjach i ambicjach zawodowych.

Jak w każdej dziedzinie działalności, tak również i tutaj o powodzeniu decydują ludzie, ich zaangażowanie i twórcza praca.

Adam Langer
Urząd Wojewódzki Gdańsk
Jan Nowak
Terenowy Ośrodek Informatyki w Gdańsku

ZE ŚWIATA

Pamięć o pojemności 1 tryliona bitów

Firma MIEROBIT (Lexington, USA) zapowiada wprowadzenie na rynek w okresie najbliższych 3 lat modułu pamięci EBAM-950 o pojemności 1 tryliona bitów, czasie dostępu 1 s oraz możliwościami zapisu i odczytu. Aktualnie MICROBIT produkuje wersję EBAM o pojemności 128 k przeznaczoną do komputera CDC STAR. Pierwsze jej dostawy zapowiedziano na czerwiec 1976 roku. W tym samym terminie przewiduje się zakończenie prac nad konstrukcją lampy pamięciowej o pojemności 4 megabajtów. Prace te stanowią kolejne etapy realizacji projektu EBAM 950, której zakończenie spodziewane jest za 30 miesięcy.

Wymiary zewnętrzne jednostki EBAM 950 będą wynosiły: 5×2×2 stopy (1 stopa = 305 mm). Cena 1 bitu pamięci nie powinna przekraczać ceny 1 bitu wymiennej pamięci dyskowej. Przewiduje się, że EBAM-950 będzie stosowana jako pamięć przejściowa między dyskiem a półprzewodnikową lub rdzeniową pamięcią główną. W przypadku gdy producent zdoła utrzymać cenę 1 bitu EBAM-950 na zapowiedzianym poziomie, można się spodziewać, że wyeliminuje ona całkowicie pamięci dyskowe w systemach *on line* z dużymi jednostkami centralnymi.

Lampy pamięciowe typu EBAM mają zaledwie 20 cali (1 cal = 25,4 mm) długości. Zapis i odczyt zapisu następują za pomocą strumienia elektronów emitowanego przez wyrzutnię elektronów skierowaną na tarczę pamięciową, wykonaną z półprzewodnikowych elementów MOS. Rozwiązanie to eliminuje kosztowne i skomplikowane połączenia występujące w pamięciach rdzeniowych i półprzewodnikowych, tak że połączenie EBAM z CPU następuje za pośrednictwem 2 przewodów. Przełączanie z zapisu na odczyt jest bardzo proste i wymaga jedynie zmiany odchylenia w lampie pamięciowej. Wadą omawianych lamp pamięciowych jest konieczność stosowania ekranów absorbujących promieniowanie, a zwiększających gabaryty lampy.

Najpoważniejszym konkurentem MICROBIT w zakresie produkcji pamięci elektrostatycznych jest GENERAL ELECTRIC, która prowadzi aktualnie prace nad konstrukcją 4 megabajtowej pamięci BEAMOS, GENERAL ELECTRIC zapowiada wyprodukowanie BEAMOS o pojemności 100 megabajtów (800 mln bitów), w której koszt 1 bita zostanie obniżony do 0,004 pensa.

Firma IBM ogranicza natomiast stosowaną technologię pamięci przejściowych do technologii pamięci pęcherzykowej, mimo że jest to równoznaczne z ograniczeniem pojemności pamięci do rzędu milionów bitów.

(Sob.)

The Trillion Bit Memory. Computer Weekly 1975 nr 476/7 s. 24.

25-lecie działalności VUMS

W końcu ub. roku w Pradze odbyła się konferencja związana z 25-leciem działalności Czechosłowackiego Instytutu Badawczego Maszyn Matematycznych (VUMS = Výzkumny ústav matematických stroju), wiodącej placówki naukowo-badawczej przemysłu informatycznego w ČSSR. W konferencji uczestniczyło około 250 zaproszonych gości i pracowników VUMS. Wygłoszono błącznie 43 referaty oraz przeprowadzono dyskusję w dwóch zespołach problemowych. Pierwszy obejmował ogólne problemy rozwoju informatyki, drugi natomiast problemy oceny stanu aktualnego oraz perspektyw rozwoju informatyki w kraju. W wyniku dyskusji pracownicy VUMS zgłosili zobowiązanie, zawierające skonkretyzowane propozycje działań w kierunku podwyższenia efektywności eksploatacji komputerów, a także zacieśnienia współpracy z przemysłem przy opracowaniu projektu technicznego wdrożenia do produkcji seryjnej sprzętu tzw. generacji 3½ oraz współuczestniczenie przy realizacji tego projektu u producenta. (WK)

(wg MAA nr 2/76)

CYBER 72 pracuje w CYFRONECIE

W rzędzie największych z zainstalowanych w kraju komputerów znajdują się dwa produkcji znanej firmy CONTROL DATA CORPORATION. Są to dwa CYBERY 70, oba zainstalowane w ośrodka naukowych z przeznaczeniem stymulowania prac naukowo-badawczych i dydaktycznych. Jeden z nich znajduje się w podwarszawskim Świerku w INSTYTUCIE BADAŃ JĄDROWYCH.

Mnie udało się obejrzeć i opisać drugi CYBER z okazji wizyty w Środowiskowym Centrum Obliczeniowym CYFRONET w Krakowie.

Dwa obiegowe zarzuty zabrałem z sobą do Krakowa. Jeden dotyczył CYBERA, któremu zarzucano, że zamiast wypracowywać zyski marnuje się w Krakowie, drugi — CYFRONETU, że gwoli pofolgowania ambicjom starał się o komputer, który — jeśli już musiał być kupiony do Polski — to po to, aby wypracowywać zyski. Zarzuty o tyle niebłahe, że dość rozpowszechnione. Należało więc zweryfikować je na miejscu.

Tam okazało się, że lista zarzutów i opinii jest dłuższa, niż przypuszczałem. Najczęstsze to:

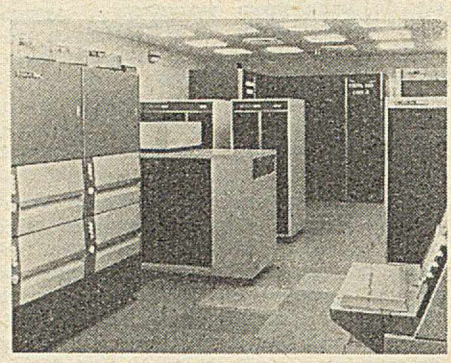
- *po co taki drogi komputer*
- *istnieje dysproporcja mocy obliczeniowej komputera do liczby podłączonych końcówek*
- *należało udostępnić komputer instytucjom pozaakademickim*
- *korzystanie z komputera jest zbyt kosztowne.*

1. Trochę historii albo skąd się wziął CYBER w Krakowie?

Drugie co do wielkości w Polsce środowisko akademickie to właśnie Kraków. Środowisko to, aczkolwiek wielce zasłużone dla rozwoju nauki polskiej, słynne z wielu kapitalnych prac z zakresu badań podstawowych, a także stosowanych w ramach procesu komputeryzacji kraju — w okresie poprzedzającym powstanie CYFRONETU nie było rozpieszczane. Ośrokom uczelnianym udało się wprawdzie zgromadzić nieco sprzętu [1]), ale w zestawieniu z coraz większą ilością i wagą zadań odczuwało się stały jego niedostatek. W trzecim co do wielkości mieście w Polsce, sprzętu było proporcjonalnie znacznie mniej, niż w stolicy, a naukowcy nie bardzo gdzie mieli szukać wsparcia przy realizacji bardziej złożonych prac.

O znaczeniu niektórych zadań, jakie były wykonywane, np. w Akademii Górniczo-Hutniczej dla priorytetowej branży (górnictwo węglowe) nawet nie trzeba pisać. Było oczywiste, że akademickie środowisko Krakowa powinno uzyskać więcej sprzętu informatycznego.

[1]) W sumie: 2 ODRY 1204 — po jednej: 1325, 1305, 1304; jeden komputer MIŃSK 32



Rys. 1. Komputer CDC CYBER 72-16 bez względu na przynależność do II generacji jest nie gorszy od najnowocześniejszych i mało zawodny. Objęty znanymi ograniczeniami embarga. po sporych perturbacjach zainstalowany w Krakowie w następującej konfiguracji:
Procesor centralny — szybkość około 1 mln operacji na sekundę, pamięć operacyjna — 96 k słów (60-bitowych), pamięć zewnętrzna: taśmowa — 8 jednostek, dyskowa — 8 jednostek po 36 mln znaków 6-bitowych, urządzenia wejścia: 2 czytniki kart (1100 kart/min) 1 czytnik — perforator taśmy papierowej, urządzenia wyjścia: 2 drukarki wierszowe (1100 wierszy/ /min), 1 perforator kart.
Zamierzenia CYFRONETU kierują się ku zwiększeniu pamięci operacyjnej do 128 k słów i rozbudowie pamięci zewnętrznych, przede wszystkim dyskowych (marzenia — dyski po 118 mln znaków 6-bitowych). Zakładane uzupełnienia mogłyby nastąpić w związku z przekwalifikowaniem komputera do klasy 72—18. Drugi komputer już by się nie zmieścił. Wystarczy tylko miejsca na uzupełnienia, które do roku 1980 powinny spowodować, że CYFRONET sprosta zapotrzebowaniu na stałych wzrost zadań. Szczęściem już opracowuje się dokumentację dla nowego własnego i już nie prowizorycznego budynku

Rys. 2. Na stanowisko dyrektora ŚCO CYFRONET kolegium Rektorów desygnowało od początku doc. dr hab. Jerzego Kolendowskiego (z prawej). Przedstawiamy go w chwili, gdy CYFRONET wpisawszy na listę użytkowników (zbiorowych i indywidualnych) użytkownika z numerem 500, postanowił ten fakt zaświadczyć i uczcić dyplomem i okolicznościowym goździkiem

Zamiast koncepcji wzmacniania rozproszonych ośrodków uczelnianych, wybrano koncepcję integracji. Podstawową jej implikacją była potrzeba dużego komputera. W końcu roku 1971 — prof. Marian Mięsowicz przedłożył Władzom Centralnym PRL koncepcję Kolegium Rektorów. W sprawę zaangażował się Premier Piotr Jaroszewicz; zapadła decyzja organizowania w Krakowie Środowiskowego Centrum Obliczeniowego.

