POLITECHNIKA ŚLĄSKA
BIBLIOTEKA GŁÓWNA
Gliwice
Nr 1877/75

5 1975

# informatyka

SPIS TERŚCI ZESZYTU 5/75


Wywiad z Ministrem Łączności prof. dr *Edwardem Kowalczykiem* 1

*Marek Hołyński*
Przegląd metod optymalizacji programów dla maszyn cyfrowych 3

*Bolesław Szymański*
Sieci komputerowe i ich zastosowania do kierowania gospodarką narodową 7

*Jerzy Dańda*
System informacyjny FAMULUS 11

*Jarosław Dobosz*
ART 73B — język do przetwarzania informacji akustycznych 13

*Jerzy R. Jaworowski, Ryszard Tadeusiewicz*
Tradycyjne a nowoczesne metody przygotowywania maszynowych nośników informacji 15

*Jerzy Sukiennik*
Tendencje rozwojowe w dziedzinie pamięci cyfrowych — oprac. *Jan Ryżko* 18

TRYBUNA CZYTELNIKA — List prof. dr hab. *Władysława M. Turkiego* 21

WIADOMOŚCI PKAPI — I Wojewódzka Konferencja Informatyków w Szczecinie — *A. Nowakowski* 22

Z KRAJU — Narodziny statków pod dyktandem komputera — *Krystyn Bernatowicz* 24

Informacje różne 27—28

ZE ŚWIATA — Analiza parku komputerowego zainstalowanego w Austrii — oprac. *Leszek Nowak* 29

Informacje różne 32, 36, 40

TERMINOLOGIA — Programowanie — języki, metody translacji — oprac. *Władysław Klepacz* 33

SZTUKA KOMPUTEROWA — Sztuka komputerowa cz. I. — *Marek Hołyński* 34

PROBLEMATYKA BAZY DANYCH — Propozycja Komitetu CODASYL w dziedzinie systemów zarządzania bazą danych — *Władysław Bogucki, Witold Staniszkis* 37

PRZEGLĄD WYDAWNICTW — Recenzja książki Z. Gackowskiego: Projektowanie systemów informacyjnych zarządzania — *Stanisław Szklarzewicz* III ok. III i IV skrz.

Bibliografia wydawnictw polskich z dziedziny informatyki — oprac. *Jerzy Klamborowski* IV skrz.

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), doc. dr inż. Zbigniew GACKOWSKI, mgr inż. Marek HOŁYŃSKI, Władysław KLEPACZ, Elżbieta KOŁODZIEJSKA, doc. dr hab. Antoni MAZURKIEWICZ,

Sekretarz Redakcji mgr Krystyna WROŃSKA — Red. tech. Józef DUSZA

RADA PROGRAMOWA

Mgr inż. Antoni BOSSOWSKI, doc. dr inż. Jan FELICKI, doc. dr inż. Zbigniew GACKOWSKI, doc. dr inż. Aleksander GOLINOWSKI, 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: 00-041 Warszawa, ul. Jasna 14/16, pokój 331, tel. 27-71-40 lub centrala 28-82-61 w. 285, dyżury redakcji 10,00—13,00

Zakład Kolportażu WCT NOT, Warszawa, ul. Mazowiecka 12

Zakł. Graf. „Tamka". Z. 2. Zam. 172. Papier druk. sat IV kl. 70 g 61 + 86. Obj. 6,75 ark. druk. Nakład 5600 + 30. B-93.

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

# Informatyka

dawniej Maszyny Matematyczne

zastosowania w gospodarce, technice i nauce

Nr 5

MIESIĘCZNIK

1975

ROK XI

Maj

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

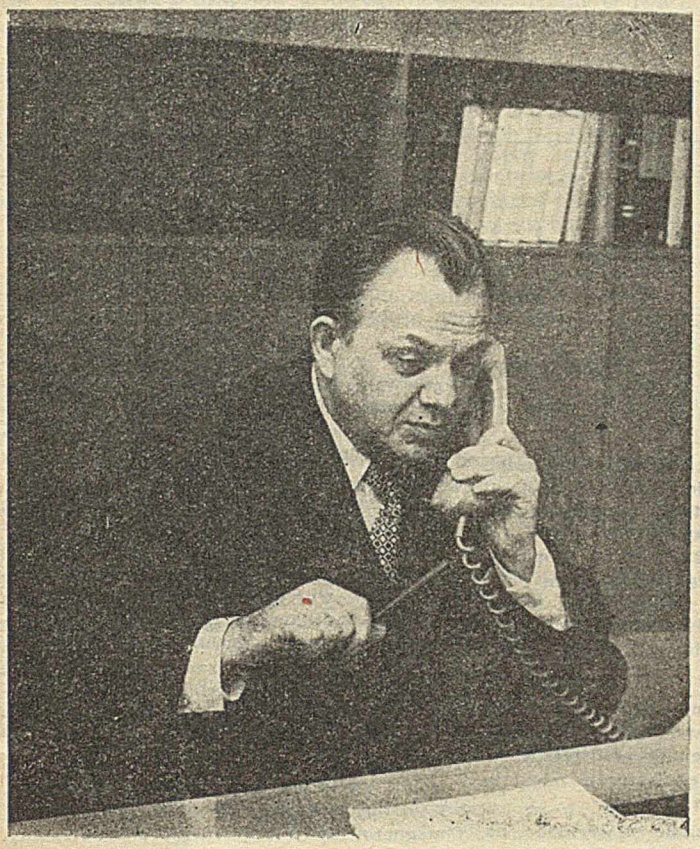


POLITECHNIKA ŚLĄSKA BIBLIOTEKA GŁÓWNA

P. 1877/75


## Wywiad z Ministrem Łączności prof. dr Edwardem Kowalczykiem

*Rozwój systemów informatycznych, sieci komputerowych, maszyn z końcówkami o zdalnym dostępie sprawił, że stają się dla nas coraz bardziej istotne problemy sieci transmisji danych. Z drugiej strony wiemy, iż do niedawna klasyczna telefonia nie dawała sobie rady z własnymi trudnościami. Jak zatem, Panie Ministrze, powinniśmy oceniać możliwości telekomunikacji w zaspokajaniu potrzeb wysuwanych przez informatykę?*



Telekomunikacja ma istotnie wiele do odrobienia. Jej opóźnienia wynikły z rozwoju innych dziedzin elektroniki, których pewne błyskotliwe osiągnięcia w zakresie sprzętu powszechnego użytku, usunęły telekomunikację nieco w cień. Obecnie jak niektórzy twierdzą sytuacja się odwróciła — telekomunikacja zaczyna przejmować pałeczkę postępu technicznego w elektronice. Dotyczy to zarówno koncepcji układowych, jak i technologii. Do idei rozwoju klasycznej telekomunikacji wrócili nawet ci, którzy wróżyli jej szybki koniec.

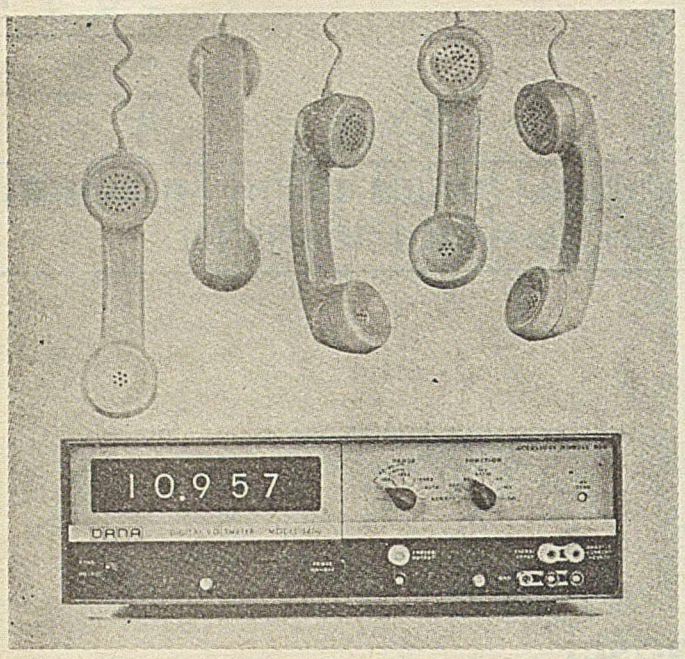
Przykładem może być telewizja wchodząca z powrotem w kabel — wiele miast na świecie przesyła lub myśli o transmisji programów telewizyjnych przez wewnętrzne linie przewodowe; ze względu na rozwój urbanistyki różne zakłócenia utrudniają tam bowiem normalną propagację.

Przetwarzanie danych za pomocą maszyn cyfrowych wymaga zebrania tych danych z miejsc czasem dość odległych od samego komputera, a następnie przesłania wyników. W tej chwili często przewozi się dane samochodem lub koleją, a jeśli jest blisko, wówczas idzie się z teczką na piechotę. Konieczne jest zatem połączenie wszystkich biorących udział w tym procesie punktów siecią teleinformatyczną, służącą do przesyłania danych z różnymi szybkościami i zabezpieczoną przed błędami. Dziś już wszyscy na świecie zdają sobie sprawę z tego, że nie będzie nowoczesnego państwa bez nowoczesnej telekomunikacji, a nowoczesnej telekomunikacji bez nowoczesnego przemysłu telekomunikacyjnego.

*Jak zatem wyglądają perspektywy naszego przemysłu?*

Na najbliższe lata przewidujemy kilkakrotny wzrost produkcji. Zakupiliśmy także licencje, które zapewniają nam odpowiednio wysoki poziom wytwarzanego sprzętu. Obecnie budowana jest w Warszawie fabryka krzyżowych central telefonicznych. W Poznaniu rozpoczynamy wytwarzanie central elektronicznych — pierwszy egzemplarz takiej centrali już podjął pracę. Nowoczesna licencyjna fabryka

kabli uruchamiana jest w Ożarowie. Nasz wysiłek koncentruje się teraz na wprowadzaniu tych nowych urządzeń w istniejącą sieć. Pamiętamy, że w kraju pracuje wiele systemów, często bardzo starych, na których wymianę nas jeszcze nie stać.




*O stanie telekomunikacji oprócz jakości i ilości sprzętu także decydują obsługujący go ludzie. Czy nasza kadra jest dostatecznie silna, aby strawić te wszystkie licencje i nadać swojej dziedzinie wystarczająco duże przyspieszenie?*

Występują tu dwa zagadnienia: ilości i poziomu kadr. Zdarza się często, że nasi fachowcy w rozmowach z przedstawicielami firm zagranicznych nie mają partnerów do rozmowy. Świadomość techniczna naszej kadry jest bardzo wysoka, brakuje nam tylko trochę możliwości technologicznych. Mamy kłopoty z zapleczem naukowo-dydaktycznym. Kadra naukowa się już w dużej części zestarzała, nowych pracowników przybywa za mało. Jest to problem szczególnie istotny np. z punktu widzenia związków telekomunikacji z informatyką. Środowisko naukowe odpowiada przecież za opracowanie problemu węzłowego rozwoju sieci telekomunikacyjnej państwa, która powinna uwzględniać również potrzeby informatyki.

*Wspomniał Pan Minister o naszych możliwościach technologicznych. Jest to chyba niezwykle istotne — współczesny sprzęt telekomunikacyjny ma z pewnością bardzo duże wymagania jeśli chodzi o stosowane w nim układy elektroniczne.*

Urządzenia telekomunikacyjne osiągnęły taki stopień złożoności i precyzji działania, że do ich konstrukcji konieczne są układy elektroniczne o bardzo dużej dokładności, o ostrych parametrach technicznych i niezawodnościowych. Produkcja sprzętu profesjonalnego zmusza nas więc do przeorientowania bazy technologicznej i wytwarzania takich właśnie układów. Mogą one ponadto zostać użyte równie dobrze do popularnych urządzeń radiowych lub telewizyjnych podnosząc przy okazji jego jakość.

*Jeśli więc te problemy zostaną pomyślnie rozwiązane możemy się spodziewać, że telekomunikacja wywiąże się z zadań stawianych przed nią przez informatykę. Ale relacja jest chyba dwustronna. Jakie są korzyści telekomunikacji ze współpracy z informatyką?*

Należałoby zadać sobie tu pytanie: poprzez jakie swoje właściwości informatyka oddziaływuje na technikę telekomunikacyjną. Oczywiście, że głównie poprzez koncepcję wykorzystania sygnałów cyfrowych w komunikacji wewnętrznej maszyn — co inspiruje telekomunikację do zastosowania tego rodzaju sygnałów w transmisji dalekopisowej. Spowodowało to wprowadzenie do eksploatacji systemów kodo-impulsowych — których krotności stale się powiększają (24, 32, 120...).

Sieć telekomunikacyjna staje się elementem organizującym przetwarzanie danych. Stoimy przed zagadnieniem sieci telekomunikacyjnej dostosowanej do przesyłania danych, sterowania współpracą komputerów i komutowaniu łącz oraz pracy w systemach informatycznych przestrzennie rozległych, które naszym zdaniem są najwłaściwszym kierunkiem rozwoju informatyki na usługach gospodarki narodowej i zarządzania.

rozmawiał: **Marek Hołyński**

---

## Rozpoczął się II Telewizyjny Kurs Informatyki

**Wykład profesora Andrzeja Straszaka inaugurował emisję II Telewizyjnego Kursu Informatyki organizowanego przez Ośrodek Badawczo-Rozwojowy Informatyki i Telewizję Polską. Poprzedniego kursu nadawanego w ubiegłym roku wysłuchało ponad 16 tysięcy osób. Ponad 60 procent uczestników ukończyło kurs i otrzymało, po zdaniu egzaminów, świadectwo.**

**Kurs ten przeznaczony jest w zasadzie dla kadr kierowniczych przedsiębiorstw, kombinatów, zjednoczeń i ministerstw, a także pracowników-organizatorów elektronicznego przetwarzania danych.**

**W programie przewidziane są wykłady na temat podstaw informatyki, jej zastosowania, systemów informatycznych i przygotowania organizacyjnego użytkowników systemów. Nadanych zostanie 39 wykładów, prowadzonych przez wybitnych naukowców i specjalistów praktyków w tej dziedzinie.**

**Choć kurs już wystartował możliwe są jeszcze zapisy. Przyjmują je punkty konsultacyjne w Zakładach Elektronicznej Techniki Obliczeniowej we wszystkich miastach wojewódzkich. W Katowicach punkt konsultacyjny zorganizowano w Ośrodku Postępu Technicznego przy ul. Buczka 12. W Warszawie działa punkt konsultacyjny we Włochach ul. Skrońskiego 7A przy Ośrodku Badawczo-Rozwojowym Informatyki.**

**W punktach tych prowadzone będą zajęcia konsultacyjne, testy i seminaria. Wykorzystane w nich będą magnetofony, magnetowidy i inny sprzęt audiowizualny. (kr)**

BOLESŁAW SZYMAŃSKI
Centralny Ośrodek Informatyki
Politechniki Warszawskiej

681.322.06

# Przegląd metod optymalizacji programów dla maszyn cyfrowych

Artykuł jest omówieniem metod optymalizacji programów dla maszyn cyfrowych ze szczególnym uwzględnieniem dwóch aspektów — metod maszynowych i maszynowo niezależnych. Podano przykłady oraz bogatą literaturę przedmiotu.

Metody optymalizacji opisane w artykule służą zmniejszeniu kosztów wykonania programów. Optymalizują one programy pod względem czasu wykonania, rzadziej pod względem wielkości pamięci zajętej przez sam program lub przez jego pola robocze. Proces optymalizacji jest szczególnie przydatny tam, gdzie program dla maszyny układany jest automatycznie. Z sytuacją taką spotykamy się w przypadku programów tworzonych przez kompilatory języków wyższego rzędu. Z tego powodu kompilatory są na ogół wyposażone w procedury optymalizujące. Używanie takich procedur wydłuża czas kompilacji. Warunkiem efektywności ich stosowania jest więc wielokrotne wykorzystanie utworzonego programu wynikowego. By zapewnić spełnienie tego warunku, tworzy się czasami dwa kompilatory jednego języka — jeden do fazy uruchamiania programu, o rozbudowanej diagnostyce błędów — drugi do generowania programu wynikowego, o rozbudowanym systemie optymalizacji (McHaffie [16]). Innym sposobem jest utworzenie dla języka specjalnego programu optymalizatora. Optymalizator przekształca program w języku wyższego rzędu w równoważny mu, bardziej optymalny program w tym samym języku (Schneck [20]).

Bardziej optymalny oznacza, że koszt wykonania programu wynikowego utworzonego przez kompilator nie wyposażony w procedury optymalizujące będzie niższy w przypadku translacji programu przekształconego. Zaletą optymalizatorów jest fakt, że mogą być one wykorzystywane na każdej maszynie cyfrowej zawierającej kompilator określonego języka. Wadą jest to, że nie mogą wykorzystywać w procesie optymalizacji szczególnych własności maszyn cyfrowych takich, jak: lista rozkazów, szybkość wykonywania operacji, liczba procesorów, liczba akumulatorów i akumulatorów indeksowych itd. Wygodnie jest podzielić metody optymalizacji na dwie grupy:

— metody maszynowo ukierunkowane wykorzystujące szczególne własności maszyn, dla których tworzony jest kompilator

— metody maszynowo niezależne wykorzystujące ogólne własności maszyn cyfrowych lub egzekucji programów.

Z definicji tych wynika, że optymalizatory wykorzystywać mogą tylko drugą grupę metod.

Jak widać na rys. 1, podział metod optymalizacji określa poziom języka i moment procesu kompilacji, w którym optymalizacja zachodzi. Grupy metod optymalizacji można więc zdefiniować także w następujący sposób:

— metody maszynowo ukierunkowane stosowane są do programu wynikowego lub w trakcie generowania tego programu, czyli działają w trakcie przejścia z języka pośredniczącego na wynikowy lub na poziomie języka wynikowego

— metody maszynowo niezależne stosowane są do programu w języku pośredniczącym, czyli działają na poziomie tego języka.

## Metody ukierunkowane maszynowo

### Przydział akumulatorów *(register allocation)*

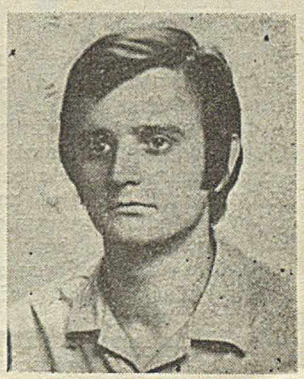
Zadaniem tego typu metod jest przydzielenie wynikom częściowym akumulatorów maszyny w taki sposób, by zminimalizować liczbę przesłań wyników częściowych do i z pamięci. Stosowanie takich metod jest bardzo krzystne, gdyż zmniejszają one czas wykonania (eliminując przesłania do i pobrania z pamięci), wielkość pamięci zajętej przez pola robocze programu (eliminując pola na wyniki częściowe), jak i przez sam program (eliminując instrukcje przesłania do i pobrania z pamięci). Istnieje kilka algorytmów rozwiązania tego problemu, od najprostszych — przydzielania akumulatorów wynikom częściowym w pętlach (Busam [7]), w bloku wyrażeń arytmetycznych (Breuer [6]), w całym programie wykonywanym na maszynie o $N$ takich rejestrach (Sethi [21]) aż do uogólnienia problemu na abstrakcyjnie zdefiniowane działania (Beatty [5]). Szczególny przypadek tego zagadnienia — przydział akumulatorów indeksowych w programie jest rozwiązany przez Kennedy'ego [15].

### Zrównoleglanie wyrażeń arytmetycznych *(parsing methods)*

Celem tych metod jest ustalenie postaci wyrażenia arytmetycznego pozwalającego zrównoleglić działania. Stosowane są w maszynach o wielu procesorach, gdzie możliwe jest wykonywanie kilku operacji arytmetycznych w jednym czasie. Niektóre algorytmy rozwiązania tego problemu podaje Nieżwicki [18], inne Breuer [6] i Beatty [5]. Wszystkie one abstrahują od różnego czasu wykonania operacji.

Jak widać na rys. 2a klasyczna metoda dla wyrażenia $A+C+E\times F+D$ wymaga 3 kroków (1 krok mnożenia i 2 dodawania).

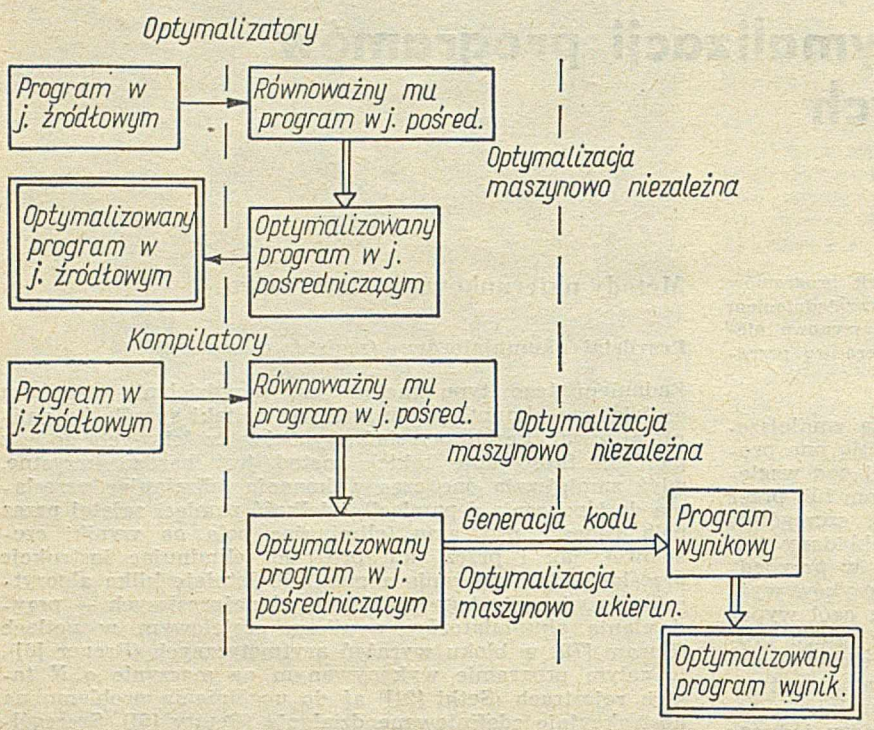
Ponieważ mnożenie trwa kilka razy dłużej niż dodawanie to można w czasie wykonywania mnożenia $E\times F$ obliczyć wartość $A+C+D$ — rys. 2b. W ten sposób wystąpią tylko dwa kroki (1 krok mnożenia i 1 dodawania). Wartość metody poważnie obniża fakt, że nie zawsze można zmienić kolejność wykonywania operacji, czy to ze względu na efekty uboczne procedur, czy na dokładność wyliczeń w arytmetyce zmiennoprzecinkowej.