W połowie roku 1972 oprócz potrzeby i powyższej decyzji — nie było nic.

Mianowano dyrektora CYFRONETU — został nim wychowanek i pracownik AGH doc. dr hab. Jerzy Kolendowski. Lokal zastępczy upatrzono w Instytucie Fizyki Uniwersytetu Jagiellońskiego — tam miał stanąć komputer, tam miano postawić barak dla zespołu. Nawiązano współpracę z bliźniaczym CYFRONETEM Warszawa; przyniosło to później owoce w postaci wykorzystania doświadczeń przy pracach adaptacyjnych pomieszczeń komputera, a także pomogło w szkoleniu kadry. Bowiem podobnie, jak Świerk, tak i CYFRONET Kraków po starannym rozpatrzeniu ofert zdecydował się kupić CYBERA z firmy CDC. Chociaż wśród oferentów nie zabrakło czołowych światowych producentów komputerów, jak IBM, UNIVAC czy francuskiego CII, Komisja oceny ofert zdecydowała się jednak na model 72-16 komputera CYBER.

Wkrótce potem powołano zespół do spraw teledacji i zaczęto szkolić kadrę.

Tylko ten, kto sądzi, że Amerykanie stosują tę samą procedurę przy sprzedaży komputerów, co np. maszyn do szycia, mógł sądzić, że dalej wszystko powinno pójść, jak z płatka. Inni wiedzą, że komputery, zwłaszcza duże, traktowane tam są jako sprzęt strategiczny, a na pewno komputery CDC, na których w USA liczy się inne zadania niż np. listy płac.

Dwukrotnie transakcja wisiała na włosku, zanim wreszcie firma uzyskała licencję eksportową i w maju 1975 roku rozpoczęła się dostawa systemu. Szczęśliwie, perturbacje w realizacji kontraktu nie przerwały prac przygotowawczych i gdy CYBER dotarł wreszcie do Krakowa, grunt był przygotowany: klimatyzowane pomieszczenia, przeszkolona kadra (szkolenie w USA, Austrii i Świerku), rozpakowane oprogramowanie i systemy operacyjne (początkowo system SCOPE 3.4.1, a następnie od października 1975 roku przejście na SCOPE 3.4.3), przeszkoleni użytkownicy.

System rozpoczęto eksploatować próbnie w dniu 27 czerwca 1975 r. a praktycznie w listopadzie tegoż roku. Poważnie zaawansowane prace progra-

mowe i sprawnie zorganizowana teledacja (Urząd Telefonów wyselekcjonował najlepsze łącza) zagwarantowały z miejsca pełne obciążenie systemu zadaniami.

Nadmienić należy, że wstępnym warunkiem uzyskania licencji eksportowej było przeznaczenie komputera; inny odbiorca, niż środowisko naukowe — nie otrzymałby nigdy CYBERA. A i CYFRONET nie otrzymał go w maksymalnej na tamte czasy konfiguracji.

2. Rodzaje zysku czyli zazdrość bezinteresowna

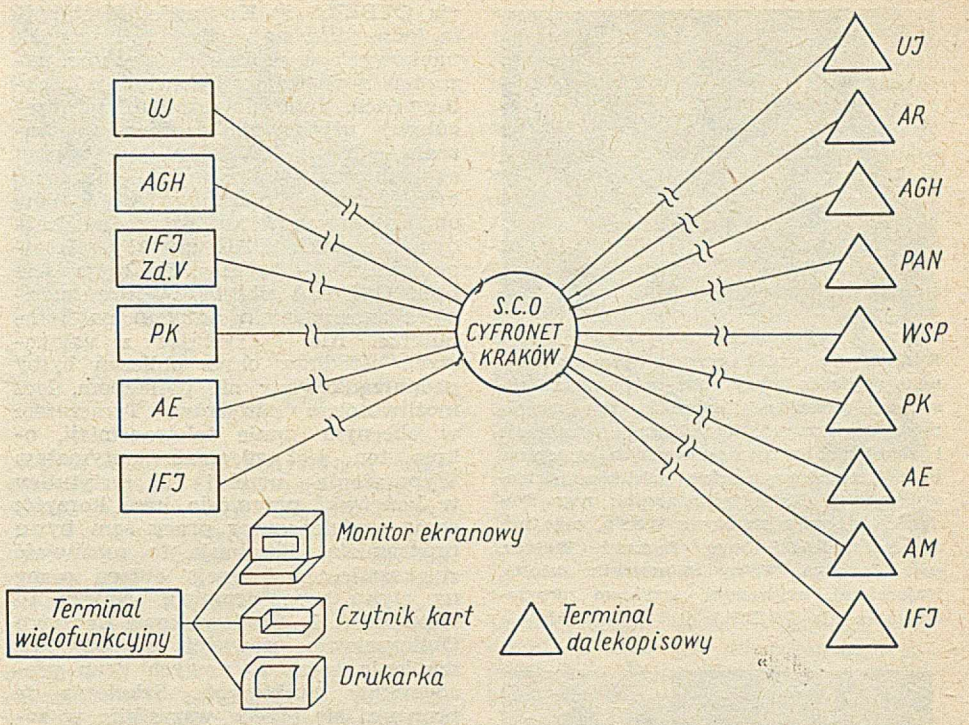
Środowiskowe Centrum Obliczeniowe CYFRONET powołane zostało do życia Zarządzeniem nr 280 Ministra Nauki, Szkolnictwa Wyższego i Techniki z dnia 23 III 1973 r. (Opublikowane w Dzienniku Urzędowym MNSzWiT nr 8 z 12 VI 73). Zarówno Zarządzenie, jak i będący suplementem Statut ŚCO określa strukturę i zakres działalności. ŚCO CYFRONET powołano jako wyodrębnioną jednostkę organizacyjną MNSzWiT, podległą bezpośrednio Ministrowi.

Zainstalowany komputer CYBER 72-16 przeznaczony jest do wykonywania obliczeń dla wszystkich wyższych uczelni, placówek Polskiej Akademii Nauk i Instytutu Fizyki Jądrowej, a więc dla instytucji finansowanych z budżetu państwa — instytucji których działalność nie jest obliczona na wypracowywanie zysku. Co nie znaczy bynajmniej, aby państwo nie czerpało zysków z pracy uczelni, PAN, czy IFJ. Podobnie rzecz się ma z CYFRONETEM; nie jest on oczywiście instytucją typu przedsiębiorstwa, trudno więc obliczyć, jakich zysków przysparza jego działalność, choć jest pewne, że zyski te nie są wcale bagatelne. Gospodarka narodowa może osiągnąć maksymalne przyspieszenie pod warunkiem ścisłych związków z myślą naukową, zaś rozwój tej ostatniej jest w znacznym stopniu uzależniony od sprawnie działających narzędzi.

Oczywiście można sobie kalkulować, ile udałoby się uzyskać złotówek przez usytuowanie CYBERA w ośrodku usługowym typu przedsiębiorstwa. Zysków, jakie przysparza CYBER nie da się wymierzyć w złotówkach, można je tylko mierzyć w korzyściach, które osiąga się z tytułu wdrażania nowych rozwiązań opracowywanych w warsztatach naukowców. Ponieważ zaś specyfika CYBERA jest powszechnie znana, nawet zazdrośnikom — zazdrość ta ma charakter czysto bezinteresowny.