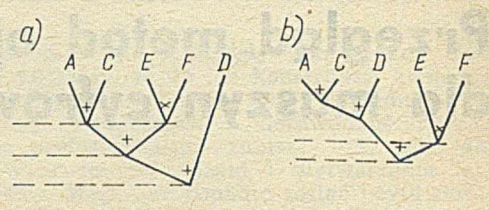
Mgr inż. BOLESŁAW SZYMAŃSKI ukończył Wydział Elektroniki Politechniki Warszawskiej, specjalność maszyny matematyczne (1973). Zdobył nagrodę II stopnia w X Ogólnopolskim Konkursie: „Młoda myśl dla kraju" za pracę magisterską pt. Kompilator języka JOS w JOSK-u.
Pracuje w Centralnym Ośrodku Informatyki PW, jako asystent (1973). Zajmuje się problemami tworzenia i optymalizacji translatorów.

**Strzałkowanie wyrażeń** *(anchor pointing)*

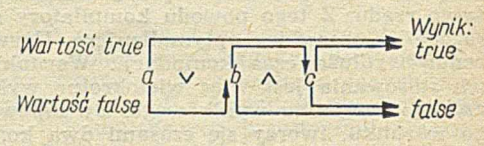
Metoda dotyczy wyrażeń boolowskich (ogólniej wyrażeń, których liczność zbioru potencjalnych wartości jest niewielka). Wyrażenia boolowskie zamieniane są na sekwencje badań wartości i skoków. Wartość operacji wyznaczana jest, o ile to możliwe na podstawie pierwszego argumentu, w przeciwnym przypadku przechodzi się do badania drugiego argumentu. Wartość ta wyznacza następny element wyrażenia do wyliczenia [9]. Zasadę tę dla wyrażenia $\vee b \wedge c$ ilustruje rys. 3. W przypadku zastosowania tej metody uzyskuje się przyspieszenie wyliczania wyrażeń boolowskich i nieangażowanie do ich obliczania arytmometru, co w niektórych maszynach może być istotne.




Rys. 1. Schemat działania optymalizatorów i kompilatorów




Rys. 2. Zrównoleglanie wyrażeń arytmetycznych




Rys. 3. Strzałkowanie wyrażeń

**Przeglądanie programu wynikowego** *(peephole optimization)*

Jest to metoda polegająca na przeszukiwaniu programu wynikowego w odcinkach o długości od kilkunastu do kilkuset instrukcji (w celu usunięcia niepotrzebnych instrukcji), łączenia kilku instrukcji w jedną lub usunięcie niepotrzebnych skoków przez przestawienie odcinków programu. Korzyści odnosi się zarówno przez skrócenie czasu wykonania programu, jak i zmniejszenia wielkości pamięci zajętej przez sam program. McKeeman [17] stwierdził, że stosunek zysku do czasu zużytego na optymalizację jest największy dla długości odcinka badanego równej 10—12 instrukcji, choć wraz z długością tego odcinka zyski rosną.

**Szczególne przypadki operacji arytmetycznych** *(special case code generation)*

W pewnych sytuacjach można zastąpić operację arytmetyczną równoważną jej operacją, lecz szybciej wykonywaną. Przykładem może być zastąpienie:
— potęgowania mnożeniem, a↑2 lub b↑3 można zastąpić odpowiednio operacjami $a \times a$ i $b \times b \times b$
— mnożenia (dzielenia całkowite) przesunięciem, $I \times 2^n$ lub $I \div 2^n$ przy $I$ będącym zmienną całkowitą można zamienić na odpowiednie przesunięcie
— dodawania (mnożenia) jednym argumentem, $a + 0$ i $a \times 1$ można zastąpić przez $a$ (wyrażenia $a + 0$, $a \times 1$ mogą się pojawiać w programie jako skutek działania procedur optymalizujących
— dzielenia mnożeniem (Gries [11]), $A/B$ w pętli, w której $B$ ma wartość stałą można zastąpić sekwencją: podstawienie $B1 := 1/B$ przed pętlą, a w pętli operacja $A \times B1$. Zastrzeżenia budzi w tym przypadku sprawa równoważności obu działań ze względu na ograniczoną dokładność arytmetyki maszynowej.
Z tego typu optymalizacjami można spotkać się dość często. Stosowane są one w różnym zakresie, wyznaczonym przez częstość występowania każdej z sytuacji w języku źródłowym.

**Zmienne sprzężone** *(shadow variable)*

Metoda polega na przechowywaniu danych w dwóch postaciach (np. decymalnej i binarnej). Pozwala to czasami uniknąć przekodowań (Allen [4]).

**Rozkład pamięci roboczej** *(storage mapping)*

Celem działania tej metody jest odpowiednie ułożenie pól pamięci roboczej. Gdy adresowanie jest z bazą dąży się do minimalizacji zmian bazy; razem występujące zmienne umieszcza się pod wspólną bazą (Allen [4]). Podobnie, w przypadku adresowania indeksowego bada się pary zmiennych, by określić które zmienne muszą występować obok siebie w pamięci, by zmniejszyć liczbę operacji adresowych (Yershow [22]).

**Kolejność instrukcji** *(instruction scheduling)*

Metoda ta jest stosowana w maszynach, w których w trakcie działania arytmometru możliwy jest dostęp do pamięci operacyjnej. Zmienia się kolejność instrukcji w ten sposób, by przeplatać instrukcje z dostępem do pamięci z instrukcjami bez tego dostępu, co skraca czas wykonania programu (Allen [4]).

## Metody maszynowo niezależne

**Dołączanie podprogramów** *(procedure integration)*

Sposób dołączenia podprogramów ma istotny wpływ na możliwości optymalizowania całego programu. Z punktu widzenia procesu optymalizacji wyróżnić można, w zależności od rodzaju dołączenia, następujące typy podprogramów:

— zamknięty *(closed)* — program i podprogram stanowią dwie całkowicie niezależne części. Nie ma możliwości wspólnej ich optymalizacji. Zakładać należy, że każda zmienna mogła w trakcie działania podprogramu zamienić swoją wartość

— otwarty *(open)* — zastąpienie wywołania treścią podprogramu. Zaletą takiego rozwiązania jest fakt, że całość podlega wspólnej optymalizacji, wadą — zwiększenie długości programu. Z tego powodu rozwiązanie jest korzystne dla małych funkcji i procedur standardowych (McHaffie [16]) lub wtedy, gdy podprogram wołany jest tylko raz (Yershow [22])

— pośrednio otwarty *(semiopen)* — stosowany w programach podzielonych na moduły. Do każdego modułu dołącza się podprogramy w nim wywoływane. Pozwala to razem optymalizować treść modułu i treść podprogramu, a jednocześnie cały program nie rozrasta się zbytnio

— pośrednio zamknięty *(semiclosed)* — podprogramy są kompilowane najpierw i informacje o nich wykorzystywane przy optymalizacji całego programu. Ważną zaletą tego rozwiązania jest możliwość określenia, które zmienne we wnętrzu podprogramu ulegają zmianie.

**Przekształcenia pętli** *(loop transformation)*

Najważniejszym sposobem tego typu jest zwiększanie kroku przez powielanie wnętrza pętli *(jamming)*. Zmniejsza się w ten sposób liczbę badań warunku wyjścia z pętli. Metoda ta jest więc istotna w algorytmach numerycznych, gdzie wykonywane są pętle o dużej liczbie kroków i niewielkim wnętrzu. Jak podaje Pager [19], stosowanie tej metody pozwala skracać czas wykonania programu — nawet do 25%. Zmniejszenie liczby badań warunku wyjścia z pętli można też osiągnąć poprzez łączenie pętli *(loop fusion)* o jednakowej liczbie przejść sąsiadujących ze sobą (Allen [4]).

**Optymalizacja globalna**

Ze względu na wagę i uniwersalność zastosowań czterech dalej omówionych metod optymalizacji rozpatruje się je wspólnie i określa nazwą — optymalizacja globalna *(global optimization* Abel [1], Allen [2], Cocke [8]). Do metod tych zalicza się:

- **eliminację wspólnych podwyrażeń** *(redudant subexpression elimination)*. Celem metody jest wyszukiwanie wspólnych podwyrażeń w wyrażeniach arytmetycznych i eliminacja powtórnych wyliczeń poprzez zachowanie wyniku pierwszego wyliczenia. Wspólne podwyrażenia mogą być na poziomie języka źródłowego ukryte, jak to dzieje się w przypadku wyliczania adresów w tablicach. Problem rozpoznania identyczności jest bardzo trudny, zwłaszcza przy złożonych wyrażeniach. Badanie wszystkich możliwości jest mało efektywne i stwarza konflikty. Praktycznym wyjściem jest alfabetyczne układanie argumentów (o ile dopuszczalna jest zmiana ich kolejności w wyrażeniu). Breuer [6] podał heurystyczny algorytm faktoryzacji wyrażeń arytmetycznych, ale nie zawsze daje on dobre wyniki

- **przesuwanie kodu** *(code motion)*. Metoda polega na przesuwaniu punktu wykonania obliczeń czy działań w celu usunięcia ich z obszarów częściej wykonywanych lub w celu zmniejszenia liczby instrukcji. Zmniejszenie takie osiąga się, gdy kilka identycznych wyliczeń w gałęziach programu zastępuje się jednym wyliczeniem przed rozejściem w te gałęzie. Częstość wykonywania obszarów programu jest znana, wobec czego przyjmuje się, że częściej wykonywane jest wnętrze pętli niż jej otoczenie. Z tego powodu przesuwanie kodu zachodzi z wnętrza pętli do jej otoczenia.

- **propagacja stałych** *(constant folding)*. Wszędzie, gdzie występują działania o znanych operandach wykonuje się je w trakcie optymalizacji. Czasami można dodatkowo wyeliminować instrukcje, do których, ze względu na znany wynik testu, nie można dotrzeć.

- **redukcja wyrażeń w pętli** *(strength reduction)*. Metoda polegająca na zamianie wyrażeń z rekursywnie zdefiniowanymi zmiennymi na wyrażenia rekuryswnie zdefiniowane. Zmienna rekursywnie zdefiniowana w pętli zmienia się o stałą wartość (krok). Wyrażenia rekursywnie zdefiniowane kolejną swoją wartość mają uzależnioną od poprzedniej. Przykładem może być pętla w ALGOL-u:

**for** i : = 1 **step** 1 **until** n **do** k : = 5 × 1 + 2↑i;

gdzie wyrażenia 5 × i, 2↑i można zastąpić rekursywnie zdefiniowanym *m:* = *m* + 5 *p:* = p× 2 i pętla użyska postać:

*m*: = 0; *p*: = 1; **for** i: = 1 **step** 1 **until** n **do begin** *m*: *m*+5; *p*:= *p*×2; *k*:= *m*+*p* **end**;

Często, jak w tym przykładzie rola zmiennej rekursywnie zdefiniowanej, sprowadza się po optymalizacji jedynie do badania warunku wyjścia i można ją wtedy zastąpić nowo wygenerowaną zmienną *(test replacement)*. W ten sposób uzyska się w rozpatrywanym przykładzie:

p: = 1; **for** m: = 5 **step** 5 **until** 5 × n **do begin** p:=px2, k:=m+p **end**;

Optymalizacja globalna wymaga opisu całego programu przy użyciu pewnych podstawowych pojęć. Należą do nich:

— **blok podstawowy** (w skrócie blok), ciąg instrukcji, które zawsze są wykonywane w tej samej kolejności. Oznacza to m.in., że istnieje dokładnie jedno wejście do bloku — jego początek (instrukcja wykonywana jako pierwsza) i dokładnie jedno wyjście z bloku — jego koniec (instrukcja wykonywana jako ostatnia). Bloki mogą mieć różne długości

— **punkt definicji zmiennej**, miejsce w programie, w którym zmienna uzyskuje nową wartość. Określenie położenia punktów definicji w programie pozwala ocenić możliwość optymalizacji, np.: jeżeli argumenty operacji mają wszystkie punkty definicji poza programem, to można wykonać ją w trakcie optymalizacji, poza pętlą, to można wykonać ją na zewnątrz pątli, a do pętli przesłać od razu wynik, jeżeli dwa wyrażenia bloku są formalnie identyczne to są równoważne wtedy i tylko wtedy, gdy ich argumenty mają te same punkty definicji w tym bloku

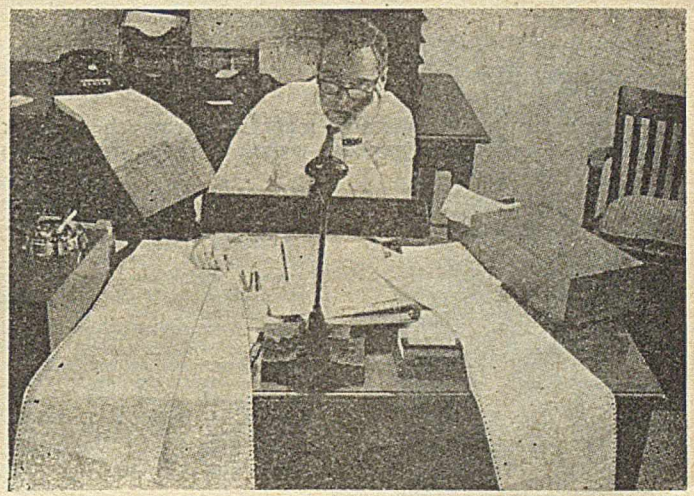
— **punkt użycia zmiennej**, miejsce w programie, w którym zmienna użyta jest w wyrażeniu.

Dla języków o jawnie zdefiniowanych pętlach (ALGOL, FORTRAN) można próbować dokonać optymalizacji globalnej na podstawie informacji o pętlach, o rozkładzie bloków oraz punktów definicji i użycia (Busam [7]). Instrukcje skoków w programie utrudniają lub wręcz uniemożliwiają taką optymalizację. Uwzględnienie skokowych zmian sterowania programu wymaga subtelniejszych metod.

Program przedstawia się w postaci grafu skierowanego, którego wierzchołki stanowią bloki podstawowe, a krawędzie reprezentują przepływ sterowania. Problem optymalizacji globalnej w odniesieniu do tak przedstawionego programu rozwiązywany jest dwoma różnymi metodami.

**Metoda cykli** *(strongly connected regions)*

Oparta jest ona na wyszukiwaniu w grafie programu pętli (Gries [11], Hopkins [12]). Graf programu rozbija się na ciąg podgrafów, z których każdy jest cyklem, a kolejne podgrafy w ciągu są rozłączne lub poprzedzający jest zawarty w następnym. Ponieważ w momencie wyboru kolejnego elementu ciągu tylko jeden, jeśli w ogóle, cykl



spełnia warunek posiadania jednego wejścia i jednego wierzchołka poprzedzającego — a warunek ten spełnia pętla — to dąży się do wybrania takich właśnie cykli. Rozbicie grafu programu na ciąg cykli odpowiada w przypadkach prostych programów rozbiciu na pętle od najgłębszych poczynając. Proces optymalizacji działa na kolejne elementy ciągu cykli. Każdy z nich zastępowany jest po zoptymalizowaniu pojedynczym wierzchołkiem w grafie i nie podlega dalszej analizie.

**Metoda interwali**

W metodzie tej wychodzi się od określenia warunków optymalizacji w postaci układu równań boolowskich. Oznaczmy przez $x_i$ zmienną boolowską, której wartość jest równa **true**, gdy rozpatrywane wyrażenie jest dostępne na wyjściu bloku $i$, $a_i$ wartość boolowską równą **true**, gdy nie ma punktów definicji wyrażenia w bloku $i$, $b_i$ wartość boolowską równą **true**, gdy w bloku $i$ wartość wyrażenia jest wyliczana poza wszystkimi punktami jego definicji, $P(i)$ zbiór bloków, dla których istnieje krawędź łącząca blok z $i$-tym blokiem. Równanie na dostępność wyrażenia na wyjściu bloku $i$ będzie przy takich oznaczeniach następujące:

$$x_i = a_i \wedge \Pi x_j \vee b_i \qquad (1)$$
$$j \in 0\,(i)$$

Współczynniki $a_i$, $b_i$ można łatwo wyznaczyć na podstawie statycznej analizy bloku $i$. Zestawiając równanie (1) dla wszystkich bloków programu uzyskuje się układ równań boolowskich, którego rozwiązanie określa dostępność wyrażenia na wyjściu wszystkich bloków. W analogiczny sposób można określić równania na warunek przesunięcia kodu (Earnest [10]), bezpieczeństwa przesunięcia kodu (z punktu widzenia przerwań programowych, Kennedy [14]), czy wreszcie analizy oddziaływania instrukcji na przebieg programu i odwrotnie, szukania źródeł stanu programu (Kennedy [13], Allen [3]).

W celu rozwiązania układu równań boolowskich typu (1) wprowadza się pojęcie interwalu. Interwal jest to zbiór bloków osiągalnych z danego wierzchołka (zwanego głową). Graf programu rozbija się na ciąg interwali, wybierając jako pierwszą głowę wejście do programu, a jako następną wierzchołek, do którego dochodzi krawędź z poprzedniego interwala i powtarzając wybieranie następnej głowy aż do wyczerpania wszystkich wierzchołków grafu.

Z danego grafu otrzymuje się jego pochodną przez zastąpienie interwali pojedynczymi wierzchołkami. Na ogół proces tworzenia grafów pochodnych kończy się otrzymaniem pojedynczego wierzchołka (zawsze można to osiągnąć poprzez rozszczepianie bloków). Kolejne pochodne grafu odpowiadają rugowaniu pętli od najgłębszych poczynając. Rozwiązanie układu równań boolowskich rozpoczyna się od najwyższej pochodnej, dla której jest ono trywialne, lecz dostarcza warunków początkowych dla niższej pochodnej. Dzięki temu można rozwiązać układ dla tej pochodnej i uzyskuje się w ten sposób warunki początkowe dla następnej niższej pochodnej itd. aż do grafu programu. Ponieważ dla przechowania zmiennej boolowskiej wystarcza 1 bit, wykonując operacje boolowskie na całych słowach maszynowych można rozwiązywać jednocześnie układ równań dla kilkudziesięciu (kilkunastu) wyrażeń.

## Podsumowanie

Wraz z rozwojem możliwości maszyn cyfrowych i zwiększaniem się znaczenia języków wyższego rzędu wzrasta rola metod optymalizacji programów. Zwiększenie liczby użytkowników znających jedynie języki wyższego rzędu pozwala zakładać, że programy przekazywane do kompilacji będą mało optymalne. Efektywność programów wynikowych zapewnić więc muszą same translatory. Uwzględnienie aspektów optymalizacji jest przy tworzeniu nowych kompilatorów nieuniknioną koniecznością. Przegląd metod optymalizacji dokonany w artykule może, jak się wydaje, pomóc przy określeniu możliwości optymalizacji lub wyborze konkretnego rozwiązania.

LITERATURA


[1] Abel N. E., Bell J. R.: Global optimization in compilers: an unified approach, First USA — Japan Computer Conference Proceedings, 1972, p. 437—331

[2] Allen F. E.: Program Optimization, Annual Review in Automatic Programming, vol. 5, 1969, Pergamon Press, p. 239—307

[3] Allen F. E.: A method for Determining Program Data Relationships, International Symposium in Theoretical Programming, vol. 5, 1974, Berlin, p. 299—308

[4] Allen F. E., Cocke J.: A catalogue of Optimizing Transformations in Design and Optimization of Compilers, Courant Computer Science Symposium vol. 5, 1971

[5] Beatty J. C.: An Axiomatic Approach to Code Optimization for Expressions, JACM, vol. 19, no. 4, 1972, p. 613—640

[6] Breuer M. A.: Generation of Optimal Code for Expressions via Factorization, CACM, vol. 12, no. 6, 1969, p. 333—340

[7] Busam V. A., Englund D. E.: Optimization of Expressions in Fortran, CACM vol. 12, no. 12, 1969, p. 666—674

[8] Cocke J., Schwarz J. T.: Programming Languages and Their Compilers, Courant Institute of Mathematical Science, New York Uniwersity, 1970

[9] Dobosz J., Halski M., Szymański B.: Translator języka JOSK dla m.c. ODRA 1304, Archiwum Automatyki i Telemechaniki, w druku

[10] Earnest Ch.: Some Topics in Code Optimization, JACM, vol. 21, no. 1, 1974, p. 76—102

[11] Gries D.: Compiler Construction for Digital Computer, John Wiley Sons, New York 1971

[12] Hopkins M.: Object code generation and problem peculiar to PL/1, International Seminar and Advanced Programming System 1970

[13] Kennedy K.: A. Global Flow Analysis Algorithm, JCM, vol. 3, no. 1, 1971, p. 5—15

[14] Kennedy K.: Safety of Code Motion, JCM, vol. 3, no. 2/3, 1972, p. 117—130

[15] Kennedy K.: Index Allocation in Straight Line Code and Simple Loops, Courant Computer Science Symposium, vol. 5, 1972

[16] McHaffie R.: PL/1 Optimising and Checkout Compilers, Data processing vol. 13, no. 1, 1971, p. 22—26

[17] Mc Keeman W. M.: Peephole Optimization, CACM, vol. 3, no. 7, 1965, p. 443—444

[18] Nieżwicki W.: Metody zrównoległeń w wyrażeniach arytmetycznych, praca magisterska, PW, W. Elektroniki, KBMM 1970

[19] Pager D.: Some Notes on Speeding Up Certain Loops by Means Software, Firmware and Hardware Means, IEEE Trans. Comput., vol. C-21, no. 1, p. 97—100, Jan. 1972

[20] Schneck P. B., Angel E. A.: FORTRAN to FORTRAN optimising compiler. The Computer Journal, vol. 16, no. 4, 1973, p. 322—330

[21] Sethi R., Ullman J. D.: The Generation of Optimal Code for Arithmetic Expressions, JACM vol. 17, no. 4, 1970, p. 715—728

[22] Yershow A. P.: Alpha — An Automatic Programming System of High Efficiency, JACM, vol. 13, no. 1, 1966, p. 17—24


CACM — Communications of the ACM
JACM — Journal of the ACM
JCM — International Journal of Computer Mathematics

JERZY DAŃDA
Instytut Maszyn Matematycznych
Warszawa

681.322:621.327.8

# Sieci komputerowe i ich zastosowania do kierowania gospodarką narodową

Opracowanie jest próbą zaprezentowania problemów technicznych (sprzętowych i programowych), organizacyjnych oraz naukowych wynikających przy próbach realizacji tzw. niejednorodnych sieci komputerowych i rozproszonych banków danych.

Przez określenie niejednorodne sieci komputerowe będziemy rozumieli w niniejszym opracowaniu zespoły komputerów połączonych siecią transmisji danych, przystosowane do rozwiązywania zadań, wykonywanych przez kilka lub wszystkie komputery tej sieci. Sieć tworzyć będą komputery różnych typów, różnych producentów i z różnym oprogramowaniem firmowym, włączając takie, w oprogramowywaniu których nie przewidywano pracy w sieci komputerowej. Przez określenie rozproszone banki danych będziemy rozumieli banki danych, w których informacje są rozmieszczone fizycznie w różnych komputerach, np. należących do niejednorodnej sieci komputerowej.

Natomiast sposób opisywania danych, metody dostępu do informacji i sposoby ich modyfikowania będą jednolite w całym rozproszonym banku danych. Taki sposób określenia odróżnia „rozproszone banki danych" od „zespołów banków danych", które również mogą być zawarte w niejednorodnej sieci komputerowej.

Celem opracowania jest zaprezentowanie szerszemu gronu osób problemów technicznych (sprzętowych i programowych) i w ograniczonym zakresie — organizacyjnych, a także tematów naukowo-badawczych wynikających przy próbach realizacji tego typu sieci i banków danych.

## Geneza niejednorodnych sieci komputerowych

Niejednorodne sieci komputerowe (lub raczej: ich zaczątki) często powstają spontanicznie, w momencie gdy jakaś organizacja dysponuje kilkoma komputerami, zazwyczaj różnych typów, dobrze oprogramowanymi (systemowo i użytkowo) i odczuwa niedobór mocy obliczeniowej. Takie były motywy budowy m.in. kilku mini-sieci realizowanych przez niektóre ośrodki uczelniane w USA, a opisanych w publikacjach kongresu IFIP 1968 r.

Nieco inne były motywy rozpoczęcia największego z dotychczasowych przedsięwzięć w tej dziedzinie — sieci komputerowej ARPA [1]. Chodziło tu przede wszystkim o udostępnienie każdemu uczestnikowi tego przedsięwzięcia unikalnych maszyn, wchodzących w skład tej sieci, a zwłaszcza komputera ILLIAC IV, który ze względu na wielką moc obliczeniową i stosunkowo wąską specjalizację nie może być lokalnie w pełni wykorzystany.

Drugim celem było stworzenie swoistego poligonu doświadczalnego w zakresie transmisji danych w sieciach łączności wyposażonych w zabezpieczenia uniemożliwiające kompletny „paraliż sieci". W związku z tym dużą uwagę poświęcono takiej organizacji sieci transmisji danych, aby nie zawierała ona punktów zagrożenia, jakimi są zwykle centralne punkty dyspozycyjne. W sieci ARPA nie ma takich punktów, ponieważ każdy uczestnik jest w pewnym sensie równouprawniony. Wreszcie wielka różnorodność typów maszyn w sieci ARPA wymagała sprawnego rozwiązania kwestii komunikacji między nimi. Zadanie komunikacji typu „każdy z każdym" przydzielono tu specjalnym komputerom pośredniczącym IMP *(Interface Message Processor)*.

Sieć ARPA jest powoli, lecz systematycznie rozwijana. I tak np. w roku 1973 przyłączono do tej sieci terminal w Institute of Computer Science w Londynie, za pomocą linii łączności satelitarnej. Przewiduje się dalsze „umiędzynarodowianie" tej sieci, co ze względu na jej wybitnie uczelniany charakter będzie stosunkowo łatwe w realizacji.

Wreszcie należy przewidywać, że niejednorodne sieci komputerowe będą powstawać przy realizacji hierarchicznych systemów informatycznych kierowania gospodarką narodową.

W systemach takich, niejednorodne sieci komputerowe stanowią, moim zdaniem, najkorzystniejszą bazę technicznej realizacji. Dotychczas koncepcje takich systemów łączono zazwyczaj z jednorodną siecią komputerów.

Realizacja komputerowo wspomaganych hierarchicznych systemów kierowania gospodarką narodową jest niewątpliwie poważnym zamierzeniem, rodzącym wielką liczbę problemów ekonomicznych, politycznych, społecznych i psychologicznych. Wśród nich problemy informatyczne stanowią część niewątpliwie istotną, choć przecież nie najważniejszą. Należałoby więc przyjąć jako zasadę taką organizację prac informatycznych, by informatyka nie generowała nowych problemów, a raczej starała się jak największą ich liczbę zlikwidować, lub umniejszyć ich znaczenie.

Jeśli system taki byłby realizowany w oparciu o jednorodną sieć komputerową wynikają następujące problemy:

- każdy użytkownik i współuczestnik takiej sieci ma ograniczony wybór dostawcy sprzętu
- użytkownik posiadający już sprzęt innego może stanąć wobec konieczności nowego zakupu
- użytkownik, który już eksploatuje systemy informatyczne na istniejącym a nietypowym dla systemu sprzęcie — stanie przed koniecznością ich przeprogramowania
- konieczność szerokiej standaryzacji przy budowie sieci. Użytkownicy bardziej doświadczenia będą przy uzgodnieniach standaryzacyjnych forsowali rozwiązania stosowane w ich dotychczasowych systemach
- konieczność przyjęcia jednolitego rozwiązania. Wielka dyscyplina formalna związana z przestrzeganiem zaleceń standaryzacyjnych może ujemnie wpływać na merytoryczną stronę rozwiązań

Mgr inż. JERZY DAŃDA jest absolwentem Wydziału Łączności Politechniki Warszawskiej (1958 r.). Pracował w Grupie, a następnie Zakładzie Aparatów Matematycznych (1956 r.), gdzie m. in. był współkonstruktorem maszyny cyfrowej XYZ. Obecnie pracuje w Instytucie Maszyn Matematycznych, gdzie kieruje Pracownią Prognoz w Zakładzie Studiów i Analiz. Jest redaktorem naczelnym dwumiesięcznika ETO NOWOŚCI i autorem wielu publikacji.

1) Advanced Research Project Agency — USA

- realizacja sieci jednorodnej, a szczególnie wykonanie jej oprogramowania wymaga stworzenia i operatywnego kierowania bardzo dużą grupą dobrych programistów, co jak wiadomo jest zadaniem szczególnie trudnym
- w przypadku hierarchicznego systemu kierowania w oparciu o sieć jednorodną można praktycznie wykorzystać oprogramowanie tylko jednego producenta, a jak wiadomo żadna firma na świecie nie dysponuje obecnie najlepszymi rozwiązaniami w zakresie wszystkich składników oprogramowania.

Wymienione powyżej niedogodności sieci jednorodnych mogą jednocześnie posłużyć jako wykaz zalet sieci niejednorodnych. Nie będziemy zajmowali się wykazywaniem istniejących może jeszcze dodatkowych zalet sieci niejednorodnych.

Nie będziemy też starali się wykazywać ich wad, które są oczywiste. Już sama niejednorodność musi kosztować i może kosztować wiele, szczególnie w zakresie konserwacji sprzętu, trudniej też o przekazywanie nagromadzonych doświadczeń. Zajmiemy się natomiast zestawieniem warunków, jakie muszą być spełnione i prac, jakie muszą być wykonane, aby niejednorodność sieci mogła przynieść jej użytkownikom więcej korzyści, niż niedogodności.

## Podstawowy problem niejednorodnych sieci komputerowych

Większość zalet sieci niejednorodnych wynika stąd, że twórca takiej sieci w żaden sposób nie może uniknąć konieczności rozwiązania podstawowego problemu — zapewnienia sprawnej i efektywnej współpracy maszyn, które różnią się od siebie w istotny sposób. Na przykład uważa się, że standaryzacja języków programowania rozwiązuje problem przenoszenia oprogramowania użytkowego z jednej instalacji na drugą. W dużym zakresie jest to prawdziwe np. w odniesieniu do FORTRANu IV, który ma zestandaryzowane procedury, ale nie jest prawdziwe w odniesieniu do ALGOLu, w którym procedury te są niestandaryzowane.

Nie tylko zresztą brak standaryzacji procedur ogranicza przenoszenie programów. Nie ma żadnej standaryzacji języków sterowania zadaniami *(job control languages)* oraz metod dostępów do zbiorów, a zaczątki takiej standaryzacji widzi się tylko w ramach oprogramowania firmowego producentów sprzętu. Wszystko to powoduje, że autor większego programu jest praktycznie związany z instalacją, na której program swój uruchamiał.

Przy budowie sieci niejednorodnych problem przenoszenia znacznej części programów użytkowych musi być rozwiązany, gdy zakłada się zwykle przy tworzeniu takich sieci, że program będzie pracował w komputerze A, korzystając z danych, przechowywanych np. w komputerach B, C i A. Każdy z nich będzie z pewnością miał inny sposób dostępu do danych i inny język sterowania zadaniami. Odpowiednia konwersja, nie tylko formatów, ale również całych sekwencji rozkazowych będzie musiała być wykonywana automatycznie, jeśli sieć ma funkcjonować sprawnie. Zadanie takie, po rozpoznaniu problemu, postawiła przed sobą grupa INDRA [2], dodatkowo je komplikując przyjęciem założenia szczególnie korzystnego z punktu widzenia użytkownika, że wymiana informacji między maszynami musi być możliwa bez jakichkolwiek zmian w systemach operacyjnych współpracujących komputerów. Realizacji tego celu służy system zwany *„store and forward"* (przechowaj i popchnij naprzód), który polega na wprowadzeniu między dwie współpracujące ze sobą maszyny odpowiednio oprogramowanego minikomputera pośredniczącego [1]. W systemie doświadczalnym grupy INDRA był to komputer PDP-9, a maszynami współpracującymi za jego pośrednictwem były CDC 6600 oraz IBM 360/195. Nie wchodząc w szczegóły rozwiązania powiemy, że jednym z istotnych składników oprogramowania maszyny pośredniczącej był analizator leksykalny, pozwalający na rozpoznawanie ciągów znaków wprowadzanych do systemu *„store and forward"*, które było podstawą do wygenerowania odpowiedniego ciągu znaków, wysyłanego do maszyn współpracujących [2]. Przyjęcie zasady, że praca analizatora leksykalnego była określana przez podanie gramatyki ciągu wejściowego, zapewniało odpowiednią elastyczność systemu w tym przypadku dla wykorzystywania znacznej części oprogramowania systemu *„store and forward"* dla współpracy różnych par maszyn.

Oczywiście, nawet rozwiązanie problemu uniwersalnej metody komunikowania się dwóch dowolnych maszyn ze sobą jest tylko rozwiązaniem podstawowego problemu oprogramowania sieci niejednorodnych maszyn.

Aby wyliczyć inne problemy podobnej natury musimy wpierw spróbować określić typy zadań, jakie mogą występować wtedy, gdy sieć niejednorodna służyłaby jako baza techniczna dla realizacji hierachicznego systemu zarządzania gospodarką narodową.

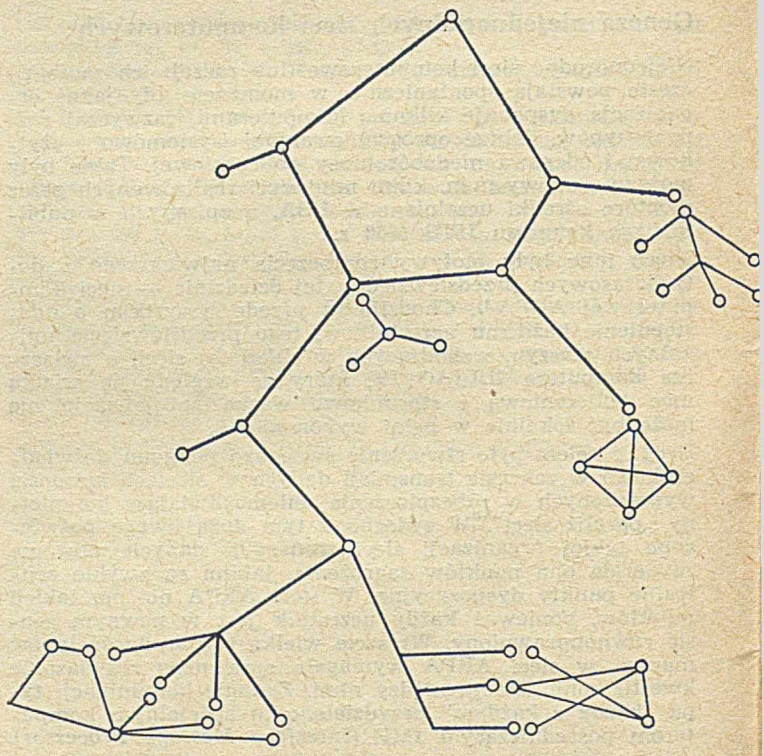
## Typy zadań w sieci komputerowej do zarządzania gospodarką narodową

Dokonajmy prostego wyliczenia typów zadań obliczeniowych, jakie mogą występować w sieci komputerowej, przeznaczonej do zarządzania gospodarką.

Przyjmiemy przy tym założenie, że nawet zastosowania (typy zadań), które na obecnym etapie rozwoju informatyki w kraju są w ewidentny sposób niewykonalne z przyczyn technicznych bądź organizacyjnych — są godne szczegółowego rozpatrzenia.

Drugim założeniem, tym razem o charakterze technicznym, jest ustalenie, że sieć ma strukturę hierarchiczną. Na tym etapie rozważań przyjmujemy, że istnieją dwa poziomy tej hierarchii. Występuje mianowicie zbiór sieci komputerowych, przy czym w każdej z tych sieci mogą występować komputery różnych typów, a więc są to sieci niejednorodne. Sieci te komunikują się ze sobą przez sieć nadrzędną — nazwiemy ją międzykomputerową siecią lub intersiecią. Intersieć jest siecią jednorodną, przy czym można dodać, że typ komputera, który będzie zastosowany w intersieci, nie musi być identyczny z komputerami stosowanymi w innych, podsieciach lokalnych, że będzie to minikomputer o średniej złożoności i że będzie to najprawdopodobniej minikomputer odpowiednio wyspecjalizowany. Informacje przesyłane od komputera jednej sieci lokalnej do komputera w innej sieci lokalnej przechodzą przez dwa komputery intersieci.

Ogólny schemat intersieci i sieci składowych przedstawia rys. 1. Typy zadań — o których była mowa w tytule określimy w sposób pośredni przez podanie funkcji intersieci, jakie będą musiały być przez nią wykonywane dla każdego typu zadania.



Rys. 1. Uproszczona struktura intersieci i niejednorodnych sieci komputerowych (lokalnych)

2) „Institute Display and Remote Access" (INDRA) — akronim przedsięwzięcia wykonywanego w Institute of Computer Science of University of London, pod kierunkiem prof. P. T. Kirsteina. Autor artykułu pracował w tej grupie w okresie półrocza 1971/1972, przed satelitarnym podłączeniem sieci INDRA do sieci ARPA w Stanach Zjednoczonych.

## Przykładowe funkcje intersieci

**Zadanie 1:** *ogólnosieciowa rejestracja stanu zasobów (np. inwentaryzacja pracowników określonej dziedziny, inwentaryzacja określonych zasobów materiałowych podlegających reglamentacji), inwentaryzacja określonych zapotrzebowań (np. na materiały trudnodostępne) i inne.*

**Komentarz:** Zazwyczaj przy wykonywaniu takiego zadania będzie podawana do sieci, jako informacja wejściowa, lista przedmiotów, o jakie się zapytujemy przy równoczesnym podaniu warunków, jakie przy wyszukiwaniu informacji mają być spełnione. Warunki mogą być podane w postaci złożonej funkcji logicznej, jak to zazwyczaj występuje w systemach wyszukiwania informacji, np. bibliograficznej. Intersieć musi „wydać" odpowiednie polecenia do podległych jej sieci komputerowych, przyjąć odpowiedzi i odpowiednio je scalić z ewentualnym (na życzenie użytkownika) wskazaniem, skąd dane pochodzą.

Niektóre funkcje intersieci przy realizacji tego zadania:

- transformacja identyfikatorów materiałowych stosowanych w intersieci na identyfikatory stosowane w podległych sieciach (w obu kierunkach przepływu informacji)
- generacja sekwencji rozkazowych służących ekstrakcji informacji z banków danych, znajdujących się w poszczególnych sieciach
- przechowywanie (czasowe) informacji o źródłach dopływu konkretnych danych
- dodanie informacji komentującej, np. o sposobie dostępu do danych
- określenie stopnia wiarygodności danych; funkcja ta jest szczególnie trudna do realizacji maszynowej — pomocne mogą się prawdopodobnie okazać tzw. techniki rozmytych zbiorów *(fuzzy-set)*
- transformacja formatów danych i ujednolicenie jednostek miar.

**Zadanie 2:** *ogólnosieciowa rejestracja przebiegu pewnych funkcji, np. przewidywanego lub faktycznego czasu zmiany zapotrzebowania na jakiś produkt.*

**Komentarz:** zadanie to jest podobne do poprzedniego. Istotną różnicą jest znaczne zwiększenie strumienia informacji, jakie muszą być przekazywane przez intersieci. W związku z tym może zachodzić potrzeba dokonywania uprzedniej agregacji danych bądź w podległych sieciach, bądź też przez minikomputery intersieci zlokalizowane blisko miejsca, skąd dane te pochodzą (ograniczenie obciążenia sieci transmisyjnej). Taka agregacja będzie nakładała dodatkowe funkcje na intersieć. Uzupełnimy więc wyżej podaną listę funkcji następującymi pozycjami:

- określenie „gęstości próbkowania" funkcji źródłowych przeprowadzonego w jednolity sposób dla wszystkich podległych sieci (stosowanie odpowiednich metod interpolacyjnych przy agregowaniu funkcji w intersieci)
- przekształcanie danych tabelarycznych przebiegu danych funkcji (jeśli w takiej postaci są przechowywane w którymś z zapytywanych banków danych) do postaci funkcyjnych, przyjętych za standard w intersieci, np. przez odpowiednie postępowanie aproksymacyjne
- generacja komentarzy, np. gdy z niektórych źródeł można uzyskać tylko niepełne informacje niewystarczające do określenia funkcji, o którą intersieć zapytuje.

**Zadanie 3:** *centralne „uruchamianie" modeli symulacyjnych funkcjonowania określonych branż gospodarki, których banki danych znajdują się w poszczególnych sieciach.*

**Komentarz:** zadanie to, niewątpliwie niezwykle skomplikowane technicznie, będzie jednak, jak się wydaje, dominowało przy wykorzystywaniu sieci komputerowych do zarządzania gospodarką. Stosowanie metod symulacyjnych, jest zakładane jako etap docelowy prawie wszystkich zintegrowanych systemów przetwarzania informacji dla potrzeb kierownictwa (IMIS). Równocześnie wiadomo, że sporządzenie względnie adekwatnych modeli symulacyjnych, nawet niewielkich przedsiębiorstw, nastręcza bardzo poważne trudności teoretyczne i praktyczne. Wiadomo również, że model nadmiernie uproszczony dostarcza zazwyczaj wyników albo zbyt odbiegających od rzeczywistości, lub też wartość prognostyczna otrzymywanych wyników jest niewielka, gdyż dotyczy banalnie oczywistych charakterystyk modelowanego zjawiska.

Równocześnie jednak, to samo zadanie nie wydaje się tak realizacyjnie odległe, jeśli przyjmiemy, że model zostanie wykonany jako „zlepek" modeli symulacyjnych, które będą projektowane przez zespoły fachowców, nawet nie komunikujących się ze sobą. Założymy przy tym, że projektanci tych modeli symulacyjnych będą mieli wystarczającą wiedzę o stosowaniu metod symulacji maszynowej (ciągłej i dyskretnej) do zagadnień ekonomiczno-organizacyjno-socjologicznych oraz, że będą doskonałymi fachowcami w tej gałęzi przemysłu, czy organizacji, której model symulacyjny będą konstruowali. W ten sposób powstanie zespół modeli symulacyjnych, być może nawet w niektórych względach niespójny pod względem przyjętych zasad ich budowy, lecz za to znacznie bardziej odpowiadający modelowanej rzeczywistości. Owe niespójności modelu nie powinny nas ani dziwić, ani zniechęcać. Wszak rzeczywistość nie jest idealnie spójna.

Oczywiście, taki sposób dochodzenia do modelu ogarniającego całą gospodarkę narodową będzie nakładał na funkcjonowanie intersieci szczególnie trudne zadania, nie wystarczy bowiem, że będzie ona realizowała funkcje „czysto techniczne". Do funkcji intersieci przy takim zadaniu będzie należało również usuwanie skutków niejednorodności poszczególnych modeli symulacyjnych.

Przy tym zadaniu wystąpią m.in. następujące funkcje w intersieci.

- inicjowanie pracy poszczególnych modeli symulacyjnych (zadawanie wartości początkowych, określenie skali czasowej symulacji)
- synchronizowanie przebiegów w poszczególnych sieciach, zapewniających warunki przekazywania informacji między poszczególnymi sieciami
- transformacja formatów oraz ujednolicanie jednostek miar przy przekazywaniu informacji między poszczególnymi modelami symulacyjnymi
- „obserwowanie" przebiegu symulacji i generowanie komunikatów o przebiegu symulacji dla użytkownika intersieci
- przekazywanie do podległych modeli symulacyjnych interwencyjnych poleceń użytkownika intersieci, zmieniających generalnie lub w szczegółach seans symulacyjny.

**Zadanie 4:** *organizowanie i zapewnienie środków technicznych dla gier kierowniczych między użytkownikami poszczególnych sieci lokalnych, w których znajdują się modele symulujące działalność poszczególnych branż.*

**Komentarz 1:** Gry kierownicze są już obecnie na świecie powszechną praktyką. Łatwo sobie wyobrazić, że korzyści dydaktyczne z przeprowadzania takich gier mogą znakomicie powiększyć się, gdy decyzje podejmowane w ramach gry będą oparte o informacje, pochodzące ze sfery działania bliskiej jej uczestnikowi.

Posiadanie modelu symulacyjnego danej sfery działania umożliwia dostęp do takich właśnie danych, a równocześnie umożliwia eksperymentalne sprawdzanie bez wywoływania ewentualnych szkód, skutków najbardziej śmiałych i nowatorskich decyzji. Parametry modelu symulacyjnego mogą być w razie potrzeby również zmieniane tak, aby doprowadzić do działania modelu w sposób najbardziej przydatny pod względem dydaktycznym.