3. Drogo kosztował, drogo kosztuje

Kolejny zarzut wymaga rozważenia. Prawda — CYBER kosztował niemało (tylko jednostka centralna z końcówkami — blisko 1,5 mln $), ale i samochód kosztuje więcej niż rower; różnica jest ewidentna, choć oba mechanizmy służą do jazdy. Mimo to samochodem jedzie się szybciej; CYBER — jak stwierdzili koledzy ze Świerka liczy identycznie, jak ODRA 1305 zadanie — 10 razy szybciej. To porównanie sprowadza do właściwych rozmiarów także zarzut o wysokich kosztach



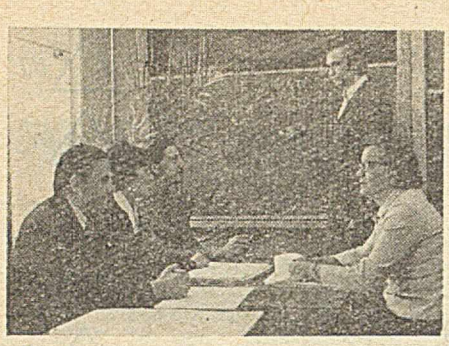
Rys. 3. Przy obecnej konfiguracji CYBERA, reprodukowany schemat systemu abonenckiego wyczerpuje możliwości komputera. Terminale duże (wielofunkcyjne) pracują w systemie *on-* lub *off-line* z możliwością przesyłu 2400 bodów, terminale dalekopisowe z szybkością 300 bodów. Koszt terminala wielofunkcyjnego — 38 000 $. Wśród użytkowników końcówek, najbardziej absorbują jednostkę centralną komputera fizycy z Uniwersytetu Jagiellońskiego i Akademii Górniczo-Hutniczej, Automatycy z Politechniki, geofizycy z AGH. Sporą grupę w ogólnej liczbie zadań stanowią prace magisterskie

Rys. 4. 30-osobowy zespół działu koordynacji i organizacji produkcji ma ręce pełne roboty. Właściwie na dobrą sprawę należałoby go dawno powiększyć jako, że tym zespołem można obsłużyć tylko 2 zmiany. Tymczasem komputer jest obciążony zadaniami przez prawie 3 zmiany. Rośnie więc liczba nadgodzin, męczą się oczy operatorek zasiadających przed migającym, niezwykle aktywnym ekranem komputera. Już w styczniu 1976 roku CYBER 72 przepracował 1,8 mln sekund systemowych (jeszcze we wrześniu 1975 r. tylko 0,96 mln). Na zdjęciu: grupa pracowników eksploatacji systemu; od lewej mgr Karol Franczak, Grażyna Widomska, Leszek Pieronek, Teresa Krawiec, Janina Sitko, mgr Marek Księżyk (kierownik działu). Barbara Zdunek, mgr Jan Chyrka, mgr inż. Zofia Mosurska

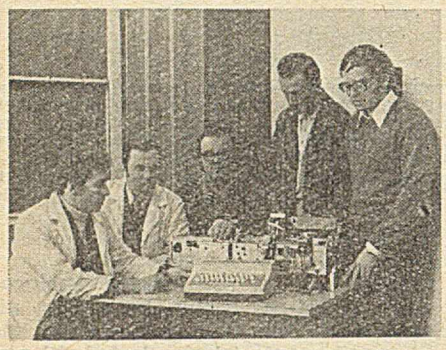
eksploatacyjnych CYBERA. To prawda, że koszt godziny pracy tego komputera waha się od 15 do 50 000 złotych (w zależności od przyznania lub nie priorytetów). Ale godzina pracy ODRY 1305 też kosztuje około 3 000 zł, a liczy ona wolniej...

Oczywiście, przy większej liczbie pracujących końcówek, korzyści tak ekonomiczne, jak i społeczne byłyby większe, ale w warunkach krakowskich problem liczby końcówek nie wynika z irracjonalnych oporów do ich przyłączenia (choć o końcówki niełatwo), co z niedostatku modemów. A w rzeczywistości, CYFRONET jest naciskany, aby objął zasięgiem swego działania uczelnie południowej Polski, z których wiele zabiega o korzystanie z CYBERA (WSP Rzeszów, Uniwersytet Śląski, Politechnika Częstochowska) oraz uruchomił łącza transmisyjne do Świerka czy Wrocławskiego WASC.

Rys. 5. Implementacją systemów operacyjnych CYBERA zajmuje się w CYRRONECIE zespół analityków. Największym sukcesem tego młodego zespołu było wprowadzenie nowego systemu operacyjnego SCOPE 3.4.3. bez przerywania pracy prowadzonej pod systemem 3.4.1. Na zdjęciu od lewej: mgr Jowita Januszewska, mgr Maria Polok (kierownik działu), mgr inż. Wojciech Zasada, mgr inż. Dorota Mynarz, mgr inż. Maria Stawiarska. Należy wspomieć, że implementacja SCOPE 3.4.3. o tyle nie należała do łatwych, że system ten był wówczas jeszcze „świeży" i mało sprawdzony nawet przez CDC



Rys. 6. Do zadań zespołu programistów należy przede wszystkim sprawdzanie pisanych przez użytkowników podprogramów użytkowych, doradztwo, konsultacje i szkolenie. Innym poważnym zadaniem było „rozpakowywanie" pakietów firmowych CDC. Na zdjęciu zespół: mgr Stefan Świąć, mgr Krystyna Wcisło, mgr inż. Janusz Galeński, mgr Ryszard Wejnar, mgr Jadwiga Wolny (kierownik działu). Najczęściej używanym językiem programowania dla CYBERA jest FORTRAN

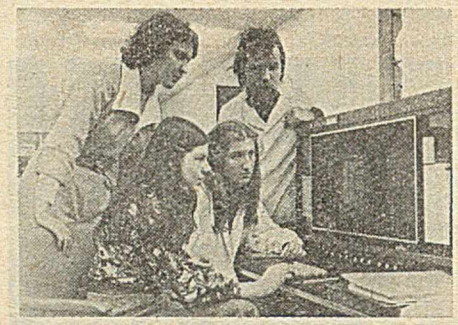


Rys. 7. Nad konserwacją CYBERA i końcówek czuwa zespół: mgr inż. Adam Świerczyna (po rocznym stażu w USA, Austrii i Holandii, gdzie między innymi w Zakładach CDC asystował przy montażu przewidzianego do CYFRONETU komputera CYBER 72-16), mgr inż. Bogusław Erbel, Stefan Gądek, inż. Zbigniew Boczar, mgr inż. Bogusław Owczarek (też po stażu w USA). Cały zespół liczy 22 osoby (starannie przeszkolone) i na brak zajęcia nie narzeka. Wprawdzie niezawodność jednostki centralnej komputera jest wysoka, ale np. jednostki pamięci taśmowej ulegają defektom stosunkowo często — podobno skażone są błędem konstrukcyjnym, do czego przyznają się producenci. Dużą pomocą w usuwaniu awarii jest rozbudowany automatyczny system kontroli technicznej lokalizujący ich przyczyny również w trakcie działania maszyny

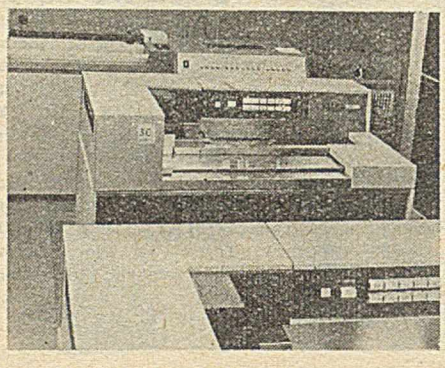
Tak jednak krawiec kraje... Naprzekór bowiem pogłoskom, że CYFRONETOWI droga różami się ściele, gdy przyszło np. do zakupu plottera (CALCOMP) stać było tylko na zakup oprogramowania podstawowego, użytkowe już trzeba było organizować własnym sumptem.
Kosztuje więc CYBER nie za tanio i nie za drogo — nie drożej, niż inne komputery, przy czym jedno jest pewne: wysoka jakość obliczeń (dobre, różnorodne oprogramowanie, niezawodność sprzętu).

4. Katalog korzyści

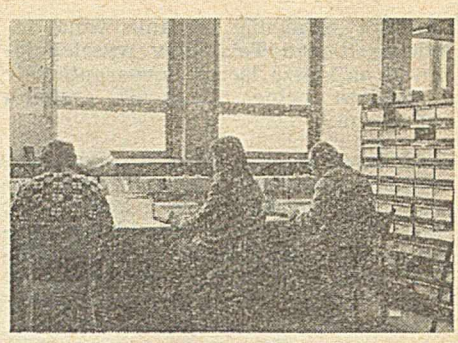
O pierwszej korzyści już nieco było: ma ona charakter ogólny — instalacja CYBERA w Krakowie udostępniła jednemu z największych środowisk naukowych w Polsce nowoczesne narzędzie wyręczające w żmudnych obliczeniach. Inne dotyczą już poszczególnych użytkowników. Idea wyposażenia ŚCO w CYBERA jest bardzo ekonomiczna — eliminuje konieczność organizowania oddzielnych ośrodków obliczeniowych w każdej uczelni, każdym instytucie PAN czy IFJ. Kolejna korzyść — to czas. Bo oczywiście brak CYBERA nie byłby jednoznaczny z zaniechaniem obliczeń na komputerze. Ale korzystając z usługowych ośrodków okres obliczeń byłby podporządkowany nie potrzebom, lecz możliwościom ośrodków. Tymczasem w obecnym stanie teletransmisji, okres ten jest nieomal optymalny. Wyposażenie uczelni i instytutów w końcówki przyniosło inną korzyść; coraz częściej autor pracy sam bywa programistą. Eliminuje to możliwość zniekształcenia intencji autora pracy na styku z programistą. Należy tu wspomnieć, że Środowiskowe Centrum Obliczeniowe prowadzi permanentne szkolenia coraz to nowych grup pracowników naukowych. Szkolenia te prowadzi się przede wszystkim w zakresie praktycznego programowania, ale także obejmują one wiadomości o systemie operacyjnym, programowania dla CALCOMPU, wiadomości z zakresu przygotowywania danych.