Intersieć może w takim przedsięwzięciu pełnić nie tylko funkcje usługowe, polegające na niezbędnych transformacjach informacji przekazywanych miedzy uczestnikami gry, ale również może przebieg tej gry rejestrować i wykrywać pewne ogólne prawidłowości, niemożliwe do zaobserwowania przez człowieka — a wreszcie odpowiednio przetworzone informacje przekazywać do wykorzystania przez bezpośredniego użytkownika lub „zwierzchnika" intersieci. W ten sposób ten typ zastosowań intersieci mógłby służyć nie tylko celom dydaktycznym. Sądzę, że wyobraźnia Czytelnika pozwoli na własne uzupełnienie obrazu wszystkich potencjalnych korzyści przeprowadzania takich gier, jeśli przyjmie założenie, że trudności techniczne takiego przedsięwzięcia są do przezwyciężenia.

**Komentarz 2:** z technicznego punktu widzenia zadanie powyżej opisane narzuca na funkcje intersieci wymagania, które ogólnie można scharakteryzować jak następuje. O ile przy poprzednich zadaniach wszystkie działania w intersieci były nastawione na wykonanie poleceń użytkownika wyróżnionego, mającego charakter jej centralnego dysponenta, o tyle w tym zadaniu sieć będzie w bardziej „demokratyczny" sposób traktować każdego z potencjalnych użytkowników. W tym sensie jej użytkownikiem jest każda sieć lokalna dołączona do intersieci. Dalsze zadania sie-

ci będą związane właśnie z takim użytkownikiem „demokratycznym". Generalizując zagadnienia, wydaje się, że jednym z warunków sprawnego działania intersieci będzie możliwość równoczesnego przeprowadzania w niej wielu złożonych działań, noszących charakter przetwarzania i przesyłania informacji. W braku lepszego określenia użyjemy sformułowania „wieloaktywność" zbliżonego do „wieloprogramowości". Nowe słowo wydaje się potrzebne, gdyż „wieloprogramowość" — o ustalonym znaczeniu — nie pociąga za sobą tak złożonych i bogatych funkcji komunikacji, jak będą występowały w intersieci wykorzystywanej równocześnie do wykonywania funkcji „centralistycznych" i „demokratycznych".

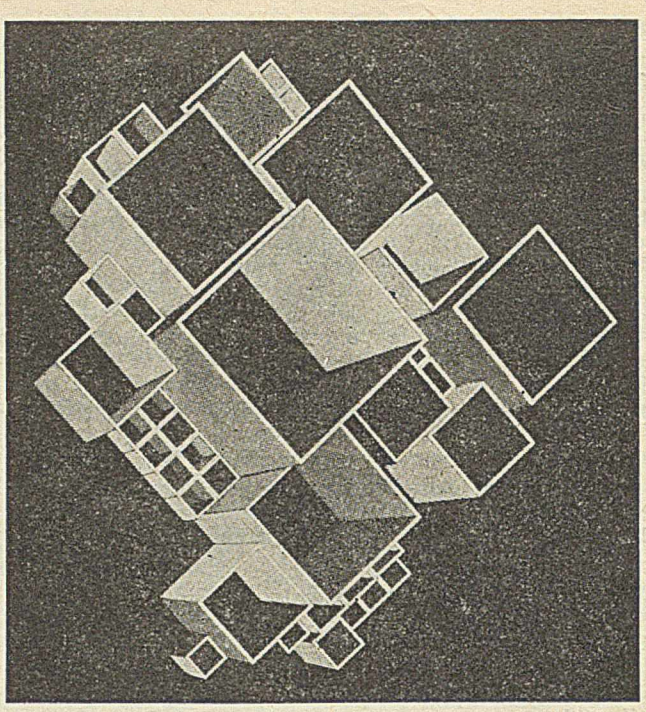
Omówmy teraz te funkcje sieci, które są nowe:

- alokacja zasobów intersieci w celu zapewnienia jej „wieloaktywności"
- obserwacja stopnia wykorzystywania sieci (tworzenie się „wąskich gardeł", nie pełne wykorzystywanie alokowanych zasobów itp. nieprawidłowości), rejestracja tych faktów dla podejmowania środków doraźnego działania i przygotowania statystycznych informacji dla długofalowych działań (np. decyzje o rozbudowie sieci, likwidowaniu pewnych jej części, ograniczaniu pewnych funkcji)
- realokacja zasobów, na podstawie informacji uzyskanych przez realizowanie wyżej omówionej funkcji „autoobserwacji działania"
- obserwacja działań przebiegających w intersieci i w podległych jej sieciach lokalnych, uogólnianie tych obserwacji zgodne z życzeniami dysponenta intersieci i przekazywanie mu tych generalnych obserwacji. Obserwacja pewnych szczególnych zdarzeń, wprowadzanie zmian do zachowania się intersieci, zarządzonych przez dysponenta dla celów eksperymentalnych i celowo nie ujawnianych jej użytkownikom (np. w trakcie gry polegające np. na na celowym i ściśle określonym fałszowaniu (przekłamywaniu) danych przekazywanych między „graczami")
- przekazywanie informacji potrzebnych do sprawnego prowadzenia gier, a luźno tylko związanych z zasadniczymi funkcjami sieci.

**Zadanie 5:** *usługi polegające na transmisji danych między poszczególnymi sieciami na zamówienie dysponentów sieci lokalnych, wykonywane okresowo lub natychmiast.*

**Przykład:** lokalna sieć zajmująca się m.in. prognozowaniem zbiorów rolniczych może zażądać od intersieci okresowego przekazywania w odpowiedniej formie prognoz pogody z obserwacji meteorologicznych z innej lokalnej sieci, np. Państwowego Instytutu Hydrologiczno-Meteorologicznego.
Przy realizacji tego zadania występują m.in. następujące funkcje intersieci:

- rejestrowanie życzeń wszystkich użytkowników (np. okresowe transmitowanie i przekształcanie komunikatów)
- planowanie aktywności intersieci (optymalizacja jej wykorzystania)
- okresowe zapytywanie podległych sieci w celu uzyskania informacji zapotrzebowanych przez inną sieć lokalną
- pomoc dla odbiorcy okresowych informacji w sytuacjach wyjątkowych.



Niewątpliwie można byłoby próbować obreślać dalsze zadania dla takiej sieci, np. określać zadania wynikające z przyjętej taktyki centralnego dysponenta polegającej na udzielaniu podległym sieciom szczegółowych lub generalnych dyrektyw. Niewątpliwie w każdej sytuacji gospodarczej takie zadania będą występowały. Nie będziemy ich jednak szczegółowo omawiać, gdyż, jak się wydaje, nie narzucają one na sposób funkcjonowania intersieci żadnych szczególnych wymagań technicznych poza tymi, które występowały przy centralnie obserwowanych i zwykle nieznacznie modyfikowanych przez centralnego dysponenta sieci grach konkurencyjnych.

**Podziękowanie**

Pierwowzór tego artykułu (znacznie szczegółowszy) pt.: „Niejednorodne sieci komputerowe i rozłożone banki danych" został napisany w maju 1973 r. w ramach współpracy autora ze Zjednoczeniem Informatyki.
Autor wyraża podziękowania mgr Władysławowi Matwinowi za zachętę do opublikowania tego tekstu oraz wiele cennych uwag przekazanych w licznych dyskusjach.

LITERATURA

[1] P. T. Kirstein: A. Raport on INDRA Actities in Remote Computing and Communications Networks, University of London, Institute of Computer Science, IOSP 122, March 1972
[2] Dańda J.: SAG-BOPL-Program INDRA Note No. 246 SDI 40, University of London, Institute of Computer Science, April 1972

## Sprostowanie do nr 2/75 INFORMATYKI

Do reportażu z SOETO, zamieszczonym, w nr 2/75 INFORMATYKI (str. 30 — 33), do podpisu na str. 31 wkradł się przykry błąd, zmieniający kolejność przedstawionych na zdjęciu osób. Prawidłowa kolejność: od lewej wicedyrektor mgr Marek ANTOŃCZYK, dyrektor mgr inż. Marek ŻELAWSKI, główny technolog inż. Aleksander BĄK.
Redakcja przeprasza Czytelników i zainteresowanych.

JAROSŁAW DOBOSZ
Instytut Informacji Naukowej,
Technicznej i Ekonomicznej
Warszawa

681.322.06.004.14:002.513.5

# System informacyjny FAMULUS

W artykule przedstawiono opis pakietu programów stosowanych w systemie CYFRONET, służących do przetwarzania i wyszukiwania informacji tekstowych.

W artykule przedstawiono opis pakietu programów, zaadaptowanego do konfiguracji S.A. CYFRONET. Omawiany system jest podzbiorem pakietów o tej samej nazwie, przeznaczonych do eksploatacji na maszynach CDC 6000, IBM 360 i 370, UNIVAC 1100.

FAMULUS jest pakietem programów przetwarzania i wyszukiwania informacji tekstowej. Jego pierwotnym przeznaczeniem było dostarczenie pracownikom nauki narzędzia wspomagającego indywidualne prace dokumentacyjne. Ze względu na swe cechy użytkowe, programy systemu znajdują jednak znacznie szersze zastosowania. Mogą być one np. wykorzystywane w niewielkich systemach bibliotecznych.

Są one szczególnie przydatne przy obróbce niezbyt licznych zbiorów (rzędu kilku tysięcy zapisów (rekordów).
System umożliwia wsadowe wykonywanie następujących operacji:

— zakładanie i modyfikacje bazy danych
— wydruki wybranych części założonego zbioru
— tworzenie słownika lub indeksu
— sortowanie i scalanie zbiorów
— wyszukiwanie w zbiorze zapisów o zawartościach odpowiadających pytaniom.

Programy są proste i elastyczne w sterowaniu, mają łatwą do przygotowywania i przejrzystą postać danych wejściowych. System ma rozbudowane możliwości wydawnicze. W połączeniu z dużą efektywnością działania stanowi to o jego wysokich walorach użytkowych.

## Baza danych

Podstawą działania wszystkich programów systemu jest zbiór o organizacji sekwencyjnej, zawierający zakodowane obrazy dokumentów. Pierwszy zapis zbioru przechowuje informacje strukturalne. W następnych umieszczane są skondensowane obrazy dokumentów wprowadzanych do bazy danych.

W każdym zapisie pamiętany jest jeden dokument, zapisy zbioru są zmiennej długości — w zależności od rzeczywistych rozmiarów wprowadzonego dokumentu, i mogą zawierać do 10 pól. Liczba i kolejność pól definiowane są przy zakładaniu bazy danych. Poszczególne pola identyfikowane są przez czteroznakowe etykiety. Długości pól są dowolne i mogą być dowolnie zmieniane przy modyfikacjach bazy danych. Jedynym obowiązującym ograniczeniem jest dopuszczalna sumaryczna wielkość zapisu 4000 znaków.

Dopuszczalna liczba zapisów w zbiorze nie jest w zasadzie ograniczona. Ze względu na strukturę bazy danych przetwarzanie dużych zbiorów może być jednak nieopłacalne czasowo. Dlatego też dla zbiorów zawierających ponad 3000—5000 zapisów (zależnie od ich średniej długości) zalecane jest wykonywanie operacji na częściach zbioru i ewentualne scalanie zbiorów wynikowych.

Zawartości pól traktowane są wyłącznie jako ciągi znaków. Przy zakładaniu lub modyfikacjach bazy danych, sortowaniu, scalaniu wydrukach wszystkie znaki w tekstach pól są równouprawnione. W programach: wyszukiwawczym, tworzenia indeksu i słownika, z tekstu pól wyodrębniane są poszczególne słowa lub frazy. Ciągi składające się z liter, cyfr i myślników traktowane są jako całości. Ogranicznikami odcinków tekstu są więc znaki specjalne i spacje. Jedno z pól może zostać zadeklarowane jako tzw. pole deskryptorowe. W obrębie tego pola jedynym ogranicznikiem tekstu jest przecinek. Umożliwia to traktowanie również dłuższych, wielowyrazowych fraz jako niepodzielnych całości.

## Programy

W skład omawianej wersji systemu wchodzi siedem programów napisanych w języku FORTRAN EXTENDED.

**Program EDIT** służy do zakładania i modyfikacji bazy danych. Przy zakładaniu bazy danych program ten czyta dokumenty przygotowane na kartach i zapisuje ich zakodowane obrazy na nośniku magnetycznym (w kolejności wczytywania). Wprowadzane zapisy są drukowane z wydzielaniem poszczególnych pól.

W przebiegu modyfikacyjnym program EDIT kopiuje wybrane części zbioru wejściowego na zbiór wyjściowy i dopisuje na końcu wprowadzane z kart nowe dokumenty. Wybieranie ze zbioru wejściowego można organizować przez wskazanie zapisów do skopiowania lub przez wyznaczenie tych, które mają być usunięte. W kopiowanych dokumentach mogą być wymieniane dowolne ciągi znaków w tekstach pól. Podczas przebiegu drukowane są zapisy o zmienianej zawartości, wprowadzane z kart oraz wskazane specjalną kartą sterującą.

Szerokość wydruku, tak jak zresztą i w pozostałych programach może być regulowana.

**Program GALLEY** drukuje zawartości wybranych zapisów zbioru. W zależności od użytych kart sterujących, drukowane mogą być zarówno całe dokumenty, jak i tylko wskazane pola (w zadanej kolejności). Dostępne są dwa formaty wydruku: z wydzielaniem poszczególnych pól i zaznaczaniem ich nazw oraz bez wydzielania pól.

**Program INDEX** tworzy indeks alfabetyczny. Drukowane są wszystkie słowa i frazy występujące w polach deskryptorowych określonej części zbioru. Z każdym deskryptorem skojarzona jest lista numerów zapisów, w których on występował.

**Program SORT** porządkuje zapisy zbioru w kolejności alfabetycznej według zawartości 1 do 10 pól. Porządkowanie wykonywane jest przez sortowanie wewnętrzne (metoda Shella) grup zapisów, a następnie wielokrotne dwustrumieniowe scalanie.

**Program MERGE** scala dwa zbiory posortowane w identyczny sposób przez program SORT. Scalony zbiór może być drukowany.

Mgr inż. JAROSŁAW DOBOSZ ukończył z wyróżnieniem Wydział Elektroniki Politechniki Warszawskiej, specjalność: maszyny matematyczne (1973). Za pracę dyplomową pt. „Translator interpretacyjny języka JOSK" otrzymał nagrodę II stopnia w X Ogólnopolskim Konkursie „Młoda myśl dla kraju".
Pracuje w Centralnym Ośrodku Informatyki Politechniki Warszawskiej jako programista (1973). Zajmuje się problemami translacji i systemami wyszukiwania informacji.

**Program VOCAB** drukuje alfabetyczne zestawienie wszystkich słów występujących w zadanych polach wybranych do przetwarzania zapisów. W celu wyodrębniania ze słownika słów trywialnych[1]), podaje się ich listę jako dane wejściowe dla programu.

W zestawieniu drukowane są one oddzielnie.

**Program SEARCH** jest programem wyszukiwawczym systemu FAMULUS. Formuły wyszukiwawcze konstruowane są w postaci wyrażeń boolowskich za pomocą operatorów OR, AND, NOT i nawiasów. Dla każdego pytania przeglądane są od początku zawartości wskazanych pól w zapisach wybranej do przeszukiwania części zbioru. Dla zbadanego zapisu wyznaczana jest wartość formuły wyszukiwawczej. Informacje o zapisach spełniających warunki pytania drukowane są w wybranym przez użytkownika formacie. Za pomocą kart sterujących można żądać wg druku wybranych pól w zadanej kolejności, wydruku z wydzielaniem poszczególnych pól lub bez wydruku jednej linii tekstu oraz wydruku jedynie numerów wyszukanych zapisów. Odpowiedzi na jedno z pytań mogą być przekopiowane na zbiór wyjściowy.

## Sterowanie

Sterowanie działaniem programów systemu odbywa się za pomocą kart sterujących systemu operacyjnego i kart sterujących FAMULUSa.

Za pomocą kart sterujących systemu operacyjnego wykonywane są:

— udostępnianie programów w systemie
— ładowanie i inicjowanie wykonania programów
— określanie systemowych nazw i statusu zbiorów, na których działają wywoływane programy.

Karty sterujące programów pakietu FAMULUS służą natomiast do:

— identyfikacji zbiorów i określania struktury zapisów
— wybierania części zbiorów do przetwarzania
— sterowania zakresem i formatem wydruków
— wprowadzania zmian i poprawek w pamiętanych tekstach dokumentów
— inicjalizacji słownika
— sterowania tworzeniem zbiorów pochodnych
— zadawania pytań dla programu wyszukiwawczego.

Wszystkie karty sterujące mają zunifikowany, czytelny i łatwy w zapisie format:
(dyrektywa) lub (dyrektywa)parametry.

W razie potrzeby parametry kontynuowane są na kolejnych kartach. Bezparametrowe karty sygnalizują programom określony stan lub wprowadzenie opcji, np.:

(PRINT BY FIELDS) — żądanie wydruku zapisów z wydzielaniem pól,

(ORIGINAL) — oznaczenie przebiegu zakładania bazy danych.

1) słowa, które nie są potrzebne określonemu użytkownikowi

Parametrami kart sterujących mogą być: ciąg znaków, etykieta pola lub lista etykiet, liczba lub lista liczb.
Np.

(ID)FILE — identyfikacja zbioru FILE,
(FIELD CHOICE (AUTH, TITL, ABST) — wybór pól do druku,
(WIDTH/ (65) — ustawienie szerokości wydruków,
(SELECT/(1—500, 600—700, 813) — wybranie do przetwarzania zapisów o numerach z podanych zakresów.

Jedynie trzy karty sterujące mają parametry zapisywane w innej postaci.

Przy inicjalizacji słownika podawana jest lista słów, np.:
/VOCABULARY/ COMPUTER, ON*, OFF*, SYSTEM, THE*
(gwiazdki wskazują słowa trywialne).

Karta sterująca wymianami zawartości pól zawiera listę opisów wymian, np.:

/REPLACE/(10) /AUTH/ * JOHN * JAN (11) (DAT)*1965* 1963* — wymiana JOHN na JAN w polu AUTH zapisu nr 10 i 1965 na 1963 w polu DAT zapisu numer 11.

W pytaniach dla programu SEARCH wpisywane są wyrażenia boolowskie tworzące formułę wyszukiwawczą.

Prosty i wygodny jest również format zapisu na kartach dokumentów wprowadzanych do bazy danych:

w pierwszej karcie kolejnego pola w kolumnach 1—4 umieszczana jest etykieta pola. Treść wpisuje się w kolumnach 6—80 tej i ewentualnie następnych kart. Poszczególne dokumenty oddzielane są pustymi kartami.
np.

```
AUTH JAN KOWALSKI
TITL PRZYGOTOWYWANIE DANYCH
ABST TO JEST
     PRZYKŁAD ZAPISU DOKUMENTU
     NA KARTACH
DATA 1975
```

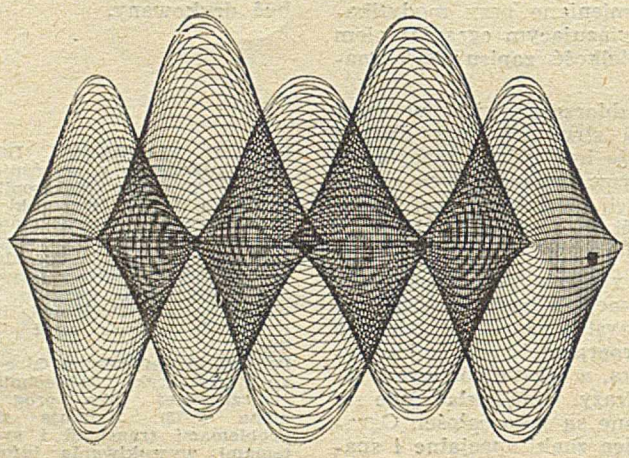
Programy pakietu drukują szereg informacji opisujących przebieg przetwarzania i stanowiących przejrzystą dokumentację wykonywanych działań.
Podawane są m.in.:

— obrazy kart sterujących
— opisy otwieranych zbiorów
— dane ilościowe dotyczące przetwarzania.

Informacje te drukowane są oddzielnie od wydruków dokumentów. Pozwala to na otrzymywanie wydruków nadających się do umieszczenia w raportach, publikacjach itp.

Bogata diagnostyka umożliwia szybkie odnalezienie i poprawienie ewentualnych błędów w kartach sterujących bądź danych.

Dzięki prostocie sterowania, łatwości przygotowywania danych i czytelnym komunikatom korzystanie z systemu nie wymaga głębszej wiedzy informatycznej. Rozbudowane wydawnictwo ułatwia otrzymywanie przejrzystych w formie wydruków dobranych do indywidualnych potrzeb czy przyzwyczajeń użytkownika.



JERZY R. JAWOROWSKI
RYSZARD TADEUSIEWICZ
Akademia Górniczo-Hutnicza
Kraków

831.322.06:534.4

# ART 73B — Język do przetwarzania informacji akustycznych

Artykuł jest opisem wyników prac i uzyskanych rezultatów przy próbie stworzenia języka służącego do przetwarzania informacji akustycznej przez komputer. Omówiono najważniejsze procedury języków grupy ART.

Wiele badań akustycznych opiera się na przetwarzaniu danych, np. takich, jak określenie parametrów statystycznych danych lub czasowych zmian wartości formatów czy momentów widma. Badania te prowadzone są zwykle za pomocą złożonej i precyzyjnej aparatury w swojej części pomiarowej, natomiast samo przetwarzanie uzyskanych danych odbywa się w sposób dość prymitywny: drogą ręcznego wykonywania obliczeń lub w najlepszym przypadku drogą ręcznego przenoszenia danych pomiarowych na nośniki informacji, akceptowane przez dostępną maszynę cyfrową i przetwarzania danych przez tę maszynę.

Nieliczne jeszcze w Polsce zestawy aparatury do badań akustycznych wyposażone w minikomputery nie stanowią rozwiązania zadowalającego i wystarczającego do zaspokojenia wszystkich potrzeb [8].

W tej sytuacji za celowe należy uznać próby, zmierzające do wykorzystania w badaniach akustycznych łatwo dostępnych, uniwersalnych maszyn cyfrowych. Praca niniejsza stanowi opis rezultatów, uzyskanych przy próbie stworzenia języka służącego do przetwarzania informacji akustycznej przez maszynę cyfrową. Język ten, a właściwie cała grupa języków, o wspólnej nazwie ART [1]) nie stanowi jeszcze w chwili obecnej tworu ostatecznego, gdyż autorzy dążą do stałej poprawy łatwości programowania w języku ART, uniwersalności języka i zapokajania nowych, pojawiających się w praktyce badań akustycznych potrzeb.

Języki grupy ART przystosowane są do programowania maszyn cyfrowych, do których pamięci wprowadzono sygnał dźwiękowy zakodowany w odpowiedniej postaci. Wprowadzanie zakodowanego sygnału dźwiękowego do pamięci maszyny cyfrowej odbywa się za pomocą specjalnego konwertera analogowo-cyfrowego KART-1, przetwarzającego sygnał akustyczny z mikrofilmu lub magnetofonu na zbiór symboli, przesyłanych do pamięci maszyny cyfrowej [7], [8]. Język ART 73 umożliwia użytkownikowi dynamiczną rezerwację odcinków pamięci dla przechowywania obrazów akustycznych będących (z punktu widzenia maszyny cyfrowej) wektorami liczb całkowitych o zmiennej górnej granicy. W języku ART 73 B przyjęto z niewielkimi zmianami strukturę dynamicznej pamięci podobnie jak w systemach ICL 1900 COMPUTER AIDED DESIGN [1], SYMBAL [2] czy MINIMAL/SET 1204 [5].

Opis systemu dynamicznej rezerwacji pamięci, stosowanego w języku ART 73B podano w pracy [8] oraz częściowo w [3].

Wszystkie procedury języka ART 73B są fortranowskimi procedurami typu INTEGER. Tego samego typu są wszystkie parametry procedur. Niżej omówimy 6 grup procedur języka ART 73B, stanowiących jego zasadniczy trzon. Pełny zestaw i obszerniejszy opis procedur języka ART 73B zamieszczono w pracy [8].

## Procedury zarządzające

Należą do nich procedury, które administrują dynamicznym przydziałem pamięci oraz procedury uwalniające użytkownika od konieczności ustawiania początkowych wartości roboczych komórek języka. Są nimi:

1. ARTLEADER: procedura odpowiedzialna za dynamiczną rezerwację pamięci i ustawienie danych początkowych.
2. SWEPER (SPACE): procedura dokonująca odzyskiwania nieużywanych obszarów pamięci.

## Procedury pomocnicze

Ich zadaniem jest realizacja typowych procedur matematycznych, używanych w języku ART oraz lokalna obsługa i wykrywanie błędów w programie, napisanym w języku ART. Procedury pomocnicze mogą być wykonywane samodzielnie, zasadnicza jednak ich rola polega na tym, że stanowią one części składowe procedur wyższego rzędu.

3. KFILTR (I,M): Wartością procedury jest wartość liczbowa amplitudy sygnału w określonym paśmie częstotliwości.
4. FORMAT (I,K): Wartością procedury jest położenie K-tego maksimum obwiedni amplitudowej I-tego widma chwilowego.
5. SREDNIA (NAME,FUN,L): Procedura dokonuje uśrednienia wartości funkcji widmowej FUN dla wszystkich widm dynamicznych obrazu o nazwie NAME.
6. COMPRESS (NAME): Po wykonaniu tej procedury w ramach obrazu o nazwie NAME wyeliminowane zostają wszystkie stany ustalone widma.
7. ERR (NAME 2, NAME 1, NAME): Jest to procedura wykrywania i wewnętrznej obsługi błędów.
8. LOOP (NAME): W wyniku działania procedury LOOP powielony zostaje obszar pamięci, odpowiadający obrazowi o nazwie NAME.

1) W skład grupy ART wchodzą języki ART 73a dla maszyny ODRA 1013 ART 73A dla ODRA 1304, ART 73B (opisywany w artykule) i ART74 dla ODRA 1204 (opracowany wspólnie z IPPT-PAN).

Mgr inż. RYSZARD TADEUSIEWICZ (ur. 1947 r.) ukończył studia z wyróżnieniem na Wydziale Elektrotechniki Akademii Górniczo-Hutniczej w Krakowie (1971 r.). Pracuje obecnie jako starszy asystent w Instytucie Automatyki i Elektroniki Przemysłowej AGH. Jest członkiem Sekcji Przetwarzania Sygnałów Komitetu Informatyki PAN. Zasadniczym polem jego badań naukowych jest bionika — kończy w tej chwili dysertację doktorską na temat bionicznego podejścia do zagadnień automatycznej percepcji słuchowej.

Mgr inż. JERZY ROMAN JAWOROWSKI (1951) ukończył studia na Wydziale Elektrotechniki Akademii Górniczo-Hutniczej (1974 r.). Jest autorem kilku prac z zakresu wykorzystania maszyn cyfrowych w badaniach akustycznych. Jest stażystą w Instytucie Informatyki i Automatyki Akademii Górniczo-Hutniczej w Krakowie.

## Procedury przetwarzania zbiorów

Są to procedury, które operują obrazami akustycznymi traktowanymi jako zbiory, łącząc je, usuwając z pamięci, kopiując i porównując. Procedury tej grupy pozwalają analizować fragmentami jednorazowo zarejestrowany przebieg akustyczny, łączyć kilka fragmentów osobno zarejestrowanych w jeden obraz złożony lub wykonywać „duplikaty" obrazów dla celów obróbki, zwłaszcza jeśli obróbka zmienia postać obrazu (np. operacja COMPRESS).

9. ZERP (NAME): Procedura wymazuje z pamięci obraz o nazwie NAME i udostępnia programowi powstałą lukę.

10. NCOPY (NAME): Procedura kopiuje obraz o nazwie NAME.

11. NSUMA (NAME 1, NAME 2): Procedura generuje obraz będący teoriomnogościową sumą obrazów o nazwach NAME 1 i NAME 2.

12. NEQUIVAL (NAME 1, NAME 2): Procedura arytmetyczna, której wartość wynosi 1, jeśli obrazy o nazwach NAME 1 i NAME 2 są identyczne.

## Procedury analizy widmowej

Procedury tej grupy obliczają wartości (typu REAL) momentów widma lub formantów widma. Procedury te analizują jedynie widma chwilowe; chcąc użyć ich do wyznaczenia przebiegu zmian formantu w czasie należy wywoływać je cyklicznie dla wszystkich widm obrazu. Z procedur tych korzystają procedury analizy statystycznej.

13. FMOMENT (I,L): Procedura, której wartością jest wartość L-tego momentu w I-tym widmie chwilowym.

14. FORMANT SQ(I,L) i FORMANT LG(I,L): Procedury te wyznaczają położenie L-tego formantu w I-tym widmie chwilowym. Zadanie procedury FORMANT LG jest identyczne, jak w przypadku procedury FORMANT SQ, jedyna różnica polega na tym, że w procedurze LG położenie formantu lokalizowane jest metodą średniej ważonej.

## Procedury analizy statystycznej

Procedury tej grupy zapewniają pełną możliwość operowania obrazami akustycznymi jako obiektami analizy statystycznej. Możliwe jest w szczególności uśrednianie widm i parametrów (momentów, formantów) widm, obliczanie dyspersji, korelacji, regresji itp. Grupa ta może być łatwo poszerzona o procedury dostępne w standardowym pakiecie statystycznym, wchodzącym w skład oprogramowania maszyn cyfrowych ODRA seria 1300, gdyż z poziomu języka ART dostępne są wszystkie możliwości FORTRANU 1900, a w tym możliwość dołączania do programu modułów bibliotecznych. Podkreślić jednak należy, że o ile procedruy ART działają bezpośrednio na obrazach, o tyle procedruy biblioteczne systemu ODRA 1300 i język FORTRAN wymagają liczb, konieczne jest więc wstępne przygotowanie danych (programem napisanym w języku ART 73B), aby możliwe było przejście do segmentów bibliotecznych lub fortranowskich. Nie ma natomiast żadnych kłopotów przy stosowaniu w ART zaczerpniętych z FORTRANU procedur i dyrektyw instrukcji warunkowych, logicznych, pętli, funkcji standartowych, procedur redakcyjnych i szeregu innych możliwości, jakich dostarcza oprogramowanie maszyny ODRA 1300.

16. DYSWID(NAME): Procedura funkcyjna, której wartością jest wektor dyspersji w poszczególnych pasmach częstotliwości w obrębie obrazu o nazwie NAME.

17. DYSPER (NAME,FU,E,L): Procedura ta oblicza dyspersję wartości funkcji widmowej FU w obrębie obrazu o nazwie NAME.

18. AKOREL(NAME,FU,E,L,FS,F,K): Procedura funkcyjna, obliczająca w obrębie obrazu o nazwie NAME współczynniki korelacji funkcji FU i funkcji FS.

19. AKORWID (NAME,M1,M2): Wartością procedury jest wartość współczynnika korelacji sygnałów w pasmach częstotliwości numer M1 i M2.

20. AKORBLOK (NAME 1, NAME 2, M1,M2): Wartością procedury (typu funkcyjnego) jest wartość współczynnika korelacji pomiędzy pasmem częstotliwości o numerze M1 w obrazie NAME 1, a pasmem częstotliwości o numerze M2 w obrazie NAME 2.

21. WIDMO(NAME): Procedura dokonuje uśrednienia widma w obrębie obrazu o nazwie NAME. Uśrednienie dokonywane jest osobno w poszczególnych pasmach częstotliwości.

22. COORDINATE (NAME,FU,Z): Procedura określa współrzędne obrazu NAME w pięciowymiarowej przestrzeni. Koordynatami przestrzeni są wartości dwu pierwszych momentów widma i trzech pierwszych formantów.

## Procedury wejścia-wyjścia

Ta grupa procedur ma na celu ułatwianie pracy programisty przy wyprowadzaniu typowych informacji o przetwarzanych obrazach akustycznych oraz przy wprowadzaniu obrazów do pamięci z konwertora KART 1 lub z taśmy papierowej czy z kart.

23. TEST(NAME): Procedura wejściowa, powodująca wprowadzenie, umieszczenie w pamięci i obdarzenie nazwą NAME przebiegu akustycznego z konwertora KART 1.

24. WL(NAME): Procedura dokonuje liczbowego wydruku wartości amplitud w poszczególnych pasmach częstotliwości obrazu NAME. Wydruk automatycznie zaopatrywany jest w teksty informacyjne.

25. WR(NAME): Procedura pozwala na wyprowadzenie obrazu o nazwie NAME w postaci rysunkowej (wykres widma). Rysunek jest opisany i przeskalowany, drukowana jest również nazwa obrazu.

26. WIDEOGRAM(NAME): Procedura wyprowadza widmo dynamiczne obrazu NAME w postaci wykresu we współrzędnych: częstotliwość, czas amplituda. Osie częstotliwości i czasu są opisywane w sposób jawny, natomiast amplituda opisywana jest stopniem zaczernienia papieru.

Przedstawiony w pracy język ART jest stale ulepszany i poszerzany o nowe możliwości. Rozwój języka i wzbogacanie jego możliwości zależne są jednak od dalszych doświadczeń.

LITERATURA


[1] Bowman C.: Package to "Name" Records, Computer Weekly. 106, 26 Sept. 1968, str. 6, 7

[2] Engeli M. C.: Al. Language and List Structure for an Algebraic Manipulation System. Symbol Manipulation Languages and Techniques North Holand, Amsterdam-London 1971, str. 103—115

[3] Jaworowski J., Tadeusiewicz R.: Określanie cech — diagnostycznych w sygnale dźwiękowym, generowanym przez pracujący układ mechaniczny, z zastosowaniem maszyny cyfrowej. Diagnostyka urządzeń technicznych. Materiały konferencji, Międzygórze, styczeń 1974

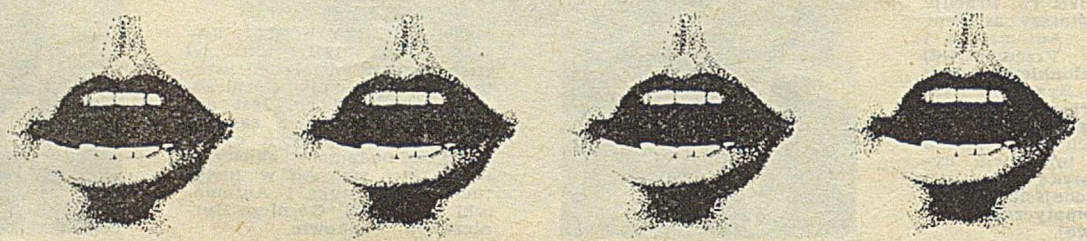
[4] Jaworowski J., Tadeusiewicz R.: Przetwarzanie sygnałów akustycznych dla potrzeb cyfrowych systemów sterowania. Cyfrowe Systemy Sterowania. Materiały konferencji, Wrocław, luty 1974

[5] Kulpa Z.: System dynamicznego przydziału pamięci i przetwarzania listowego MINIPAL/SET 1204. Opracowanie wewnętrzne ICS PAN, Warszawa, 1973

[6] Sakodą J. M.: Dynamic Storage Allocation Language in Fortran Symbol Manipulation and Techniques, North Holland, Amsterdam, London, 1968, str. 302—311

[7] Tadeusiewicz R.: Układ do bezpośredniego wprowadzania informacji akustycznych do maszyny cyfrowej. Metody bezpośredniego wprowadzania i wyprowadzania informacji tekstowej i obrazowej w systemach informacyjnych. Materiały konferencji, Jabłonnna październik, 1973

8. Jaworowski J., Tadeusiewicz R.: Przetwarzanie informacji akustycznej za pomocą maszyny cyfrowej, Archiwum Akustyki (praca w druku)

JERZY SUKIENNIK
Stocznia im. Komuny Paryskiej
Gdynia

681.322.4.11:681.327.11:681.327.64

# Tradycyjne a nowoczesne metody i techniki przygotowania maszynowych nośników informacji

Omówiono praktyczne zastosowanie 2 systemów SEECHECK, których eksploatację rozpoczęto w 1974 r. w Ośrodku Obliczeniowym Stoczni im. Komuny Paryskiej w Gdyni. Podkreślono wady i zalety systemów wielostanowiskowych i przekazano wnioski dotyczące stosowania nowych metod przygotowania maszynowych nośników danych.

Artykuł stanowi pewną kontynuację opublikowanych wcześniej [1] artykułów na temat wielostanowiskowych systemów do przygotowania maszynowych nośników danych oraz praktycznego zastosowania systemów SEECHECK w Stoczni im. Komuny Paryskiej w Gdyni. Celem artykułu jest praktyczna ocena zalet i wad nowych metod przygotowania maszynowych nośników danych na podstawie ponad rocznego doświadczenia eksploatacyjnego.
Istota procesu przygotowania maszynowych nośników danych za pomocą wielostanowiskowych rejestratorów danych na dyskach i taśmach magnetycznych (zwanych umownie *„key-to-disc"*) może być syntetycznie scharakteryzowana następująco:

- małe monitory ekranowe wyposażone w klawiatury alfanumeryczne są uniwersalnie wykorzystywane do wykonywania wielu funkcji takich jak: klawiszowe wprowadzenie i sprawdzanie danych, przygotowanie programów pracy systemu, sterowanie pracą całego systemu, wyszukiwanie danych w zbiorach zapisanych na dyskach itp. Monitory te podłączone są do jednostki centralnej równolegle. System operacyjny umożliwia niezależne wykonywanie różnych funkcji na różnych klawiaturach
- podczas klawiszowego wprowadzania i sprawdzania danych zarówno dane, jak i informacje sterujące oraz sygnały błędów pojawiają się na bieżąco na ekranach monitorów w celu ułatwienia pracy operatorek i operatora nadzorczego
- system wyposażony jest w bibliotekę programów do: formatowania, sortowania, wstępnego przetwarzania, wyprowadzania (drukowania) i transmisji danych; programowanie indywidualne ułatwia w przypadku systemu SEECHECK specjalny język programowania VALIDATOR
- wymienione operacje na zbiorach danych mogą być wykonywane niezależnie i równolegle w stosunku do operacji ręcznego wprowadzania i sprawdzania danych przy czym istnieje możliwość ustawienia priorytetów wykonywania poszczególnych funkcji.

Systemy *key-to-disc* eliminują w pierwszym rzędzie papierowe nośniki danych, mimo że niejednokrotnie mają również możliwość konwersji danych z nośników papierowych. Nośnikami pamięci są dyski magnetyczne *(dysketty)* do tymczasowego przechowania danych oraz standardowe i kasetowe taśmy magnetyczne, na których dane są zapisywane po wprowadzeniu z dysków.
Porównania szeregu cech tradycyjnej techniki przygotowania nośników maszynowych oraz techniki rejestracji magnetycznej przedstawiono w tabeli I.
Dane zawarte w tej tabeli wskazują na olbrzymią przewagę zalet systemów *key-to-disc*.
Należy jednak podkreślić, że dane tabeli I nie uwzględniają stosowania tzw. kart dualnych, na których drogą reprodukcji oraz opisywania emitowane są źródłowe dokumenty produkcyjne i księgowe (zastosowania typowe w zakładach o produkcji masowej i wielkoseryjnej).
Z kolei systemy wielostanowiskowe mają również dwie podstawowe wady:

1) p. INFORMATYKA nr 3 i 5/74

— wysokie nakłady inwestycyjne na zakup urządzeń
— awaria jednostki centralnej systemu (minikomputer, dysk i taśma magnetyczna [2]) unieruchamia cały system (wszystkie klawiatury).

Awarie urządzeń nie są wprawdzie częste, ale w przypadku dużej liczby klawiatur wskazane jest zdublowanie jednostki centralnej.

## Praktyczne doświadczenia ekploatacyjne

W Stoczni im. Komuny Paryskiej zainstalowano, uruchomiono i rozporzęto wdrażanie 2 systemów SEECHECK w drugim półroczu 1973 roku (wrzesień-październik). Wdrażanie to zakończono w I kwartale 1974 roku. Poprzednio stosowano technikę kart dziurkowanych do przygotowania maszynowych nośników danych oraz zgromadzono znaczny zasób doświadczeń w realizacji systemu APD na dużych komputerach: ICT 1904 oraz ICL 4-50 i 4-70.
Przygotowania do wdrożenia systemów SEECHECK objęły szereg równolegle prowadzonych działań:
— przeszkolenie 2 inżynierów-elektroników (10 tygodni), 1 projektanta systemu APD, 1 programisty oraz 3 operatorek nadzorujących system SEECHECK (po 2 tygodnie)
— przetłumaczenie wszystkich list operacyjnych operatorek oraz operatorek nadzorujących (tzw. listy HELP) oraz komunikatów wyświetlanych przez system na język polski (jest to sprawa bardzo istotna ze względu na konieczność szybkiej orientacji operatorek i operatorek nadzorujących w działaniu systemu)
— przygotowanie dla systemów odpowiednich pomieszczeń wraz z instalacjami
— opracowanie formatów dokumentów wyjściowych (rekordów wejścia) rozplanowanie danych na monitorze ekranowym oraz ustalenie pełnej specyfikacji cech rodzajowych i metod kontroli poprawności każdego pola
— opracowanie użytkowych programów kontroli, wstępnego przetwarzania, wydruku i formatowania zbiorów na standardowej taśmie magnetycznej wykorzystywanej na dużych komputerach

2) W przypadku awarii jednostki pamięci na taśmach magnetycznych praca na klawiaturach oraz wykonywanie innych funkcji systemu jest możliwe. Nie można jedynie wyprowadzać danych lub zapisywać dane na taśmie magnetycznej z dysku w celu zabezpieczenia zbiorów oraz przygotowania ich do dalszego przetwarzania w komputerze głównym.

Inż. JERZY SUKIENNIK ukończył Wyższą Szkołę Ekonomiczną w Pradze (CSRS, 1966 r.). Pracuje w Stoczni im. Komuny Paryskiej (od 1967 r.), gdzie obecnie jest kierownikiem Zakładowego Ośrodka Przetwarzania Informacji (1970 r.). Jest również wykładowcą na Uniwersytecie Gdańskim (analiza i projektowanie systemów APD).