Rys. 8. Migawka z codziennej rzeczywistości CYFRONETU: przy konsoli operatorskiej mgr inż. Dorota Mynarz, Teresa Krawiec, Danuta Olszczyńska i Jerzy Mierniczek



Rys. 9. Szybkie czytniki kart zainstalowane w CYFRONECIE pełnią kluczową rolę w pracach realizowanych w trybie *off-line*. W głębi autotomatyczna kreślarka firmy CALCOMP, której jedyny egzemplarz z trudem zaspokaja liczne (z uwagi na charakter większości prac) potrzeby abonentów końcówek. Zwiększenie liczby urządzeń graficznych jest jedną z najpilniejszych potrzeb CYFRONETU



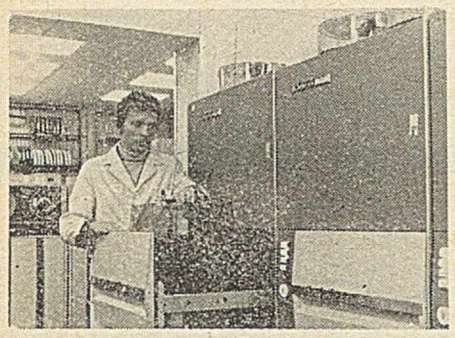
Rys. 10. Użytkownicy często korzystają z tego pomieszczenia. W sąsiedztwie hali komputera wydzielono pokój dla użytkowników, w którym można wydziurkować karty programów, uzupełnić lub reprodukować karty danych, odebrać wydruki zadań. Bezustanny ruch w tym pomieszczeniu jest jeszcze jednym dowodnym świadectwem zapotrzebowania na pracę CYBERA

Rys. 11. Z pewnością już niedługo będziemy oglądali to zdjęcie z sentymentem należnym dziś protoplaście ENIACOWI lub sędziwej ODRZE 1003. Zanim to jednak nastąpi, nie tylko w polskiej geografii komputerów, produkt CONTROL DATA CORPORATION — CYBER 72 jest symbolem nowoczesności. Liczba operacji na sekundę zbliża się do miliona. Na tle jednostki centralnej komputera prezentujemy przedstawicielkę młodej załogi CYFRONETU — operatorkę Barbarę Zdunek

Szkolenia prowadzą pracownicy ŚCO. Jednym z efektów jest ograniczenie zatrudnienia; im więcej ludzi potrafi samodzielnie programować, tym mniej zawodowych programistów muszą zatrudniać uczelnie i instytuty.

Obecny stan przygotowania użytkowników do samodzielnego korzystania z usług komputera nie może być jeszcze najwyższy. Wymaga to czasu. W tej sytuacji ekipa programistów musi jeszcze często udzielać konsultacji, a zespół operatorek przygotowywać dane do obliczeń. Kadra CYFRONETU jest stosunkowo nieliczna: aktualnie 106 osób na pełnym i 13 na pół-etatu, w tym 6 analityków, 20 programistów, 30 pracowników eksploatacji. Ale jest to kadra młoda i pełna zapału, a co najważniejsze dobrze wyszkolona. Dużo dobrego usłyszałem o

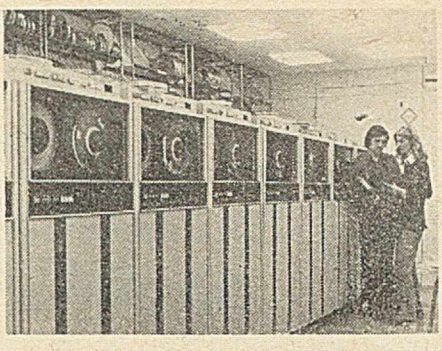
absolwentach pomaturalnej szkoły informatyki. Mimo dobrego poziomu wyjściowego, kadra ta przechodzi systematyczne szkolenia.



Rys. 12. Jednostki pamięci dyskowej CYBERA są już zbyt mało pojemne, jak na potrzeby ŚCO. Kiedy Amerykanie uchylą embargo na duże dyski?

5. Zamiast podsumowania

Przekonaniu, że CYBER jest niezbędny i dobrze wykorzystany w Środowiskowym Centrum Obliczeniowym CYFRONET w Krakowie starałem się dać wyraz w tym, co wyżej napisałem. Gdyby jednak argumentacja



Rys. 13. Jednostki pamięci taśmowej stanowią względnie najsłabsze ogniwo w konstrukcji komputera CYBER

ta była niezbyt przekonywująca mogę dorzucić tylko to, co zobaczyłem — odchodząc — w hali komputera. Oto, gdy rzuciłem okiem na ekran CYBERA, lista zadań kończyła się na pozycji 323; była za parę minut godzina 15^{00}, z nastaniem której system zawsze drukuje informację: INTERCOM, DROP (Koniec pracy konwersacyjnej).
Następowała dwugodzinna przerwa, w czasie której procesor „nadrabiał zaległości", wykonując wprowadzone do momentu przerwy programy.

Rys. 14. Minister Nauki, Szkolnictwa Wyższego i Techniki prof. dr hab. Sylwester Kaliski wizytował z okazji otwarcia ŚCO CYFRONET. Na zdjęciu Minister — w asyście wiceministra prof. dr hab. Waleriana Kujawskiego (trzeci z lewej) i I Sekretarza KW PZPR w Krakowie Wita Drapicha (drugi z lewej) — słucha objaśnień dyrektora ŚCO doc. dr hab. Jerzego Kolendowskiego

Liczba zadań, które codziennie wykonuje CYBER jest dowodem użyteczności tego komputera w środowisku krakowskim.

Krystyn Bernatowicz
Foto W. Wójczyk (KAW)

ZE ŚWIATA

Konferencja CONVENTION INFORMATIQUE

W Paryżu w dniach od 20 do 24 września br. odbędzie się po raz drugi w powiązaniu z międzynarodową wystawą sprzętu komputerowego SICOB (Salon international de l'iformatique, de la communication et de l'organisation du bureau) międzynarodowa konferencja informatycza CONVENTION INFORMATIQUE.

Organizatorzy tej imprezy, którymi są francuskie organizacje SYNTEC INFORMATIQUE (zrzeszenie instytucji badawczych i konsultacyjnych informatyki) S.N.I.M.A.B.I. (krajowe zrzeszenie importerów materiałów biurowych i sprzętu informatycznego) oraz S.F.I.B. (krajowe zrzeszenie producentów sprzętu informatycznego i maszyn biurowych), powołują się na sukces konferencji ubiegłorocznej, uznanej za największą europejską imprezę informatyczną roku 1975. Zgromadziła ona ok. 2000 uczestników, wygłoszono 116 referatów oraz przeprowadzono 16 dyskusji panelowych. Imponujące rozmiary w r. 1975 miała również towarzysząca konferencji wystawa SICOB 75, w której uczestniczyło łącznie 1652 firmy z 29 krajów, zajmując powierzchnię ok. 83 000 m². Wystawę tę zwiedziła rekordowa licza 280 065 osób. Tegoroczny SICOB 76 odbędzie się w terminie od 23 września do 1 października br.
Głównym hasłem tegorocznej konferencji, która stanowi niewątpliwie inspirację tematyczną treści wystawy, jest „Informatyka: rzeczywistość i perspektywy" („Informatique: realities et perspectives").

Podstawowym akcentem konferencji jest konfrontacja potrzeb użytkowników z aktualnym i spodziewanym rozwojem postępu technicznego w dziedzinie informatyki.

Ogólny schemat obrad konferencji przewiduje podział na 4 grupy tematyczne, z których 2 (A i B) dotyczyć będą stanu aktualnego, a 2 (C i D) perspektyw rozwoju informatyki.

W szczegółach tematyka ta, odzwierciedlająca aktualne światowe problemy i trendy rozwojowe informatyki, przedstawia się następująco:

— grupa tematyczna A: rozwiązania techniczne

- sieci informatyczne (krajowe i międzynarodowe)
- przetwarzanie rozproszone (architektura sprzętu i oprogramowanie)
- bazy danych (wybór rozwiązań, zastosowanie minikomputerów)

— grupa tematyczna B: wdrożenia

- informatyka w administracji publicznej
- informatyka i telekomunikacja
- minikomputery dla zarządzania (problemy wdrożenia, efektywność stosowania)
- automatyzacja przetwarzania tekstów i dokumentacji (automatyczny skład w poligrafii, mikrofilmowanie)
- pakiety zastosowań (kryteria wyboru, standaryzacja, okres żywotności)

— grupa tematyczna C: ewolucja rozwiązań technicznych

- sprzęt (architektura, technologia)
- systemy operacyjne rozproszone
- bazy danych
- telekomunikacja

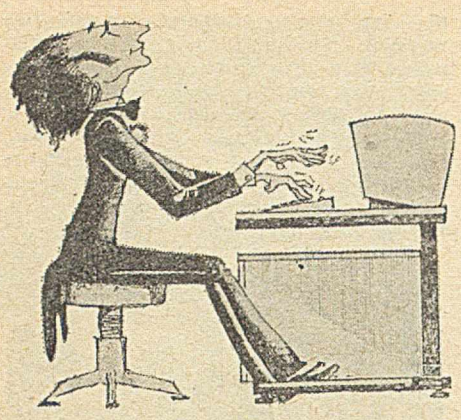
— grupa tematyczna D: ewolucja potrzeb

- decentralizacja (centralizacja w przedsiębiorstwach i administracji publicznej — problemy wyboru)
- stosunki pomiędzy administracją a administrowanymi
- automatyzacja (nowe dziedziny i ograniczenia)
- niezawodność i zabezpieczenie (systemów i danych)
- zarządzanie informatyką (projektowaniem, eksploatacją, kadrami).