Tabela I. Zestawienie cech charakterystycznych przygotowania maszynowych nośników danych na kartach dziurkowanych i taśmie magnetycznej

| Lp. | Technika kart dziurkowanych | Rejestracja magnetyczna (na przykładzie systemu Seecheck) |
|---|---|---|
| 1 | 2 | 3 |
| 1. | Karta perforowana na stałą, organiczoną długość (np. 80 lub 90 kolumn). Konsekwencją tego jest znaczny stopień niewykorzystania karty w przypadku zapisów krótkich (np. 20-kolumnowych) oraz kłopoty z zapisami dłuższymi (np. 81- lub 91-kolumnowymi). W takim przypadku dla jednego zapisu (rekordu) trzeba przeznaczyć 2 karty, które muszą mieć przynajmiej jedną informację wspólną (tzw. klucz pracujący, np. numer dowodu źródłowego). Wymaga to poświęcenia szeregu kolumn karty oraz podwójnego dziurkowania i sprawdzenia klucza parującego | W systemie Seecheck minimalna programowana długość zapisu wynosi 1 znak a maksymalna ponad 9 tys. znaków. Podział na pola jest dowolny. Najdłuższe pole może mieć 999 znaków.<br><br>Umożliwia to zaoszczędzenie znacznej części pracy operatorek: w technice kart klucz parujący o długości np. 6 kolumn bywa dziurkowany i sprawdzany na kilkudziesięciu tysiącach kart miesięcznie. Dodatkowym kłopotem były przy powtórnym odczycie numeru dowodu (klucza parującego), co powodowało błędne parowanie zapisów na taśmie magnetycznej komputera. |
| 2. | Cechą kart perforowanych jako nośnik danych jest sztywność formatów rozplanowania pól. Jak wiadomo, dla prac powtarzalnych stosowane są karty nadrukowane w drukarni (podział na pola rubryki, nagłówki i człony pól). Ze względu na częste i liczne zmiany formularzy dokumentów źródłowych, zmiana formatu (wzoru) kart wymaga kilkumiesięcznego okresu wyprzedzenia projektu nowych kart w stosunku do momentu ich wdrożenia. | Zmiana formatu dokumentu wejściowego w systemie SEECHECK polega tylko na korekcie odpowiedniego programu, co zajmuje przeciętnie kilka godzin pracy programisty lub operatora nadzorczego. Nie wymagają aktualizacji żadne instrukcje operatorskie ,dotyczące np. ustawiania maszyn, ponieważ operacja ta odbywa się tu automatycznie. W tej sytuacji w praktyce stosuje się maksymalnie elastyczne dopasowywanie formatu zapisu wejściowego do dokumentu źródłowego, ponieważ z różnych zapisów wejściowych można programowo wyprowadzić identyczny zbiór wyjściowy. |
| 3. | W warunkach eksploatacji maszyn do perforacji i sprawdzania kart obowiązuje ścisła specjalizacja (jednofunkcyjność) zarówno maszyn, jak i operatorek. Powoduje to szereg kłopotów z koordynacją i prawidłowym obciążeniem stanowisk, szczególnie w warunkach dużej absencji oraz zmiennej wydajności pracy operatorek. Dodatkowe trudności występują w sytuacji, gdy operatorka, np. sprawdzarki musi przejść do pracy na dziurkarce (spadek wydajności, więcej błędów).<br>Nawet w obrębie tej samej grupy maszyn występują problemy z zamianą maszyny, ponieważ operatorka przyzwyczaja się się do swojej (minimalnie różne rozstawienie klawiszy, równa siła uderzeń w klawisze itp.) | Wszystkie klawiatury systemu SEECHECK są identyczne: alfanumeryczne z wydzielonym (powtórzonym) blokiem cyfrowym. Każda z klawiatur może być w dowolnym momencie przełączona na potrzebny tryb pracy (wprowadzanie, sprawdzanie, badanie zawartości zbioru itp.), a nawet na tryb pracy operatora nadzorczego sterującego pracą całego systemu.<br>Umożliwia to zaniechania podziału operatorek na poprzednio występujące ,,specjalizacje'': dziurkarki — sprawdzarki, alfanuemeryczne — numeryczne, maszyny różnych producentów itd. Mimo, że każda operatorka zarówno wprowadza jak i sprawdza dane, stosowana jest zasada,że operatorka nie sprawdza ,,swoich'' paczek dokumentów, ale paczki wprowadzone przez koleżankę. |
| 4. | Popełnienie błędu przez operatorkę dziurkarki powoduje konieczność bądź powtórnej perforacji nowej karty, bądź jej reprodukowania. Konsekwencją tego jest niszczenie dużej ilości kart oraz pojawianie się błędów popełnianych w procesie reprodukcji i kompletacji kart. Nakłady na zakup nośników papierowych są w związku z tym wysokie. | Korekta danych na nośnikach magnetycznych w systemie SEECHECK dokonywana jest za pośrednictwem buforu w pamięci operacyjnej jednostki centralnej.<br>Umożliwia to posiadanie zawsze aktualnej postaci danych.<br>Nośniki magnetyczne są wielokrotnie wykorzystywane (często w okresie kilku lat). |
| 5. | Zarówno magatynowanie kart przed użyciem do perforacji (sezonowanie) oraz przechowywanie międzyoperacyjne, jak i archiwowanie po wykorzystaniu, wymaga spełniania określonych warunków, szczególnie pod względem wilgotności względnej powietrza w pomieszczeniu. Mała gęstość zapisu na kartach wymaga wielkich powierzchni pomieszczeń specjalnie wentylowanych.<br>Mimo to, w procesie wprowadzania danych z kart przy pomocy czytników komputerowych występują częste kłopoty (tzw. ,,zarywanie''), wynikające głównie z nieodpowiednich warunków przechowywania, nie mówiąc już o częstych przypadkach niszczenia i gubienia kart przez operatorów, | Przechowywanie taśm magnetycznych o znacznie większej gęstości zapisu (32 zn/mm) nie wymaga tak dużych pomieszczeń. Zarówno nośniki, jak urządzenia systemu SEECHECK nie wymagają pomieszczeń klimatyzowanych. Przykładowo 2 systemy zainstalowane w stoczni (30 monitorów, drukarka wierszowa, magazyn taśm magnetycznych itp.) zajmują jedynie 100 $m^2$ powierzchni. |
| 6. | Klasyczne dziurkarki i sprawdzarki nie mają odpowiedniej pamięci oraz programów pozwalających na automatyczne wykonywanie niektórych funkcji, zwłaszcza zapewniających wszechstronną programową kontrolę danych. | System komputerowy do przygotowania danych na taśmie magnetycznej na rozbudowany system operacyjny sterujący pracą zarówno urządzeń, jak i operatorek w procesie ręcznego wprowadzania i sprawdzania danych. |
| 7. | Kontrola kompletności i poprawności zbioru przygotowanego na kartach dziurkowanych jest bardzo utrudniona (pojedyńcza karta łatwo może ulec zagubieniu lub podwójnemu wprowadzeniu do komputera).<br>Brak programów kontroli w dziurkarkach i sprawdzarkach kart uniemożliwia automatyczne (programowe) sprawdzanie poprawności danych w dokumentach źródłowych oraz kontrolę pracy operatorek. | Zbiór zapisanych na dyskach i taśmach magnetycznych nie stwarza możliwości utraty lub zdublowania zapisów (partii) danych.<br>Zarówno oprogramowanie podstawowe systemu jak i własne programy kontrolne użytkownika zapewniają szczególnie skuteczną kontrolę poprawności danych. |
| 8 | Konwersja danych z kart lub taśmy papierowej na dyski lub taśmy magnetyczne komputera głównego jest procesem czasochłonnym oraz kosztownym ze względu na angażowanie dużej części urządzeń tego komputera na operacje stosunkowo proste. | Zbiory zapisane na taśmach magnetycznych w systemie SEECHECK nie wymagają dodatkowych przebiegów konwersji na inny nośnik. |
| 9. | Stosunkowo niska wydajność pracy operatorek maszyn perforujących oraz duża ilość błędów w nośnikach papierowych przy dużej uciążliwości i trudności pracy. | Znaczne zwiększenie wydajności pracy operatorek a tym samym ogólnej przepustowości systemu przygotowania nośników maszynowych dzięki:<br>— wyeliminowaniu konieczności ustawienia maszyn przed zmianą rodzaju pracy<br>— szybszemu automatycznemu wykonaniu niektórych często występujących operacji (skoków, duplikacji, wyszukiwania zapisów lub pól itp.)<br>— zaniechaniu ręcznego sprawdzania wielu pól<br>— zmniejszeniu uciążliwości pracy, głównie w wyniku cichej pracy urządzeń. |
| 10. | Duży ciężar nośników papierowych, co stwarza problemy w ich fizycznym przemieszczaniu. | Eliminacja uciążliwości transportu nośnika dzięki:<br>— olbrzymiej pojemności informacyjnej krążka taśmy magnetycznej<br>— możliwości transmisji danych ze zdalnie podłączonych stanowisk klawiaturowych lub za pomocą systemu do komputera głównego. |

— przeszkolenie operatorek klawiatur (1 tydzień) oraz praktyczny trening w celu osiągnięcia wydajności przynajmniej poprzednio osiąganej na dziurkarkach i sprawdzarkach kart, co wymagało okresu 1—3 miesięcy osiągnięcie pełnej, potencjalnie wyższej wydajności pracy w nowym systemie następowało w okresie 5—8 miesięcy.
Podkreślenia wymaga fakt, że tabela III zawiera dane o indywidualnej wydajności pracy poszczególnych grup operatorek, przy czym system SEECHECK ma specjalne programy rejestracji liczby uderzeń (naciśnięć klawiszy) oraz czasu efektywnej pracy operatorek (tzw. statystyka pracy operatorek).
Innym zagadnieniem jest natomiast wzrost ogólnej przepustowości zespołu operatorek dzięki zarówno wyższej indywidualnej wydajności pracy jak i zasadniczemu zmniejszeniu ilości znaków powtórnie wprowadzanych (sprawdzanych ręcznie — klawiszowo), co ilustrują dane w tabeli II.
O ile średni wzrost indywidualnej wydajności pracy operatorek możliwy jest wg naszych doświadczeń w granicach 20—25%, o tyle wzrost przepustowości systemu przygotowania danych możliwy jest w granicach 40—50%.
Możliwe to jest w wyniku prawidłowego ustawienia (zorganizowania) procesu emisji dokumentacji źródłowej w przedsiębiorstwie, a następnie prawidłowej kontroli wstępnej i międzyoperacyjnej dokumentów wkraczającej w proces szeroko rozumianego przygotowania danych.
Ilustracją tego problemu są następujące liczby pracowników:
Dział Kontroli Wstępnej i Wynikowej — 25 osób,
Zespół Przygotowania Danych (SEECHECK) — 35 osób,
Zespoły te przerabiają miesięcznie następujące liczby dokumentów:

| | |
|---|---|
| Płace | — 70.000 [3] |
| Gospodarka Materiałowa | — 38.000 [3] |
| Indeks | — 2.500 |
| Normy Materiałowe | — 13.000 |
| Planowanie | — 2.000 |
| | 125.000 (ok. 16 mln znaków) |

Dalszym warunkiem osiągnięcia wysokiej przepustowości systemu SEECHECK jest wykorzystanie udogodnień zawartych w programach standardowych:
— system operacyjny
— kompilator programów parametrycznych zapisów (rekordów) wejścia
— kompilator języka programowania VALIDATOR
— generator programów sortujących
— programy pomocnicze zabezpieczenia zbiorów danych i bibliotek, diagnostyka itp.

System SEECHECK jest bardzo elastyczny w programowaniu i operowaniu. Przeciętnie uzdolniony programista wymaga szkolenia kursowego w rozmiarze 100 godzin oraz praktyki trwającej około 2 miesiące.

Po tym okresie może on napisać, wprowadzić samodzielnie do systemu (poprzez klawiaturę i monitor), skompilować i wytestować program w przeciągu 1—2 dni, co w porównaniu do „klasycznej" organizacji procesu przygotowania programów daje okres przynajmniej dwa razy krótszy.

Tabela II

Specyfikacja metod sprawdzania pól wprowadzanych z dokumentu źródłowego do systemu SEECHECK

| Lp. | Nazwa pola | Liczba znaków wprowadzanych | Metoda sprawdzania |
|---|---|---|---|
| 1. | Symbol wydziału | 3 | Tabela wartości stałych |
| 2. | Nr dowodu źródłowego | 6 | Ręcznie |
| 3. | Symbol zlecenia | 12 | Tabela wartości stałych kontrola powiązań logicznych |
| 4. | Symbol Zespołu konstrukcyjno-technologicznego | 12 | Cyfra kontrolna |
| 5. | Symbol indeksu materiałowego | 12 | Cyfra kontrolna |
| 6. | Symbol magazynu | 3 | Tabela wartości stałych |
| 7. | Symbol branżysty | 3 | Tabela wartości stałych |
| 8. | Pobierana ilość materiału | 10 | Suma kontrolna |
| 9. | Cena jednostkowa | 10 | Konfrontacja ze zbiorem ideksu materiałowego |
| | Ogólna liczba znaków wprowadzanych oraz sprawdzanych z czego: | 71 | |
| | a) ręcznie (klawiszowe): | 6 | |
| | b) automatycznie (programowo): | 65 | |
| | Uwaga: poprzednio, stosując technikę kart dziurkowanych istniała konieczność sprawdzania ręcznego (klawiszowego) w każdym zapisie wszystkich pól danych, tzn. 71 znaków. | | |

3) szereg dokumentów wielopozycyjnych, z których tworzonych jest kilka lub kilkanaście zapisów (rekordów).

Tabela III

Ilustracja wydajności pracy operatorek

| Kategoria operatorek | Technika kart dziurkowanych | | Wydajność w systemie SEECHECK (znaki/godzinę) |
|---|---|---|---|
| | liczba kart/godzinę | Szacunkowa liczba znaczków/godzinę | |
| Najbardziej wydajne | 300 | 15.000 | 18.000 |
| Średnio wydajne | 200 | 10.000 | 13.000 |
| Mało wydajne | 150 | 7.500 | 10.000 |

Systemy SEECHECK są wykorzystywane w zasadzie na jednej zmianie, ale w okresach spiętrzeń uruchamiane są trzy zmiany i niejednokrotnie systemy pracują „non-stop" przez około 72 godziny. Mimo to, awaryjność systemów jest niewielka i nie przekracza 2%, a czas konserwacji systemów 3% (wykonywanej w dni wolne od pracy).

## Podsumowanie

Najważniejszą zaletą systemów wielostanowiskowych (w tym również systemu SEECHECK) jest wyposażenie w specjalny wielofunkcyjny system operacyjny oraz w generator programów sortujących i kompilator języka programowania. Umożliwia to użytkownikowi przygotowywanie własnych programów kontroli poprawności, wstępnego przetwarzania i formowania zbiorów danych.

Ilustracją efektów dobrego oprogramowania standardowego i użytkowego systemu SEECHECK są dane zawarte w tabeli II. Podają one zestawienie metod sprawdzania na przykładzie jednego z typów dokumentów przygotowanych w Stoczni na systemie SEECHECK (dokument pobrania materiałów). Wykorzystanie standardowych metod kontroli (tabela wartości stałych, cyfra kontrolna, bilansowanie sum kontrolnych itp.) oraz zastosowanie własnych programów kontroli danych, np. kontrola dopuszczalnych powiązań logicznych i formalnych występujących między zleceniami i wydziałami oraz kontrola poprawności rachunkowej danych (typowa dla danych płacowych) umożliwiło w Stoczni wyeliminowanie ręcznego sprawdzania szeregu pól danych, co znacznie zmniejszyło pracochłonność procesu przygotowania nośników maszynowych oraz spowodowało skrócenie cyklu przetwarzania danych.

Usprawnienie procesu przygotowania maszynowych nośników danych warunkuje w znacznym stopniu efektywność całego procesu automatycznego przetwarzania danych.

Coraz większe moce obliczeniowe komputerów narzucają konieczność wzrostu wydajności pracy pracowników zatrudnionych w procesie przygotowania danych, jak również poprawy i kontroli jakości ich pracy. Spełnienie tych warunków zapewnia niewątpliwie zastosowanie systemów wielostanowiskowych rejestracji i wstępnego przygotowania danych na nośnikach magnetycznych, które w najbliższych latach powinny całkowicie zastąpić tradycyjne metody przygotowania danych na nośnikach papierowych.

# Tendencje rozwojowe w dziedzinie pamięci cyfrowych*)

Proces nieustannego rozwoju, który doprowadził — od wynalezionych w końcu XIX wieku przez Holleritha kart dziurkowanych do współczesnej bardzo szybkiej pamięci półprzewodnikowej przebiega ostatnio jeszcze gwałtowniej. Jedną z jego cech jest trwałość technik, które się przyjęły i które zwykle długo opierają się nowym tendencjom. I tak, np. karty dziurkowane są do dziś używane, a pamięciom półprzewodnikowym dopiero ostatnio udało się przełamać dominację wśród pamięci operacyjnych dzięki znacznym środkom przeznaczonym na usprawnienie tych technik oraz obniżeniu kosztów wytwarzania.

Aby odnieść sukces, nowe rozwiązanie musi nie tylko znacznie przewyższać parametry pracy, jakie charakteryzują istniejące rozwiązania konstrukcyjne — lecz musi mieć także potencjalne możliwości dalszego rozwoju.

W ciągu ostatnich kilku lat — dzięki opanowaniu techniki wielkiej integracji — koszty pamięci półprzewodnikowej (wraz ze wzrostem gęstości upakowania) znacznie się obniżyły. Jednocześnie kontynuowany jest też postęp w dziedzinie pamięci magnetycznej, aczkolwiek nie tak szybki w ostatnich latach, gdyż technika ta jest bardziej dojrzała.

W tym samym czasie pojawiły się nowe rozwiązania takie, jak cylindryczne domeny magnetyczne *(bubbles)* i zachowujące elektrycznie wymienianą zawartość — pamięci półprzewodnikowe (np. MNOS) [1]), które posiadają własność zachowywania informacji i prawdopodobnie będą tańsze. Aby więc osiągnąć sukces, techniki optyczne muszą współzawodniczyć w silnej konkurencji i to w sytuacji, gdy pamięci hierarchiczne z możliwościami przemieszczeń zmniejszą znaczenie tego, co było ich ważną zaletą — swobodnego dostępu poprzez wybieranie wiązką świetlną. Poniżej przedstawiono graniczne tendencje kształtowania się kosztów oraz działania układów pamięciowych w ciągu najbliższych 10 lat. Ich analiza prowadzi do wniosku, że o ile układy optyczne wejdą do hierarchii pamięciowej, będzie to miało miejsce najwcześniej przy bardzo dużych pojemnościach rzędu $10^{12}$ — $10^{13}$ bitów.

Ponadto muszą one poprawić swe parametry pracy conajmniej o rząd wielkości w stosunku do istniejących rozwiązań, a także w identycznej proporcji obniżyć koszt.

## Zapis magnetyczny

W ciągu dwudziestu lat od wprowadzenia pierwszych rozwiązań pamięci taśmowych i prawie dwudziestu od wprowadzenia dysków, techniki zapisu magnetycznego stale się rozwijały, osiągając coraz lepsze parametry pracy przy obniżonych kosztach.

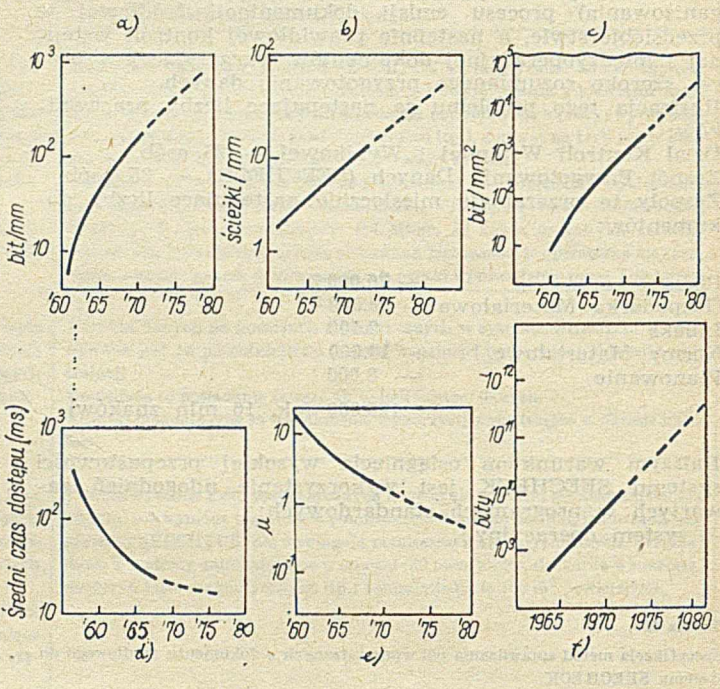
W roku 1973 czołowe firmy amerykańskie IBM, STORAGE TECHNOLOGY CORP. i TELEX wprowadziły rozwiązania taśmowe o gęstości zapisu 246 bajtów/mm, co oznacza postęp 250-krotny w ciągu 20 lat [2]). Przy prędkości przesuwu ok. 5 m/s daje to prędkość przekazywania informacji 1,25 Mbajtów/s, a więc skok ponad 1000-krotny w ciągu 20 lat.

Jednocześnie te usprawnione pamięci taśmowe charakteryzują się dziś prawie o dwa rzędy wielkości niższym poziomem kosztów w przeliczeniu na 1 bit.

Jeszcze większy postęp został osiągnięty i nadal jest kontynuowany w dziedzinie pamięci dyskowych. Jest to pokazane graficznie na rys. 1. Poprawia się zarówno gęstość bitowa jak i zagęszczenie ścieżek dając co 6 lat ciągły 10-krotny wzrost gęstości powierzchniowej.

Obecnie jednostka pamięci dyskowej IBM 3330 o podwójnej gęstości pozwala na zapis około 220 bitów/mm przy ok. 12 ścieżkach/mm co daje ponad 2300 bitów/mm² przy czasach dostępu rzędu kilkudziesięciu ms i maksymalnej szybkości przesyłania informacji 7 Mbitów/s. RANK XEROX oferuje dysk o pojemności ponad biliard bitów, a UNIVAC trzykrotnie większej. Cena kształtuje się obecnie na poziomie $2{,}5 \cdot 10^{-4}$ centa/bit.

Gęstości te występują przy prędkości przesuwu zbliżonej do 50 m/s i wówczas szczelina głowicy magnetycznej i wysokość na jakiej ona się unosi, a także grubość ośrodka magnetycznego muszą być rzędu mikrona (rys. 1e). Do wytwarzania cienkowarstwowych głowic zastosowana została technika mikromaszynowa [1], uzyskując szczeliny poniżej 0,8 μ do stosowania przy ośrodku o grubości 1,27 μ [1, 2].




Rys. 1 Tendencje rozwojowe pamięci dyskowych, a) liniowa gęstość informacji, b) gęstość ścieżek, c) powierzchniowa gęstość informacji, d) średni czas dostępu, e) typowe wymiary zapisu, f) pojemność maksymalna

Jak dotychczas nie ujawniło się żadne zasadnicze ograniczenie w technice zapisu i można tu oczekiwać dalszego postępu. Np. VRC California podaje osiąganą gęstość prawie 60 ścieżek/mm w układach ze stałymi głowicami.

Te osiągnięcia techniczne pozwalają na obniżanie kosztu na bit informacji przechowywanej w rozwiązaniach z zapisem magnetycznym z prędkością większą niż jeden rząd wielkości w ciągu 10 lat (rys. 2).

Na rynku większą uwagę przywiązuje się obecnie do poziomu kosztów niż do parametrów użytkowych urządzeń. Można o tym wnioskować z faktu, że spadek wskaźnika kosztu utrzymuje się, podczas gdy skracanie czasu dostępu (rys. 1d) odbywa się w coraz wolniejszym tempie. Gdyby bowiem parametry użytkowe były najistotniejsze, czasy dostępu wpływałyby na zmianę cen.

*) Na podstawie artykułu E. S. Berrekette'a: „Trends in storage of digital data" w: Applied Optics no 4 vol. 13 z kwietnia 1974

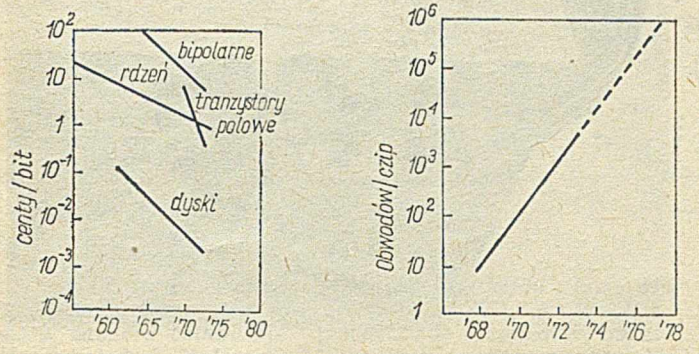
1) MNOS = Metal Nitride Oxide Semiconductor — ulepszona technologia jednopolarnych pamięci półprzewodnikowych wykorzystująca azotek,

Na początku 1975 r. (DATA PROCESSING no 1, vol. 17, styczeń — luty 1975) IBM ogłosiło parametry swojej pamięci 3850 MASS STORAGE SYSTEM, zawierającej m.in. pojemniki z taśmami o gęstości zapisu 2569 bajtów/mm. Pamięć ta, będąca kombinacją rozwiązań taśmowych i dyskowych o maksymalnej pojemności $4{,}72 \times 10^{11}$ bajtów, ma być wprowadzona na rynek w I połowie 1976 r.

## Pamięci półprzewodnikowe

W dziedzinie pamięci operacyjnych rdzenie ferrytowe dominowały przez ponad 20 lat [3]. Postęp w tej dziedzinie obserwuje się nadal.