Zgodnie z aktualną światową strukturą zastosowań organizatorzy przewidują możliwość dodatkowego branżowego wyodrębnienia tematyki interesującej podstawowe grupy użytkowników informatyki (bankowość, ubezpieczenia, handel, przemysł), a także problematykę wdrażania informatyki w krajach rozwijających się.

O dalsze informacje można zwracać się do organizatorów pod adresem: CONVENTION INFORMATIQUE 6, place de Valois, 75001 PARIS (France).

(WK)



Operacja sumowania

Kiedy ponad rok temu zdecydowaliśmy się na publikowanie materiałów dotyczących artystycznych zastosowań komputerów i wydzieliliśmy im specjalną rubrykę SZTUKA KOMPUTEROWA — przypuszczaliśmy, że będzie to chwilowa atrakcja, bardziej marginalna ciekawostka, niż poważny naukowy temat.

Tymczasem dział SZTUKA KOMPUTEROWA gościł na naszych łamach w trzynastu numerach! Okazało się, że problemy komputeryzacji sztuki są nader ciekawe i złożone, a podjęte w tym kierunku prace dały już konkretne efekty.

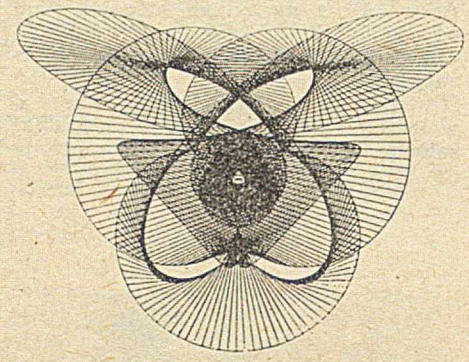
Pisaliśmy więc o zagadnieniach związanych z udziałem maszyn w muzyce, literaturze, plastyce, teatrze, filmie, balecie oraz o ich możliwości w łączeniu różnych dyscyplin artystycznych. Prezentowaliśmy najbardziej interesujące zagraniczne i krajowe rezultaty twórczości komputerowej (w tym również dzieła nadesłane przez naszych Czytelników). Przedstawialiśmy opinie ludzi odbierających sztukę komputerową i tych, którzy ją tworzą (np. napisany specjalnie dla INFORMATYKI artykuł *) inż. Oskara Beckmana, reprezentującego wiedeńską grupę *„ars intermedia"*).

*

Z pewnością nie powiedzieliśmy wszystkiego i nie rozwialiśmy wszystkich wątpliwości. To prawda, że autentyzm masowej twórczości komputerowej stoi pod znakiem zapytania. Ale przecież problem ten pojawił się wcześniej — przy okazji „puszkowanej" muzyki z płyt i taśmy, powielanych tradycyjnymi środkami w tysiącach egzemplarzy, dziełach plastycznych, gdzie za oryginał przyjęto uznawać tylko pierwszą odbitkę.

Kto wie zresztą, czy to właśnie nie maszyna pozwala idealnie zachować oryginalność oryginału, przechowując jego dokładny opis w swej pamięci. Liczby się nie zużywają i można je przetrzymywać dowolnie długo. A np. obraz po kilkudziesięciu latach starzeje się, odbiegając coraz bardziej od swojej początkowej postaci.

W przypadku komputerowych serii nie ma mowy o powtarzalności. Każda z dobieranych kombinacji jest inna, chociaż ogólny charakter dzieł pozostaje taki sam. Przy dzisiejszej unifikacji otaczających nas przedmiotów z wielkich serii, komputer daje możliwość różnicowania wytworów oferowanych indywidualnemu odbiorcy. Zawszeć to przyjemniej mieć na ścianie kompozycję unikalną, niż tanią reprodukcję znanego obrazu.



„Lew" — komputerowa grafika wytworzona przez Jana Chlouba z Czechosłowacji na ploterze CALCOMP przy wykorzystaniu maszyny ELLIOT 503 (program w ALGOLU)

Duże możliwości stwarzają również maszyny w projektowaniu sztuki użytkowej. Dysponując formacjami na temat pożądanych walorów estetycznych i na temat koniecznych parametrów eksploatacyjnych, jednoczą w sobie punkt widzenia artysty i producenta. Zdolne są do tworzenia wzorów ubrań, mebli lub karoserii samochodowych optymalnych ze względu na sprawność, koszt, wygodę i wygląd zewnętrzny albo proponowania wielu tanich wariantów, których wytworzeniem w fabryce mogą zarazem sterować bezpośrednio.

Za największą zaletę sztuki komputerowej Marc Adrian uważa to, że „sprowadza ona dzieło sztuki do jedynie możliwych dziś proporcji: przedmiotu użytkowego, rozpowszechnionego na wielką skalę, niemal jak chusteczka do nosa czy gazeta, które po użyciu można podarować albo zniszczyć". Daje to także okazję do usunięcia odwiecznych barier między sztuką a rzemiosłem.

Komputer nie powinien również stać się, jak sądzą niektórzy, grobem twórczości amatorskiej. Wręcz przeciwnie. Artysta będzie mógł tworzyć swobodnie, nie skrępowany potrzebami odbiorcy, problemami wykonawczymi lub trudnościami finansowymi.

Technika zwiększa jego możliwości, dając mu okazję wyboru. Tak, jak postęp w komunikacji: kto chce, może latać samolotem, jeździć samochodem lub koleją. Ale nie znaczy to bynajmniej, aby nam zabroniono używać roweru lub własnych nóg. Całkiem przeciwnie. Wynalazek samochodu spowodował, że piesze marsze zaczęto traktować jako popularny sport.

Zastosowanie maszyn cyfrowych w sztuce prowadzi więc raczej do nowego sposobu jej rozumienia: problem jest otwarty, nie ma jednoznacznych odpowiedzi, są tylko propozycje. Udział przypadku i reguł, dozowanych w różnym stopniu przez komputer uczy przyjmowania sztuki w sposób spontaniczny i różnorodny, szybko dostarczając dużych ilości materiału eksperymentalnego, daje wiele wariantów. Pozwala docenić piękno istniejące poza słowami, dźwiękami i obrazami, zawarte w samej konstrukcji intelektualnej. „Jakaż głęboka ironia tkwi w stwierdzeniu — pisze D. M. Davis — że maszyna daje poczucie radości od dawna nieobecne w sztuce współczesnej".

A oto, co powiedział na ten temat jeden z najbardziej cenionych w Europie autorytetów, prof. Wiktor Głuszkow, dyrektor Instytutu Cybernetyki ZSRR: „Maszyny mogą wnieść dużo nowego w dziedzinę sztuki... Maszyna może doskonale sprawdzić przydatność idei twórczej — czy było już kiedyś coś podobnego, czy pomysł człowieka jest prawidłowy w swoim założeniu? Przy sprawdzaniu, maszyna bierze pod uwagę taką liczbę faktów, której nie jest w stanie objąć nie tylko jeden człowiek, ale i cały kolektyw. W ten sposób maszyna stwarza człowiekowi komfortowe warunki twórczości. Powstaje swojego rodzaju symbioza: i człowiek i maszyna dają z siebie wszystko, co mają najlepszego... W tej sytuacji niedadaleko już do tego, aby uznać maszynę duchowym spadkobiercą człowieka".

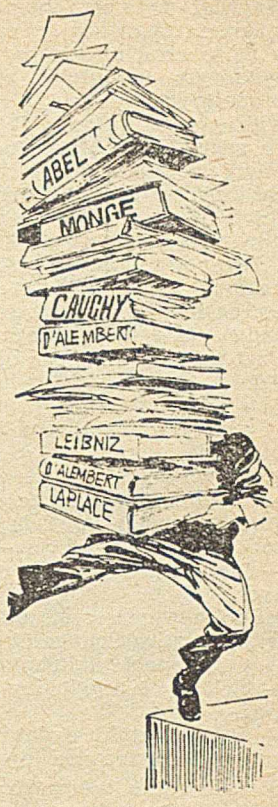
*

Kończąc comiesięczny cykl SZTUKI KOMPUTEROWEJ nie mamy rzecz jasna, zamiaru rozstawać się z tą tematyką. Być może doczeka się ona nawet szerszego omówienia. Wydawnictwo „Wiedza Powszechna" zainteresowało się problematyką poruszoną na naszych łamach i kto wie, czy wrótce nie pojawi się na rynku książka o komputerach w sztuce.

Nadal też prosimy o nadsyłanie wszystkich materiałów związanych z artystyczną działalnością maszyn cyfrowych. Najciekawsze z nich mamy zamiar od czasu do czasu publikować.

Marek Hołyński

*) Aspekty sztuki komputerowej — Oskar Beckman, tłum. i oprac. Władysław Klepacz INFORMATYKA 1976, nr 3, str. 35.



Recenzja książki Teuvo Kohonena: Elementy elektronicznych maszyn cyfrowych*)

Niezwykle szybki rozwój techniki cyfrowej w ostatnich latach sprawia, że literatura z tego zakresu jest pilnie poszukiwana przede wszystkim przez rzesze inżynierów i techników interesujących się zawodowo tymi zagadnieniami.

Kolejną pozycją z tego zakresu jest wydane przez Wydawnictwa Naukowo-Techniczne w Warszawie tłumaczenie z języka angielskiego książki fińskiego autora Teuvo Kohonena pt. „Elementy i układy elektronicznych maszyn cyfrowych".