Prowadzi to do produkcji gabarytowo mniejszych, szybszych i tańszych urządzeń. Np. firma DATA PRODUCTS oferuje pamięć rdzeniową o pojemności $6 \times 10^5$ bitów przy koszcie 0,5 centa/bit. Nie mniej jednak, mimo tego postępu, rdzenie stały się w końcu ofiarą półprzewodników.




Rys. 2 Tendencje rozwojowe kosztów różnych pamięci
Rys. 3 Tendencje w liczbie bitów na podzespół (czip)

Parametry użytkowe pamięci półprzewodnikowych już od pewnego czasu okazały się lepsze. Przyczyniło się to do sukcesu hierarchicznych układów pamięciowych, gdzie wykorzystane zostały półprzewodnikowe pamięci buforowe, a rdzenie pozostały tylko w pamięci głównej. Jednakże, wraz ze spadkiem kosztu pamięci półprzewodnikowej rozpoczął się proces przechodzenia od układów magnetycznych do półprzewodnikowych. Np. firma DATA GENERAL w 1970 r. wprowadziła pamięć bipolarną w minikomputerze SUPERNOVA SC, a następnie pamięci na tranzystorach z efektem polowym typu MOS zostały wprowadzone do pamięci operacyjnych większości nowoopracowanych systemów.

Podzespół CZIP firmy INTEL o pojemności 1k(1024 bitów) wprowadzony początkowo po koszcie 1 c/bit został szeroko rozpowszechniony na rynku. W roku 1972 firma SINETICS oferowała pamięci na elementach typu MOS po około 0,3 centa/bit. Pod koniec 1972 r. firma INTEL wprowadziła czip o pojemności 4k, a w roku 1973 takie podzespoły oferowały firmy MOSTEK [4], PHILIPS [5], podczas gdy IBM [6] opracowała czip o pojemności 8k. Wszystkie te podzespoły mają być wykorzystane w pamięci z natychmiastowym dostępem. Dla pamięci stałych poziom integracji jest jeszcze wyższy — 1k bitów na czip w technice bipolarnej i 16k przy tranzystorach z efektem polowym.

Liczba bitów na jednostkę powierzchni krzemu wzrasta o rząd wielkości co 3 lata (rys. 3), podczas gdy koszty spadają około 50% rocznie (rys. 2).

Tutaj również — jak dotąd — nie osiągnięto zasadniczych ograniczeń i można oczekiwać dalszego postępu. Opracowano rozwiązania [7] wykorzystując jeden układ na komórkę pamięciową, zajmujący zaledwie ok. $6{,}5 \cdot 10^{-4}$ mm² na komórkę bitową, a za pomocą technologii wykorzystującej strumień elektronów [8] osiągnięto nawet $0{,}8 \cdot 10^{-4}$ mm²/bit przy szerokościach linii 1 μ.

Układy o sprzężeniu ładunkowym *(charge coupled devices* — CCD) zezwalają na dalsze usprawnienia w gęstości, a tym samym i obniżenie kosztu. Opisano dwufazowy układ CCD [9], który zajmuje połowę powierzchni komórki jednoukładowej oraz szeregowo-równoległy układ adresowania szeregowego [10] wymagający N+1 elektrod od adresowania N bitów, zamiast konwencjonalnych 2N elektrod [4]. Ponadto nowe nakłady mają zredukowane zapotrzebowania na moc, co powinno rozwiązać wkrótce problem zachowania informacji w pamięci półprzewodnikowej.

Np. firma MOTOROLA wprowadziła pamięć stałych na układach typu CMOS (MOS komplementarne) wymagają tylko 11 nW mocy spoczynkowej, a ponadto dokonano postępu w dziedzinie obwodów małej mocy technologii krzemu w szafirze, wymiennych, zachowujących informację w układach typu MNOS i podobnych [11].

## Pamięci na cylindrycznych domenach magnetycznych

Pamięci na cylindrycznych domenach magnetycznych dokonały rekordowego postępu w laboratoriach badawczych (rys. 4). Aczkolwiek nie są one jeszcze dostępne na rynku [5].

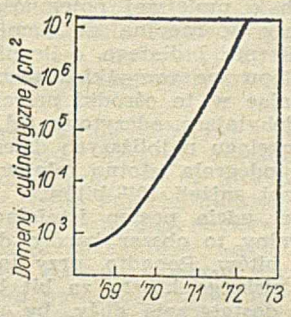
Jeśli tendencja ta utrzyma się mogą być osiągnięte fantastyczne gęstości. Istnieją już pracujące prototypy [12] i analizowane są pełne systemy. Firma BELL TECHNICAL LABORATORIES (BTL) opisała [13] podzespół czip o pojemności 20 k bitów.

Warto zauważyć, że na tym bardzo wczesnym etapie swojego rozwoju, podzespoły na cylindrycznych domenach magnetycznych osiągają większe pojemności od układów na tranzystorach z efektem polowym. Przewiduje się że koszt jednego bitu pamięci na cylindrycznych domenach magnetycznych będzie niższy, niż pamięci półprzewodnikowych.

Do osiągnięć, które wydają się potwierdzać słuszność tych przewidywań należy zaliczyć: ostatnie opracowania BTL w zakresie techniki wytwarzania rejestrów przesuwnych na cylindrycznych domenach magnetycznych za pomocą pojedynczej maski [14], opracowanie w IBM amorficznych warstw magnetycznych [15] (które można znacznie łatwiej uzyskać w porównaniu z konsekwencjonalnymi monokrystalicznymi granatami). Tutaj również przewidywany jest wiekszy postęp.

## Pamięci optyczne

Od czasu wprowadzenia przez IBM w 1966 roku pamięci PHOTO DIGITAL MASS STORAGE SYSTEM [16] tylko pamięć TERRABIT firmy AMPEX i UNICON [17] firmy PRECISION INSTRUMENTS [6] — ukazały się na rynku w grupie pamięci o pojemności rzędu $10^{12}$ bitów wyprodukowane zaledwie w kilku egzemplarzach. Wśród tych rozwiązań pamięć firmy AMPEX zawiera taśmę magnetyczną, natomiast w systemie IBM zapisu dokonuje się za pomocą strumieni elektronów, a odczyt odbywa się optycznie. Tylko rozwiązanie firmy UNICON jest na prawdę optyczną pamięcią stałych systemów „bit po bicie" [7]. Wszystkie te pamięci charakteryzują się wysokim kosztem (około miliona dolarów) i wolnym dostępem rzędu kilku sekund, ze względu na ograniczenia dostępu mechanicznego.




Rys. 4 Tendencje w gęstości informacji laboratoryjnych układów na cylindrycznych domenach magnetycznych

3) Dokładniejsze porównanie aktualnej produkcji pamięci ferrytowych i półprzewodnikowych, a także omówienie różnych typów pamięci półprzewodnikowych można znaleźć w artykule H.F. Koehlera: Advances in Memory Technology, COMPUTER DESIGN, nr 6, vol. 13, z czerwca 1974 r.

4) W nr 1 czasopisma ELECTRONICS z 9.01.1975 r. podano, że w najbliższym czasie firma FAIRCHLID SEMICONDUCTOR wprowadzi na rynek pierwszą pamięć na elementach typu CCD o pojemności 9216 bitów, składającą się z 9 rejestrów przesuwnych o pojemności 1k każdy. Stanowi ona podzespół w standardowym 18-nóżkowym opakowaniu i posiada średni czas dostępu 165 μs maksymalny 33 μs. W ciągu kilku miesięcy opracowany ma być nowy model o pojemności 16 k.

5) W drugiej połowie 1974 r. (COMPUTER DESIGN, no 9, vol. 13, z sierpnia 1974 r.) firma BELL TELEPHONE LABORATORIES opracowała pamięć do przechowywania 27 000 numerów telefonicznych. Pamięć ta ma pojemność 460 544 bitów, a wymiary układu domen cylindrycznych wynosiły 9,5 × 4,5 × 2 cm, średni czas dostępu 2,7 ms, szybkość przesyłania informacji 700 kbitów/s, błąd odczytu rzadszy niż raz na $6{,}3 \cdot 10^{14}$ operacji.

6) Nowa wersja tej pamięci nazywa się SYSTEM 190 — patrz uwaga w artykule: Optyczne pamięci cyfrowe, INFORMATYKA Nr 12/74, s. 18

7) ang.: *bit by bit*

Czasy dostępu porównywalne do osiąganych w pamięciach dyskowych — możliwe są do użytkania w pamięciach magnetycznych adresowanych wiązką promieniowania elektromagnetycznego [18, 19], w których zapisu dokonuje się przy przekraczaniu temperatury Curie, a odczytu w oparciu o efekt Kerra lub Faraday'a. Zaczynają pojawiać się materiały, które mogą pracować w temperaturze pokojowej (20—23 °C). Pojawiły się również niemagnetyczne, adresowane wiązką fal elektromagnetycznych dyski, np. wykorzystujące warstwy amorficznych chalkogenidów, półprzewodnikowych [24].

Należy jednak zauważyć, że główna zaleta tych systemów, w porównaniu do konwencjonalnego zapisu magnetycznego — przewidywana wysoka gęstość zapisu możliwa przy optycznych technikach zapisu i odczytu — powoli przestaje być decydująca, gdy gęstości informacji w konwencjonalnych pamięciach dyskowych zbliżają się do wartości gdzie występują ograniczenia w pamięciach optycznych (rys 1e).

Zaczynają też pojawiać się systemy wizyjnych pamięci stałych, takie jak PHILIPS VLP [25], MCA DISCO VISION [26] i TELDEC. Pamięci te jednak nie mają stanowisk zapisu (jak to ma miejsce w przypadku systemów UNICON czy PHOTO-DIGITAL MASS STORE) i dlatego nie są jeszcze gotowe do bezpośredniej współpracy z maszyną cyfrową w zastosowaniach domowych czy oświatowych, do których były projektowane.

Prace nad pamięciami holograficznymi, motywowane perspektywą szybkiego dostępu do dowolnego adresu oraz możliwością osiągnięcia wysokiej gęstości informacji, prowadzone są od około 10 lat, lecz nie dały dotychczas na rynku zadowalającego systemu pamięci operacyjnej do uniwersalnych maszyn cyfrowych.

Obecnie możliwości systemów holograficznych są dosyć ograniczone i wydaje się mało prawdopodobne, by mogły one zastąpić pamięci dyskowe o podobnej pojemności i poziomie kosztu na jeden bit oraz krótkich czasach dostępu, porównywalnych w tym zakresie z parametrami pamięci półprzewodnikowej, co często było podawane za podstawowy cel pamięci holograficznej. Główne trudności polegają na ograniczonej rozdzielczości [27], braku naprawdę natychmiastowego twornika stronicy [8] [28, 19], wydajnych, szybkich, odwracalnych ośrodków pamiętania [30] i efektywnych deflektorów światła [31, 32].

## Wnioski

Ze względu na to, że ceny pamięci półprzewodnikowych i pamięci z zapisem magnetycznym ciągle spadają w dużym tempie (rys. 2), układom optycznym jest bardziej trudno wejść na rynek konstrukcji pamięciowych.

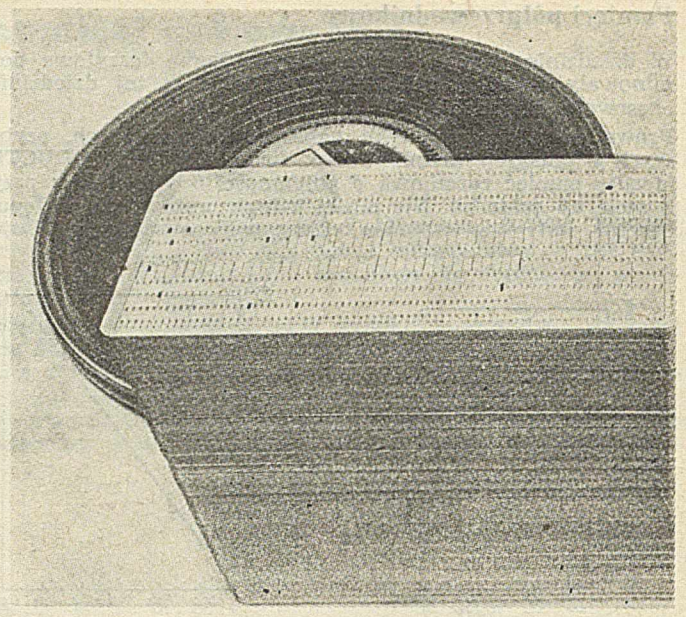
W szczególności uwzględniając hierachię pamięci, w najbliższych dziesięciu latach oczekiwany jest postęp na wszystkich poziomach o mniejszej pojemności — od pamięci buforowej poprzez operacyjną do pamięci dyskowych o bezpośrednim dostępie. I dlatego, o ile nie uzyska się zdecydowanego postępu we wszystkich podzespołach pamięci optycznej, włączając w to ośrodki przechowywania informacji, układy odchylające, odczytowe itd., jest mało prawdopodobne, by w ciągu najbliższych dziesięciu lat cyfrowa pamięć optyczna odegrała istotną rolę w systemach o pojemności mniejszej aniżeli $10^{11}$ bitów.

Jedyną dziedziną, gdzie postęp istniejących technik jest stosunkowo powolny to obszar bardzo dużych pojemności — powyżej $10^{12}$ bitów. Ponadto, przy tak dużych pojemnościach, mimo niskiego kosztu za bit łączny koszt całej konstrukcji jest dostatecznie duży, by umożliwić projektantowi dołączenie skomplikowanych i raczej drogich podzespołów wymaganych do systemów optycznych, zwłaszcza holograficznych. Niemniej jednak koszt ten nie może być tak wysoki, jak w obecnie istniejących rozwiązaniach. Wydaje się, że wartością graniczną dla powstawania takiego rynku jest koszt 100 tys. dolarów za $10^{12}$ bitów czyli $10^{-5}$ centa/bit.

Jednocześnie czas dostępu musi być zredukowany z kilku sekund do poziomu conajmniej kilku dziesiątych części sekundy, aby zmniejszyć dystans do istniejących obecnie pamięci dyskowych, gdzie wynosi on kilka setnych sekundy. Zakres pamięci o bardzo dużej pojemności oczekuje na nowe rozwiązania, a stawiane wymagania nie są zbyt odległe od parametrów obecnie już osiąganych. Tutaj więc pamięć optyczna ma większe szanse do wejścia na rynek w ciągu najbliższych 10 lat.

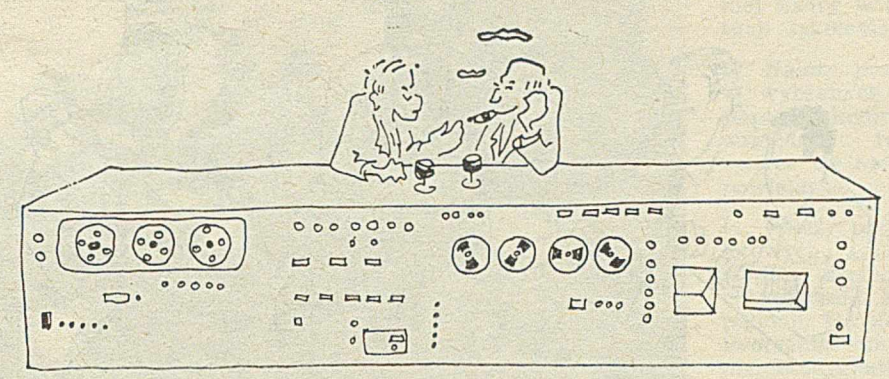
oprac. **Jan Ryżko**

8) ang. *page*



LITERATURA:


[1] L. T. Romankiw, P. Simon "Batch Fabrication of Thin Film Magnetic Recording Heads of Vertical Confiquration" to be published September, 1974, IEEE Trans on Magnetics.
[2] E. P. Valstyn, L. F. Shew, IEEE Trans, on Magnetics, MAG — 9, 317 (1973)
[3] R. L. Mattson, Appl. Opt. 13, 755 (1974)
[4] R. Proebsting, R. Green IEEE Intl. Solid State Cirucuits Conf. Digest of Tech. Papers, 26, 28 (February 1973)
[5] C. W. Lambrechtse, R. Salrtes, L. Boonstra, IEEE Intl. Solid State Cirucuits Conf. Digest of Tech. Papers 26, 26 (February
[6] W. K. Hoffman and H. L. Kalter, IEEE Intl. Solid State Circuits Conf. Digest of Tech. Papers 26, 64 (February 1973)
[7] K. V. Stain and H. Friedrich, IEEE Intl. Solid Statę Circuits Conf. Digest of Tech. Papers 26, 30 (February 1973)
[8] H. Yu. R. Dennard, T. Chang, M. Hatzakis, IEEE Intl. Solid State Circuits Conf. Digest of Tech. Papers. 26, 98 (February 1973)
[9] D. Collins i in.: Double Level Anodized Aluminum CCD, Intl. Electron Devices Meeting, paper 24.1, p. 168, 4—6 December 1972
[10] Collins i in.: Intl. Solid Circuits Conf. Digest of Tech. Papers 26, 136 (February 1973)
[11] S. Fields, Electronics 45, No. 22, pp. 66—67, (23. October
[12] P. C. Michaelis, IEEE Trans, Magnetics, MAG-9, 436 (1973)
[13] P. I. Bonyhard, J. E. Geusic, A. H. Bobeck, Yu-Ssu Chen, P. C. Michaelis, J. L. Smith, IEEE Trans, Magnetics, MAG-9, 433
[14] A. H. Bobeck, I. Danylchk, F. C. Rossol, and W. Strauss, IEEE Trans, Magnetics, MAG-9, 474 (1973)
[15] P. Chaudhari, J. J. Cuomo, and R. J. Gambino, IBM J. 17, 66 (1973)
[16] J. D. Kuehler, H. R. Kerby, in Procedings Fall Joint Computer Conference (IEEE Computer Society, New York, 1966), page 735
[17] K. McFarland, M. Hashiguchi, Proc. AFIPS 23, Pt. 2, 1369 (1968)
[18] R. P. Hunt, IEEE Trans, Magnetics, MAG-5, 700 (1969)
[19] G. J. Fan and J. H. Greiner, J. Appl. Phys. 39, No. 2, Part II 1216 (1968)
[20] R. Aagard, F. Schmit, T. Liu, and D. Chen, IEEE Trans. Magnetics MAG-9, 463 (1973)
[21] H. Wieder, R. A. Burn, IEEE Trans, Magnetics MAG-9, 463 (1973)
[22] C. F. Shelton, IEEE Trans, Magnetics MAG-9, 398 (1973)
[23] S. Wittekoek, J. M. Robertson, T. Popma, and P. F. Bongers, AIP Conf. Proceedings No. 10, Pt. 2, Magnetism Magnetic Materials, 1418 (1972)
[24] Lapidus, IEEE Spectrum 10, No. 1, 44 (January 1973)
[25] K. Compaan, P. Kramer, Philips Tech, Rev. 33, 178 (July 1973)
[26] D. Kay, Electronic Design 21, 35 (1 February 1973)
[27] A. Vander Lugt, Appl. Opt. 12, 1675 (1973)
[28] W. Stewart et al. RCA Rev. 34, 3 (1973)
[29] G. Labrunie, J. Robert, and J. Borel, Appl. Opt. 13, (1974)
[30] „Optical Storage and Display Media" Special Issue of RCA Rev. 33, 3—310 (1972)
[31] O. N. Tufte and D. Chen, IEEE Spectrum 10, 26 (March
[32] O. N. Tufte and D. Chen, IEEE Spectrum 10, 48 (March



*Szanowny Panie Redaktorze,*

*Ponieważ sprawozdanie ze Sztokholmskiego kongresu IFIP, pióra korespondenta INFORMATYKI, p. W. Klepacza, zawiera całkowicie bezpodstawne insynuacje, obraźliwe względem Federacji i szkodliwe względem przyszłej współpracy międzynarodowej, stanowczo proszę o zamieszczenie niniejszego listu na łamach INFORMATYKI. Poprawę błędów faktograficznych, występujących w artykule p. Klepacza pozostawiam Redakcji; szkoda, tylko, że przed wydrukowaniem nie poddano tekstu weryfikacji przez kogoś, kto zna temat.*

*Pozostawiam na uboczu mych uwag ocenę poziomu referatów i referentów, tak apodyktycznie wyrażoną przez p. Klepacza, wiem tylko, że musieliśmy chodzić na różne sesje, bo mnie się zdarzało z trudem znajdować miejsce siedzące na niektórych sesjach, a wiele referatów zachwyciło mnie i swą treścią i sposobem wygłoszenia (może jestem mniej wyrobionym słuchaczem niż specjalny korespondent INFORMATYKI).*

*Nie mogę natomiast pominąć milczeniem uwag p. Klepacza o braku obiektywizmu przy selekcji referatów nadesłanych, który miał dotknąć referentów z krajów socjalistycznych. Niestety, nie mogę posłużyć się danymi liczbowymi i cytatami opinii recenzentów, gdyż te są, ze zrozumiałych względów, poufne i znane w całości tylko Komitetowi Programowemu, który jako ciało całkowicie niezależne, powoływane do przygotowania kongresu i po kongresie rozwiązane, składa globalne sprawozdanie przed zgromadzeniem ogólnym IFIP.*

*Mechanizm selekcji referatów zgłoszonych w IFIP jest bardzo ostry, gdyż nadesłanych prac jest zazwyczaj około 7 — 8 razy więcej niż można przyjąć. Każda praca jest recenzowana przez conajmniej dwu ekspertów z międzynarodowego wykazu, a w przypadku rozbieżności ich opinii — przez trzeciego; każdą pracę recenzuje też przewodniczący odpowiedniej dziedziny kongresu.*

*W przypadku nie przyjęcia pracy autorom wysyła się bardzo dyskretnie sformułowany list bez uwag krytycznych. Autorzy prac przyjętych otrzymują wskazówki recenzentów co do pożądanych poprawek.*

*Przez sito recenzji przedostawali się autorzy polscy na poprzednich kongresach (z pamięci cytując: p. mgr Dańda — w Edynburgu, p. dr Olszewski — w Lublanie, niżej podpisany — w Nowym Jorku) są też zapraszani do wygłoszenia referatów głównych (dr Waligórski — Edynburg, prof. Seidler — Sztokholm). Nie wiem skąd p. Klepacz uzyskał informacje o licznych zgłoszeniach prac polskich na Kongres Sztokholmski, na pewno nie uzyskał tej wiadomości od nikogo kompetentnego.*

*Ponieważ nie mogę, jak napisałem powyżej, posłużyć się danymi liczbowymi, proponuję by p. Klepacz udowodnił na czym konkretnie polega brak obiektywizmu, i wyjśnił czy dotknął on tylko Polskę, czy może też Holandię (ani jednego referatu) albo Bułgarię (dwa referaty). Ponieważ argumentów rzeczowych p. Klepacz nie przytacza — to jego sformułowanie, bardzo przepraszam, ale przypomina mi niedobre tradycje okrzyku „naszych biją"!*

*Zamiast więc drukować niesprawdzone plotki i insynuacje warto może podjąć rzeczową analizę trybu krajowych przygotowań do kongresu IFIP. Dowiedzielibyśmy się wtedy dlaczego referaty z Polski nie są poddawane wewnętrznej, krajowej selekcji (a robi tak i USA, i Francja, i wiele innych krajów), dlaczego w licznej grupie Polaków na kongresie brakuje osób, które są przez Federacje powołane do pełnienia funkcji Przewodniczących sesji (co zmusza do zmian programu), a nawet, jak to miało miejsce w Edynburgu — osób zaproszonych do wygłoszenia referatów.*

*Kongres IFIP w Toronto odbędzie się za niecałe trzy lata. Jeśli chcemy by reprezentacja Polski wypadła na nim godnie i odpowiednio do naszych ambicji, należy prawie bezzwłocznie przystąpić do przygotowań organizacyjnych i merytorycznych.*

**Prof. dr hab. Władysław M. Turski**

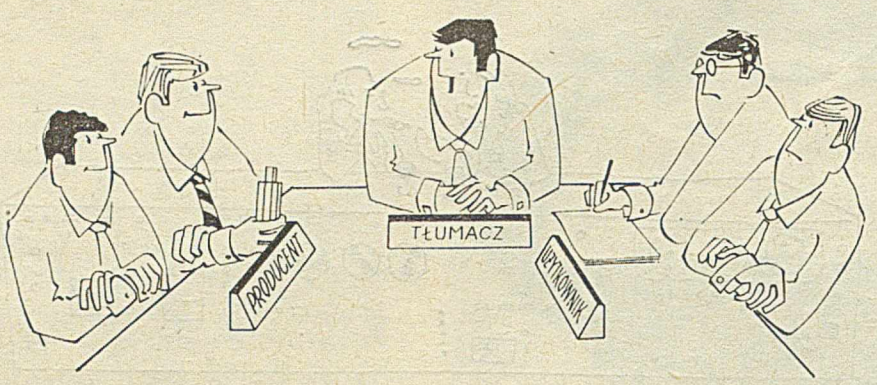
Jestem rad, że moja relacja z Kongresu IFIP 75 została przez prof. W. Turskiego tak uważnie przeczytana.