Wachlarz problemów poruszanych w książce jest bardzo szeroki, jednak sposób przedstawienia tej problematyki przez Autora sprawił, że zawarty w niej materiał stanowi zamkniętą całość. Materiał rzeczowy został przedstawiony w sposób jednolity, co pozwoliło dać Czytelnikowi ogólny pogląd na opisywaną tematykę (dziedzinę).

Książka składa się z dwóch części zatytułowanych kolejno:
— Projektowanie logiczne (rozdziały 1 — 5)
— Elektroniczne układy cyfrowe (rozdziały 6—10).

Pierwsza część książki nie uwzględnia konkretnej realizacji układów pozostawiając wiele możliwości. Autor rozróżnia „układy logiczne" i „układy cyfrowe" ze względu na funkcje, które układy te spełniają. Zgodnie z tym kryterium układy cyfrowe w zasadzie przeznaczone są do wykonywania operacji na liczbach (zmiennych matematycznych) natomiast układy logiczne stosowane są do symulacji decyzji logicznych np. do sterowania współzależnych operacji opisanych za pomocą rachunku zdań. W rozdziale pierwszym przedstawiono zagadnienie kodowania liczb oraz zdefiniowano język symboliczny operacji przetwarzania cyfrowego. Tematyka rozdziału drugiego i trzeciego obejmuje problematykę projektowania kombinacyjnych i sekwencyjnych układów logicznych. Jako metody minimalizacji przedstawiono metodę tablic Karnaugh'a oraz metodę Quine'a-Mc Cluskey'a. Należy jednak sprostować, że minimalizacji może podlegać postać wyrażenia logicznego, natomiast posługiwanie się terminem „minimalizacja funkcji" uznać należy za niepoprawne. Funkcji rozumianej jako pewne przyporządkowanie elementom jednego zbioru elementów zbioru drugiego nie można minimalizować. Synteza dekodera wskaźnika siedmiosegmentowego przedstawiona w podrozdziale 2.8.3. jest niepoprawna. Wyrażenia na a,e,f,g są błędne. Problematykę projektowania układów sekwencyjnych zawężono do omówienia:
— syntezy układów asynchronicznych metodą Huffman-Mealy'ego, a nie metodą Huffmana-Moore'a jak to wynika z kontekstu zdania: „Istnieje inna metoda zwana..." (str. 95_{4-5})
— syntezy układów synchronicznych metodą tablic wzbudzeń (tłumaczonych jako „warunki sterowania").

Zamieszczenie podrozdziału 3.5.1 w ramach rozdziału 3.5 jest nieuzasadnione ze względów merytorycznych. Omawiany bowiem układ jest asynchronicznym przerzutnikiem typu T. Zastrzeżenie budzi sposób zapisu wyrażeń logicznych odpowiadających realizacjom układów sekwencyjnych asynchronicznych.

Rozdziały czwarty i piąty uznać można za najważniejsze, co zresztą Autor sugeruje we wstępie. Niestety nie znajduje to odbicia w objętości i zakresie tematycznym. W rozdziale czwartym omówiono syntezę liczników szeregowych i równoległych, elementy arytmetyki maszyn cyfrowych, realizację operacji arytmetycznych. Podrozdział 4.3. w całości powinien raczej stanowić uzupełnienie podrozdziału 3.6., gdzie Autor omawia zagadnienie syntezy układów synchronicznych. Podrozdziały 4.5. oraz 4.6 poświęcone analizie algorytmów arytmetycznych za pomocą operacji rejestrowanych i przetwarzaniu przyrostowemu należy uznać za zbyt skrótowe. W rozdziale piątym omówiono zagadnienie sterowania sekwencyjnego oraz mikroprogramowego. Dokonano interesującego porównania tych dwu rodzajów sterowania na przykładzie arytmometru, co już przy tak prostym układzie pozwoliło uwypuklić zalety mikroprogramowania. Problematykę programowania omówiono na przykładzie „zasady pamiętanego sterowania programowego von Neumana", która to zasada jest realizowana na poziomie makro w większości współczesnych maszyn cyfrowych.

Rozdział szósty zapoczątkowuje drugą część książki i poświęcony jest teorii nieliniowych przełączników prądowych i napięciowych. Najistotniejszą rzeczą, z punktu widzenia projektanta, jest zdefiniowanie dla łańcucha inwertorów statycznego i dynamicznego marginesu zakłóceń oraz odporności na zakłócenia, a następnie rozszerzenie tych pojęć dla przypadku bramek wielowejściowych.

W rozdziale siódmym Autor omawia różne techniki realizacji elementów logicznych z tranzystorami bipolarnymi (techniki RTL, DTL, HTL, TTL, ECL) oraz z tranzystorami unipolarnymi (MOS FET, COS-MOS).

Rozdział kończy porównanie podstawowych parametrów logicznych omawianych elementów oraz podaje zalecenia odnośnie do wyboru elementów scalonych zrealizowanych w różnych technikach.

Rozdział ósmy poświęcony jest omówieniu podstawowych elementów syntezy i analizy typowych układów synchronizacji (układ rozszerzania impulsów, formowania impulsów, multiwibrator monostabilny i astabilny) zrealizowanych w oparciu o bramki logiczne oraz wzmacniacz operacyjny. Szkoda tylko, że Autor zrezygnował z omówienia zagadnienia projektowania układów logicznych z uwarunkowaniami czasowymi.

Rozdział dziewiąty poświęcony jest skrótowemu przeglądowi pamięci masowych maszyn cyfrowych. Omówiono zasadę działania i organizacji pamięci półprzewodnikowych oraz pamięci ferrytowych. Opisano ponadto pamięci stałe i dokonano przeglądu powszechnie stosowanych kinetycznych pamięci magnetycznych.

*) WNT, Warszawa 1975, str. 397, ark. wyd. 30,0.

Rozdział dziesiąty obejmuje zagadnienia związane z przesyłaniem sygnałów cyfrowych przy założeniu modelu linii o parametrach rozłożonych. Problem przesyłania sygnałów cyfrowych w obrębie projektowanego układu lub współpracujących z nim układów jest bardzo istotny z punktu widzenia projektanta. Z bardziej istotnych zagadnień, które sygnalizuje Autor można wymienić: przesyłanie informacji w liniach krótkich, średnich i długich, powstawanie zakłóceń w liniach, przesyłanie informacji na duże odległości. Wartość książki podnosi: stosunkowo obszerna bibliografia (131 pozycji książkowych i ponad 200 artykułów i opracowań firmowych), zamieszczenie przez Autora po każdym rozdziale komentowanego odsyłacza do literatury szczegółowej, zamieszczenie po każdym rozdziale tematów zadań do samodzielnego rozwiązania, co pozwala ocenić Czytelnikowi stopień opanowania tematyki danego rozdziału. Strona redakcyjna książki budzi pewne zastrzeżenia:

— niezbyt ujednolicona i poprawna terminologia (np. przeplata się „komputer" z „maszyną cyfrową", używa się pojęć „zapalanie" i „gaszenie" w znaczeniu „wpisywanie", „zerowanie", tak samo zresztą „wejścia zapalające i gaszące" zamiast „wejścia wpisujące i zerujące", przełącznica — wybierak, tabela — tablica, itd.)

— brak ujednolicenia oznaczeń symboli operacji logicznych.

Mimo pewnych zastrzeżeń zasygnalizowanych w przedstawionych uwagach należy stwierdzić, że książkę cechuje:

— przystępność i komunikatywność

— ciekawy i oryginalny sposób ujęcia szerokiej problematyki

— duża wartość dydaktyczna, jako „przewodnika po zagadnieniach techniki cyfrowej".

W celu nieco szerszego naświetlenia pewnych wybranych zagadnień z zakresu tematycznego prezentowanej książki należałoby Czytelnika polskiego zachęcić do zapoznania się z pracami zamieszczonymi w poniższej literaturze.

LITERATURA

[1] Chu Y.: Maszyny cyfrowe, Warszawa 1967, PWN

[2] Anderson W. D. i inni: Projektowanie układów z TTL obwodami scalonymi, Warszawa 1973, Ośrodek Informacji o Energii Jądrowej

[3] Eimbinder J. i inni: Pamięci półprzewodnikowe scalone, Warszawa 1975, WNT

[4] Flores I.: Arytmetyka maszyn cyfrowych, Warszawa 1970, WNT

[5] Kulka Zb., Nadachowski M.: Liniowe układy scalone i ich zastosowanie, Warszawa 1973, Ośrodek Informacji o Energii Jądrowej

[6] Łakomy M., Zabrodzki J.: Cyfrowe układy scalone TTL, Warszawa 1975, PWN

[7] Martin J.: Wprowadzenie do transmisji danych, Warszawa 1975, WNT

[8] Misiurewicz P., Grzybek M.: Półprzewodnikowe układy logiczne, Warszawa 1975, WNT

[9] Morris Mano M.: Projektowanie systemów logicznych maszyn cyfrowych Warszawa 1975, WNT

[10] Nowak E., Sawicki Z.: Pamięci maszyn cyfrowych, Warszawa 1972, WNT

[11] Pieńkos J., Turczyński J.: Przetwarzanie i przechowywanie informacji w systemach automatyki, Warszawa 1974, WNT

[12] Siwiński J.: Układy przełączające w automatyce, Warszawa 1968, WNT

[13] Traczyk W.: Układy cyfrowe automatyki, Warszawa 1974, WNT

[14] Wagner F.: Liczniki elektroniczne w przemysłowych układach sterowania, Warszawa 1971, WNT

Bolesław Pochopień
Instytut Automatyki Przemysłowej
i Pomiarów Pol. Śl. w Gliwicach
Zespół Teorii Automatów Cyfrowych

Bibliografia wydawnictw polskich z dziedziny informatyki

● **Mikroelektroniczne elementy i układy cyfrowe — LOGISTER** — CYRAN A., GRZYBEK M., MARCZEWSKI M., PIEŃKOS J. — WKiT, Warszawa 1975, ss. 462, cena zł 80,—

Cz. 1. Wiadomości podstawowe: operacje logiczne i arytmetyczne, podzespoły elektroniczne i mikroelektroniczne, mikroelektroniczne układy cyfrowe. Cz. 2. Elementy i układy systemu LOGISTER: elementy podstawowe, logiczne zespoły funkcjonalne, zasilacze. Cz. 3. Urządzenia, projektowanie, konstrukcja, przykłady zastosowań: projektowanie urządzeń cyfrowych, montaż i konstrukcja urządzeń, przykłady wybranych układów i urządzeń cyfrowych.

Materiały przeznaczone są dla inżynierów automatyków i elektroników projektantów i użytkowników przemysłowych układów sterowania.

● **Zebranie i wprowadzanie informacji do komputera.** Tłum. wyd. ang. Wyd. Ośrodka Badawczo-Rozwojowego INFORNA, Warszawa 1975, ss. 45. Europejski Program Badawczy DIEBOLDA. Zeszyt 71 (E 122 M). Cena zł 92.—

Streszczenie dla kierownictwa. Zbieranie i wprowadzanie informacji do komputera. Sprzęt do wprowadzania danych na nośnikach papierowych. Terminale wideograficzne do wprowadzania danych. Wprowadzanie danych głosem. Klawiaturowe urządzenia do wprowadzania danych.

Materiały przeznaczone są dla projektantów i użytkowników urządzeń wejścia i wyjścia.

● **Kryteria oceny wydajności operacji APD.** Tłum. wyd. ang. z roku 1974. Wyd. Ośrodka Badawczo-Rozwojowego Informatyki INFORNA, Warszawa 1976, ss. 99. Europejski Program Badawczy DIEBOLDA. Zeszyt 73 (E 120 P). Cena zł 92.—

Wprowadzenie: tło badań, streszczenie ustaleń i wniosków, zakres i metodyka opracowania. Ustalenie zasad oceny wydajności: wprowadzenie, zasadnicze zadanie operacji APD, wyznaczanie probierzy wydajności, rola norm. Kryteria wydajności i techniki jej pomiaru: wprowadzenie, wymiary wydajności, pomiar wydajności i jego stosowanie, sprawozdawczość i ocena wydajności.

Materiały przeznaczone są dla kadry kierowniczej przedsiębiorstw, kierowników ośrodków obliczeniowych oraz ekonomistów.

● **Przykłady finansowania informatycznych systemów zarządzania** — 1971. Tłum. wyd. ang. z roku 1971. Wyd. Ośrodka Badawczo-Rozwojowego Informatyki INFORNA Warszawa 1975, ss. 200. Europejski Program Badawczy DIEBOLDA. Zeszyt 64 (E 91), cena zł 92,—

Wprowadzenie oraz wnioski ogólne. Przegląd przykładów finansowania EPD. Porównanie sposobów finansowania EPD w piętnastu dziedzinach gospodarki. Profil nakładów ponoszonych na EPD: piętnaście dziedzin gospodarki. Finansowanie EPD w przedsiębiorstwach handlowych, usługowych, transportowych, w bankach handlowych w firmach ubezpieczeniowych.

Materiały przeznaczone są dla kadry kierowniczej przedsiębiorstw, kierowników ośrodków obliczeniowych oraz ekonomistów.

● **Elektroniczna maszyna cyfrowa Jednolitego Systemu R 20** — BUKOWY A., JĘDRZEJCZYK R., PREGIEL R. Wyd. Ośrodka Badawczo-Rozwojowego Informatyki INFORNA, Warszawa 1975, ss. 113, cena zł 58.—

Problemy informatyki.

Jednolity System Elektronicznych Maszyn Cyfrowych. Charakterystyka techniczno-eksploatacyjna systemu R 20. Oprogramowanie systemu R 20. Zasady generowania dyskowego systemu operacyjnego DOS/JS. Translatory systemu DOS/JS. Zastosowania. Zagadnienia wdrożeniowe. Załączniki.

Materiały przeznaczone są dla użytkowników komputera R 20.

● **Rachunek ekonomiczny jako metoda badawcza systemów informatycznych** — KISIELNICKI J., KOTULECKI W. Wyd. Ośrodka Badawczo-Rozwojowego Informatyki INFORNA, Warszawa 1975, ss. 97, cena zł 58.— Problemy informatyki.

Wstęp. Rachunek ekonomiczny jako narzędzie oceny efektywności systemów informatycznych. Analiza metod oceny efektywności systemów. Badania opłacalności obiektowych systemów informatycznych za pomocą analizy rachunkiem sald. Ocena efektywności ekonomicznej wdrożenia systemu informatycznego na przykładzie przedsiębiorstwa przemysłowego.

Materiały przeznaczone są dla kadry kierowniczej przedsiębiorstw projektujących i wdrażających systemy informatyczne.

Oprac. A.K.

Uwagi i propozycje do terminologii struktur danych

Nawiązując do artykułu p.t. „Terminologia" D. Kołackiej i Innych, opublikowanego w miesięczniku INFORMATYKA nr 1/1976 r., str. 39, wnoszę uwagi i nowe propozycje do niektórych pojęć (nazw) i ich definicji (określeń), a mianowicie:

— BANK DANYCH
— BAZA DANYCH
— DANA
— DANA PROSTA
— DOKUMENT
— GRUPA, POZYCJA ZŁOŻONA
— PLIK
— POZYCJA, POZYCJA ELEMENTARNA
— STRUKTURA DANYCH.

Uwagi:

1. Niektóre różne nazwy są jednakowo definiowane (nie przestrzegano prawa wyłącznej sprzeczności nieprawda, że A i nie A) np.
— DOKUMENT — dana, której wartością jest zbiór pozycji i/lub grup
— GRUPA, POZYCJA ZŁOŻONA — dana, której wartością jest zbiór pozycji i /lub grup.
2. Definicje niektórych pojęć stanowią tautologię z definiowanym pojęciem
— **daną** określa się przez **daną** np.:
— DANA — para (nazwa i wartość), gdzie wartością może być stała, **dana** lub struktura.
3. Niektóre pojęcia wynikają z dosłownego tłumaczenia wyrazów angielskich, co nie zawsze jest zgodne z naszymi pojęciami np.:
— value (wartość) — lepiej w zakresie pojęć informatycznych przetłumaczyć jako **treść**, a nie wartość, ponieważ wartość w naszym znaczeniu odnosi się wyłącznie do spraw finansowych — wyceny pieniężnej i w przetwarzaniu danych słuszne jest tylko w tych sprawach.
4. Należało by unikać niektórych pojęć, które nie są adekwatne w dziedzinie przetwarzania danych, takich jak: DOKUMENT, PLIK, np. PLIK może oznaczać większą ilość dokumentów źródłowych lub perfokart, ale nie jest zręcznym pojęciem do określenia treści tych dokumentów zapisanych np. na taśmie magnetycznej.
5. Pojęcia w artykule są przedstawione chaotycznie tzn. nie usystematyzowano ich wg jakiegoś klucza logicznego co ułatwiało by ich rozumienie, przyswojenie.
6. Brak w artykule rozgraniczenia danej w liczbie pojedynczej, od danych rozumianych w liczbie mnogiej np.

— DANA PROSTA — Dana, której .....
— DOKUMENT — Dana, której .....

7. Wielkie uznanie należy się Autorom omawianego artykułu za próbę podjęcia nabrzmiałego problemu i jego opublikowanie celem wywołania „burzy mózgów", co jest moim zdaniem jedyną metodą umożliwiającą rozwiązanie palącego problemu struktur danych w skali ogólnokrajowej.

Propozycje

Celem usystematyzowania pojęć (nazw) wymienionych w punkcie UWAGI proponuję klucz logiczny — idąc od pojęcia jednostki informacyjnej do zbiorowych pojęć coraz bardziej zagregowanych, z wyraźnym rozgraniczeniem danej rozumianej w liczbie pojedynczej (użytej tylko do określenia pierwszego pojęcia) od danych rozumianych w liczbie mnogiej — używanych we wszystkich dalszych pojęciach będących agregacją pojęcia pierwotnego. Myślę, że pomocne w tym przypadku będzie wyraźne rozgraniczenie angielskiego wyrazu „datum", jako pojęcia jednostkowego, elementarnego, rozumianego w liczbie pojedynczej, od wyrazu „data" odnoszącego się wyraźnie do liczby mnogiej, ponieważ wyrazy te są bezpośrednio wzięte z języka łacińskiego i pochodzą od czasownika:
do — daję (pierwsza osoba liczby pojedynczej, czas teraźniejszy)
dare — dawać (bezokolicznik)
dedi — dałem (pierwsza osoba liczby pojedynczej, czas przeszły dokonany)
datum — dane np. jabłko (imiesłów czasu przeszłego, trzecia osoba **liczby pojedynczej**, rodzaj nijaki, strona bierna).
natomiast:
data — dane np. jabłka — oznacza **liczbę mnogą** od imiesłowu „datum".
W związku z powyższym proponuję następujące uszeregowanie pojęć, wymienionych w punkcie UWAGI i zastąpienie ich następującymi pojęciami:

DANA
(dana prosta)
(pole elementarne)
(pozycja prosta)
(datum)

Informacja do przetwarzania, składająca się z dwóch parametrów: treści i nazwy, gdzie nazwa jest symbolem treści, np. treść — „5555" a symbol tej treści „STANOWISKO KOSZTÓW".
Treść może być stała lub zmienna np.
— treść stała — „1321" — nazwa „NUMER PRACOWNIKA"
— treść zmienna „ " — nazwa „ZAROBEK NETTO", tzn. zarezerwowane pole elementarne, w którym zostanie zapisany zarobek po obliczeniu

POZYCJA ZŁOŻONA
(grupa pozycji)
(group, data agregatte)

— Dana złożona (dane), której treścią jest zbiór danych prostych, np.:
— treść „29.11.1926" — nazwa DATA URODZENIA", składająca się z trzech danych prostych t.j.:
— dzień urodzenia
— miesiąc urodzenia
— rok urodzenia

ZAPIS
(rekord)
(dokument)
(logical record, eutry)

— Logiczne zestawienie danych prostych i/lub złożonych odzwierciedlający treść pewnego zdarzenia, np. pracownik X przepracował Y godzin w dniu Z.

STRUKTURA ZAPISU
(Struktura rekordu)
(Struktura danych)
(data strukture)

— Zestaw danych prostych i/lub złożonych w zapisie, np. wielkość pól danych, kolejność ich występowania, ich nazwy.

ZBIÓR
(magnetoteka)
(kartoteka)
(file)

— Zestaw zapisów (rekordów) jednorodnych tematycznie, np. zarobki netto pracowników za miesiąc X przedsiębiorstwa Y.

BANK DANYCH
(data bank)

— Zestaw zbiorów utrzymywanych zależnie od systemów informatycznych te zbiory wykorzystujących, np. Zbiory zapisów o pracownikach, środkach trwałych, materiałach są tworzone przez konkretne systemy informatyczne i tylko te systemy mają dojście do danych zawartych w poszczególnych odrębnych zbiorach.

BAZA DANYCH
(data base)

— Zbiór lub zbiory utrzymywane niezależnie od systemów informatycznych zbiory te wykorzystujących, np. systemy informatyczne dotyczące pracowników, środków trwałych, materiałów korzystają ze zbioru stanowiącego wspólną bazę danych poprzez manipulator zbiorami.

mgr Jan Kozak
Strzybnica

PROGRAMOWANIE

Symbol doskonałości firmy *Hewlett-Packard*

Dzięki tym klawiszom HP-25 otwiera nowe możliwości

Nowy HP-25 jeszcze raz potwierdza wiodącą rolę firmy Hewlett-Packard wśród producentów minikalkulatorów.
HP-25 ma jasną i przekonującą koncepcję: nowoczesna technologia przy korzystnym stosunku ceny do wydajności.

Prościej nie da się programować

Programy mogą zawierać do 49 kroków programowych. Dzięki zastosowaniu złożonego kodu użycia klawiszy nawet przy funkcjach z użyciem klawisza wybiorczego registrów wymagany jest tylko jeden krok programowy. Podczas korektury lub zmiany programu można łatwo wprowadzać nowe kroki, kasować je lub zmieniać. Zaprogramować można również rozgałęzienia oraz implikacje logiczne.

„Techniczny zapis wyników" — nowa możliwość wskaźnika

Oprócz stałego i zmiennego przecinka istnieje dodatkowa możliwość stosowania tzw. „technicznego zapisu wyników": stanowi to ważną zaletę podczas używania jednostek typu kilo-, mega-, giga-, mili-, mikro-, nano-, i in. (Wynik jest przedstawiany jako wielokrotność 10 ± 3)

Nadzwyczajne możliwości programowania wybiorczego

Model HP-25 umożliwia programowanie w zakresie 72 funkcji, operacji i rozkazów.
Należą do nich wszystkie funkcje logarytmiczne i trygonometryczne (w stopniach, gradach lub miarach łukowych).

Przekształcenie współrzędnych prostokątnych we współrzędne biegunowe i odwrotnie, przeliczenie zapisu dziesiętnego w zapis sekundowo/minutowo/stopniowy, wartości średniej i standardowych odchyleń. Sumowanie. Operacje dodawania, odejmowania, mnożenia i dzielenia są we wszystkich 8 pamięciach adresowych. Możliwe jest również wprowadzanie rozkazów rozgałęzień, %, π, $\sqrt{x}$, $1/x$, x^2, y^x

Przykładowa pojemność pamięci

Trzynaście pamięci, w tym cztery robocze w układzie „piętrowym" (stack), jedna dla ostatniej wprowadzanej zmiany wartości „X", a osiem oddzielnie adresowanych z możliwością wykonywania operacji arytmetycznych.

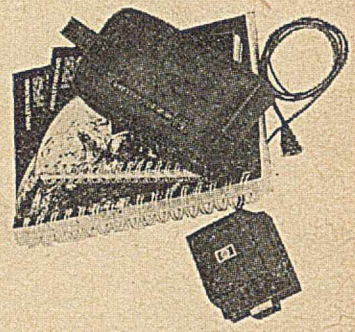
Za HP-25 przemawia wiele. Głównie nowoczesna technologia.

Oczywiście również w HP-25 zastosował Hewlett-Packard wypróbowaną technikę „piętrową". System, który jest stosowany we wszystkich innych typach minikalkulatorów firmy Hewlett-Packard, o którego zaletach przekonało się około 1 miliona użytkowników. Zalety tego systemu: sprawny i prosty. Wyniki pośrednie są zapamiętywane automatycznie i w miarę potrzeby automatycznie wywoływane.
To, że HP-25 jest doskonałym rachmistrzem ujawnia się na każdym kroku. Kalkulator jest dopasowany do dłoni. Kontaktowanie klawiszy jest niezawodne (klik-klak). Nowy typ wskaźnika umożliwia dobry odczyt przy świetle zewnętrznym. Wszystkie połączenia są pozłacane. Klawiatura jest zabezpieczona przed wilgocią. HP-25 jest dostarczany jako pełnosprawny z naładowaną baterią. Zasilacz sieciowy w etui. Poradnik. Instrukcja stosowania oraz bogaty zbiór przykładów aplikacyjnych. Oprócz tego przy kupnie jest wręczana jednoroczna gwarancja.

Żądajcie wyczerpujących materiałów informacyjnych.

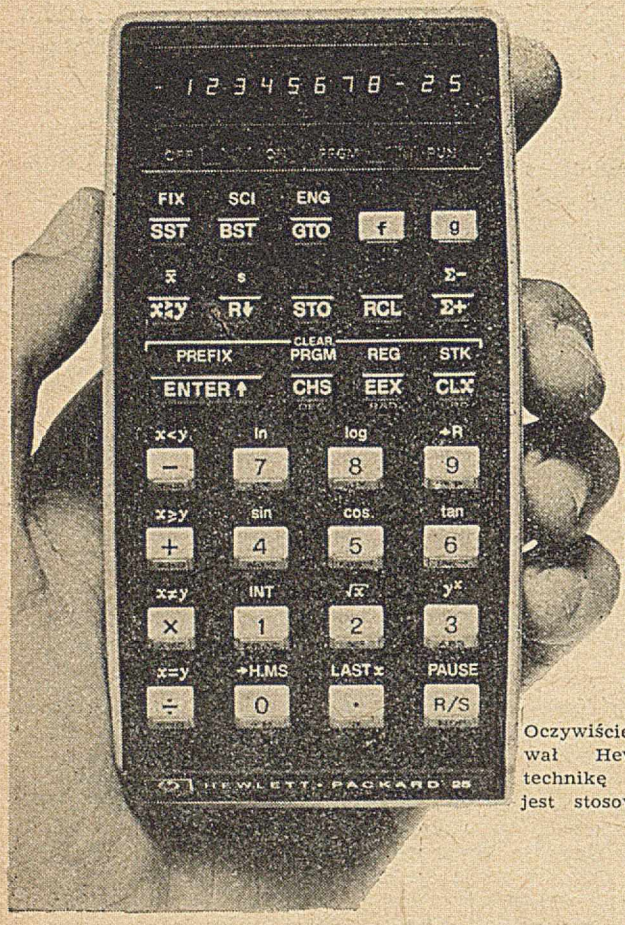
H-P25

— najnowszy członek wybitnej rodziny.



Znajdziecie nas Państwo na Międzynarodowych Targach Technicznych w Poznaniu w dniach od 6 do 17 czerwca w pawilonie USA.

Biuro Informacji Technicznej
HEWLETT-PACKARD
ul. Stawki 2, 6 piętro
00-950 Warszawa
Telefon: 39-51-87, 39-59-62,
Teleks: 812453 hepa pl



Sprzedaż i serwis w 172 punktach znajdujących się w 65 krajach.

HEWLETT hp PACKARD

WCT/410/K/76