Jako sprawozdawca prasowy (a nie naukowy!) korzystałem z chwytanych na gorąco opinii naszych naukowców uczestniczących w imprezie. Moja rzekomo „apodyktyczna ocena referatów i referentów" była więc w głównej mierze oddaniem poglądów osób mogących uchodzić w głównej mierze za wyrazicieli naszej fachowej opinii publiczej. Chociaż jako dziennikarz osobiście uczestniczyłem zapewne w posiedzeniach innych sekcji tematycznych niż prof. Turski, to jednak odniosłem w całości tak samo pozytywne wrażenia, ale obowiązany byłem z tytułu swych funkcji również odnotować poglądy odmienne. Chciałbym podkreślić, że użyte przeze mnie sformułowanie „stosunkowo niski poziom merytoryczny części wygłoszonych referatów..." odzwierciedla w sposób bardzo umiarkowany poglądy niektórych polskich uczestników Kongresu, wygłaszane publicznie i często ze znaczną impulsywnością. Należy jednak podkreślić, że na każdej większej imprezie naukowej, zwłaszcza o zasięgu światowym, obok prezentacji doskonałych, pojawia się z reguły niemała liczba wystąpień wręcz słabych wynikających z tzw. dystrybucji geograficznej, nakazującej uwzględnienie również dorobku krajów rozwijających się.

Odrębną sprawą był brak na Kongresie referatów polskich, który powodował wśród naszych uczestników najwięcej rozgoryczenia. Tym bardziej, że na stosunkowo niedawno odbytym Kongresie Nauki Polskiej mówiło się wiele o wysokim poziomie naszych osiągnięć naukowych w dziedzinie informatyki, zwłaszcza w zakresie teorii. Stąd w zestawieniu z niektórymi słabymi referatami z innych krajów mogły u wielu osób powstać uzasadnione wtąpliwości, że „nie byliśmy obiektywnie oceniani". Przyczyniło się do tego również szeroko rozpowszechnione mniemanie, że liczba nadesłanych z Polski referatów była znaczna (w istocie nadesłano ich 10)

Jeżeli więc wyeliminowano wszystkie referaty polskie, to jest to znacznie lepiej, niż gdyby zastosowano do nas „taryfę ulgową", przewidzianą dla krajów rozwijających się! Chciałbym przy tej okazji dołączyć się do słusznego postulatu prof. Turskiego, aby dokonać analizy trybu krajowych przygotowań do Kongresu IFIP 74 i wykazać rzeczywiste przyczyny braku naszych referatów, jak również powstania innych przykrych niedociągnięć, jak np. sposób ostatecznego skompletowania składu polskiej delegacji. Sądzę, że analiza taka wskaże również sposoby skutecznego pobudzenia środowiska naukowego do reprezentowania naszego kraju na następnym Kongresie IFIP w Toronto zgodnie z jego rzeczywistą pozycją międzynarodową

**Władysław Klepacz**




# I Wojewódzka Konferencja Informatyków w Szczecinie

W dniach 21—22 stycznia br. odbyła się w Szczecinie I Wojewódzka Konferencja Informatyków, której podstawowym celem była ocena stopnia realizacji programu rozwoju informatyki w regionie szczecińskim oraz nakreślenie kierunków jej dalszego rozwoju przed VII Kongresem Techników Polskich i III Krajową Konferencją Informatyków. Miała ona również na celu integrację młodego przecież środowiska szczecińskich informatyków.

Organizatorami konferencji byli:

— Polski Komitet Automatycznego Przetwarzania Informacji przy Oddziale Wojewódzkim NOT w Szczecinie

— Urząd Wojewódzki w Szczecinie

— Zakład Organizacji Przetwarzania Danych Politechniki Szczecińskiej

— Zakład Elektronicznej Techniki Obliczeniowej Szczecin

— Zarząd Oddziału Towarzystwa Naukowego Organizacji i Kierowania w Szczecinie.

Patronat nad konferencją objęli i w obradach uczestniczyli

— prof. dr hab. inż. Andrzej Straszak — Przewodniczący ZG PKAPI

— Jerzy Kuczyński — Wojewoda Szczeciński

— prof. dr inż. Józef Kępiński — Przewodniczący OW NOT, Rektor Politechniki Szczecińskiej

— doc. dr hab. Zygmunt Zieliński — Prezes Oddziału Wojewódzkiego TNOiK

W czasie obrad wygłoszono 9 referatów (zamieszczonych w materiałach konferencyjnych), ujmujących trzy podstawowe grupy problemów:

— tendencje rozwoju gospodarczego regionu szczecińskiego

— oceny dotychczasowego rozwoju informatyki w regionie oraz perspektyw jej rozwoju widzianych od strony regionu, jak i od strony założeń krajowych

— teoretyczne i praktyczne podstawy systemów regionalnych.

Konferencja połączona była z prezentacją poszczególnych środowisk informatyki regionu. W ramach prezentacji uczestnicy konferencji mieli okazję zwiedzenia ośrodków obliczeniowych Szczecina, w tym ośrodka Stoczni im. A. Warskiego wyposażonego w komputer ICL System 4-52 oraz system kreślący KING MATIC DC 316/ /1830. Charakterystykę tych ośrodków podano w materiałach konferencyjnych.

Z imprez towarzyszących konferencji należy wymienić wystawę systemów opracowanych w regionie oraz cykl wykładów przedstawicieli Kombinatu Robotron z NRD.

Do uroczystych i miłych akcentów konferencji zaliczyć można moment otrzymania z rąk Wojewody Szczecińskiego odznaczeń „Gryf Pomorski" przez pionierów informatyki szczecińskiej.

Po szerokiej i wszechstronnej dyskusji przyjęto następujące wnioski konferencji:

1. W regionie szczecińskim pilną sprawą staje się opracowanie programu rozwoju informatyki na lata 1975—1980 i perspektywicznie do 1990 r. Wersja ta powinna uwzględniać dotychczasowy dorobek środowiska szczecińskiego, uchwały władz politycznych i państwowych w sprawie przyspieszonego rozwoju województwa szczecińskiego oraz rozwoju makroregionu nadmorskiego, a także założenia społeczno-gospodarczego rozwoju regionu do 1990 r. W opracowanym programie szczególna uwagę należy zwrócić na rozwój informatyki w następujących wiodących dla regionu działach:

— gospodarka morska

— kompleks żywnościowy

— przemysł maszynowy i elektrotechniczny

— nauka i rozwój

— transport

— energetyka

— budownictwo.

Opracowanemu programowi należy nadać odpowiednio wysoką rangę, a koordynację powierzyć specjalnemu ośrodkowi (INTORG-Szczecin). Ocenę realizacji programu powierzyć OW PKAPI, pełniącemu jednocześnie rolę zespołu konsultacyjnego władz regionu ds. rozwoju informatyki.

W programie należy zwrócić uwagę na narosłe dysproporcje w rozwoju informatyki. Szczególnie dotyczy to zastosowań w gospodarce morskiej (żegulga, port), gdzie dysproporcje w stosunku do regionu gdańskiego są zbyt rażące. Dlatego programy resortowe, dotyczące rozwoju informatyki w gospodarce morskiej powinny być konsultowane z władzami regionu, a do ich opracowania winni być włączeni specjaliści ze środowiska szczecińskiego.

2. W szerszym zakresie postuluje się włączać region szczeciński do budowy systemów rządowych, takich jak: SPIS, PESEL, CENPLAN i SINTO.

3. Niezbędne jest uznanie informatyki jako narzędzia usprawniającego działalność biblioteczno-informacyjną. Postuluje się uwzględnienie w programie rozwoju informatyki w regionie szczecińskim zastosowań do działalności biblioteczno-informacyjnej bibliotek naukowych i ośrodków informacji regionu szczecińskiego.

4. W ciągu najbliższej pięciolatki władze regionalne, zjednoczenia, przedsiębiorstwa powinny zintensyfikować działania zmierzające do wyrównania w podstawowych wskaźnikach ilościowych i jakościowych (kadry, sprzet. zastosowania) poziomu informatyki szczecińskiej do przynajmniej średniego poziomu krajowego. Władze resortowe — w resortowych programach rozwoju informatyki powinny bardziej uwzględniać potrzeby regionu szczecińskiego, zgodnie z jego rangą w skali gospodarki narodowej.

5. W zastosowaniach informatyki należy skoncentrować się na dziedzinach najbardziej efektywnych i najbardziej

Foto 1 Na sali obrad (od lewej: doc. dr hab. Zygmunt Zieliński — prezes TNOiK Szczecin, prof. dr hab. inż. Andrzej Straszak — przew. ZG PKAPI, prof. dr inż. Józef Kępiński — przew. NOT Szczecin, Jerzy Kuczyński Wojewoda Szczeciński)

opłacalnych. Należy zwrócić uwagę na konieczność opracowywania systemów powielarnych, sparametryzowanych, a dla ich twórców stosować odpowiedni system bodźców ekonomicznych.

6. Postuluje się pod adresem władz regionalnych i centralnych udzielenie pomocy w rozwiązywaniu spraw socjalno-bytowych, głównie nabrzmiałych problemów mieszkaniowych dla kadry specjalistycznej. Ze względu na szczególną i stale rosnącą rolę Międzyuczelnianego Instytutu Informatyki oraz ZETO Szczecin w regionie (ok. 40% potencjału kadrowego i sprzętowego), wybudowanie przynajmniej jednego wspólnego bloku mieszkalnego dla pracowników Instytutu i ZETO Szczecin. Wniosek ten kierowany jest pod adresem władz administracyjnych regionu (zapewnienie mocy przerobowych) i Ministerstwa Nauki, Szkolnictwa Wyższego i Techniki (zapewnienie środków finansowych). Przedsięwzięcie tego typu powinno zaspokoić potrzeby mieszkaniowe obecne, a także zapobiec niepokojącemu i znacznemu odpływowi kadry specjalistycznej do innych regionów, ewentualnie pozyskanie dla regionu deficytowych kadr elektroników, matematyków-numeryków, itd. z innych regionów.

7. Postuluje się konieczność przyspieszenia terminów oddawania do użytku budujących się ośrodków obliczeniowych (ZETO Szczecin). Należy także przystąpić w obecnej pięciolatce do budowy nowych ośrodków, przede wszystkim dla:

— Międzyuczelnianego Instytutu Informatyki

— handlu wewnętrznego

— Wojewódzkiego Urzędu Statystycznego

— głównych przedsiębiorstw gospodarki morskiej (wspólny ośrodek).

Brak ośrodków obliczeniowych stanowić może bowiem w przyszłości poważną barierę rozwoju informatyki.

8. Szczególnie sprzyjające warunki należy stworzyć rozwijającemu się szczecińskiemu ośrodkowi naukowemu informatyki w drodze zapewnienia udziału w rządowych i węzłowych tematach badawczych, zapewnienia nowoczesnego sprzętu informatycznego, udziału w różnych formach kształcenia kadr specjalistycznych na różnych kierunkach studiów. Szczególnie sprzyjające warunki w tym zakresie rysują się w kształceniu w Szczecinie specjalistów informatyków dla transportu. Postuluje się tu utworzenie na Wydziale Inżynieryjno-Ekonomicznym Transportu Politechniki Szczecińskiej nowego kierunku studiów: informatyka transportu o profilu inżynieryjno-ekonomicznym w specjalnościach: projektanci systemów i organizatorzy wdrażania.

9. W związku z zarysowującym się deficytem kadr informatyków postuluje się:

— zapewnić przyspieszoną realizację (w stosunku do programu) Międzyuczelnianego Instytutu Informatyki. Jest to niezbędna w tym zakresie pomoc resortu nauki

— powołać, obok istniejących, dalsze kierunki studiów informatycznych, jak: informatyka transportu, metody programowania, techniki laserowe w informatyce (optocyfronika)

— w celu doskonalenia zawodowego informatyków powołać na Politechnice Szczecińskiej studia podyplomowe i doktoranckie

— stworzyć warunki kształcenia średniej kadry w drodze powołania instytucji szkolenia zawodowego.

10. Należy przystąpić do opracowania w wybranych przedsiębiorstwach systemów kompleksowych, które pod względem rozwiązań stanowiłyby wzorce do naśladowania i mogły być powielarne.

11. Należy przystąpić do realizacji pilotowych systemów zarządzania regionem szczecińskim, po uprzednim opracowaniu koncepcji informatyzacji regionu. Docelowo należy dążyć do utworzenia banku informacji w regionie. Politechnika Szczecińska powinna podjąć odpowiednie badania w tym kierunku. Zagadnienia informatyzacji regionu powinny znaleźć odzwierciedlenie w prasie regionalnej i pismach naukowych („Przegląd Zachodniopomorski"). Ponadto widzi się celowość utworzenia biuletynu specjalistycznego wydawanego w ramach makroregionu przez PKAPI. W zagadnieniach regionalnych widzi się konieczność stworzenia sprawnej sieci transmisji danych oraz włączenie jej do sieci krajowej.

12. Postuluje się zorganizować w województwie szczecińskim klub użytkowników minikomputerów. Zadaniem klubu będzie poza wymianą doświadczeń w projektowaniu, programowaniu i eksploatacji systemów informatycznych, także podnoszenie kwalifikacji zawodowych użytkowników oraz wspólne dążenie do likwidacji usterek i awarii.

13. Należy opracowywać sukcesywnie zestawienia bibliograficzne eksploatowanych systemów w województwie, w celu poinformowania zainteresowanych przedsiębiorstw o stanie istniejących systemów, jak również uniknięcia dublowania w opracowywaniu nowych systemów.

Uczestnicy konferencji uznali za cenną inicjatywę zwołanie I Wojewódzkiej Konferencji Informatyków. Uważa się za potrzebne jej cykliczne powtarzanie przed każdą Krajową Konferencją Informatyków.

**Antoni Nowakowski**

Foto 2. Dekoracje „Gryfem Pomorskim" z rąk Wojewody Szczecińskiego (od lewej: mgr Edward Kolbusz, mgr Henryk Mielcarek, mgr inż. Henryk Wojdała)

# Narodziny statków pod dyktando komputera

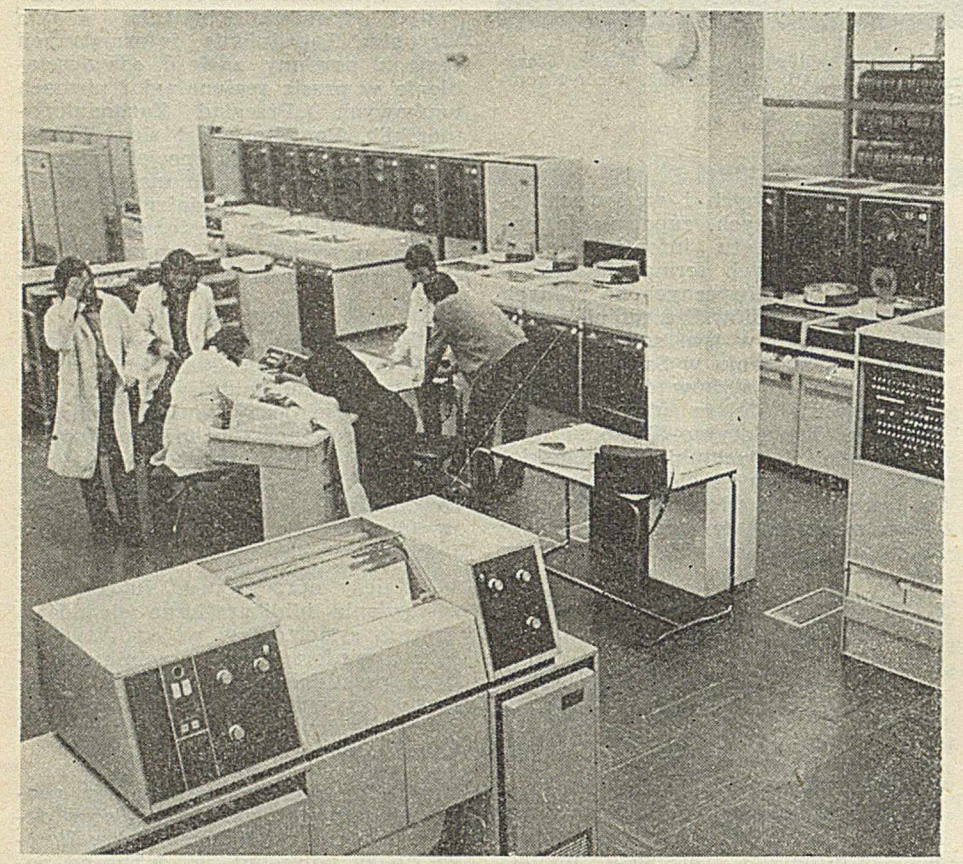
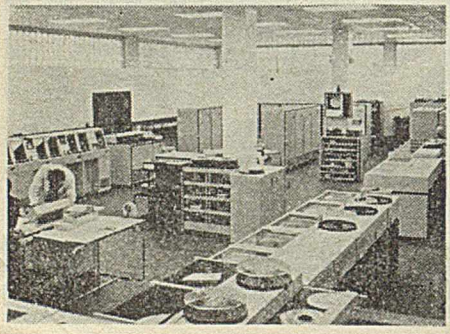


Foto 1 i 2. Zdecydowanie niepozorny z zewnątrz, zlokalizowany na terenie „sypialni Trójmiasta" — PRZYMORZU — ZIPO, widziany od wewnątrz zaskakuje znakomitą salą komputerów wybudowaną przez producenta znajdujących się w niej komputerów — firmę ICL. Dzięki postępom elektroniki, wielokroć potężniejsze komputery systemu 4 zajmują tyleż lub mniej miejsca co te z rodziny 1900. Zdjęcie 1 eksponuje przede wszystkim jednostki pamięci taśmowej i dyskowej oraz szybką drukarkę wierszową.

Zdjęcie 2 — jednostki centralne komputerów i czytnik taśmy dziurkowanej (z lewej).



Nie tylko z licznych doniesień prasy codziennej, lecz także ze strony informatyków mnożyły się liczne sygnały o sporych, uchwytnych sukcesach Zakładu Informatyki Przemysłu Okrętowego w Gdańsku w dziedzinie wykorzystania komputera w procesach technologicznych budowy statków.

Wypadałoby przyjrzeć się temu na miejscu, zwłaszcza że — jak przypuszczałem (i okazało się, że słusznie) — nie jest to jedyna działalność ZIPO na rzecz Zjednoczenia Przemysłu Okrętowego. Moc obliczeniowa będąca w dyspozycji Ośrodka ZIPO — to dwa nowoczesne komputery III generacji o łącznej pamięci operacyjnej 768 kB oraz zasłużony ELLIOTT 803, stanowiące razem wystarczający potencjał, aby zainteresować się bliżej pracą ZIPO.

Przewodnikiem po raczej niezbyt długich, ale zawiłych losach dzisiejszego ZIPO był mgr inż. Franciszek Thomas. W roku 1967 piastował obowiązki kierownika Branżowego Ośrodka Przetwarzania Informacji, związany z informatyką w przemyśle okrętowym niemal od początku, gdy w roku 1960 rozpoczynał swą pracę z 3-osobowym zespołem. W roku 1963 — już z 16-osobowym zespołem — przy Centralnym Biurze Konstrukcyjno-Badawczym rozpoczął eksploatację nowozakupionego ELLIOTTA 803.

Oczywiście zmiany w nazewnictwie, funkcjach, strukturze i przyporządkowaniach protoplastów ZIPO mogłyby stanowić pouczającą lekturę dla Czytelników zainteresowanych organizacją przedsiębiorstw. Jednak nie pozostałoby to bez wpływu na ramy reportażu, powiedzmy więc tylko, że początki informatyki w przemyśle okrętowym datują się od roku 1960, gdy jeszcze korzystano z ELLIOTTa w Instytucie Elektroniki w Międzylesiu. Nazwę, status i lokal otrzymało ZIPO w październiku roku 1971.

Warto tu jeszcze dodać, że wszystkie koleje losu informatyki w branży okrętowej były wypadkową przeobrażeń zachodzących wewnątrz całej branży. Tak więc zaczątki wiązały się z problemami, których CBKO a ściślej jego sekcja wytrzymałości, nie było w stanie dłużej rozwiązywać metodami konwencjonalnymi. Stąd — współpraca ze wspomnianym Instytutem w Międzylesiu, stąd — później, zakup takiego samego jak tam komputera ELLIOTT, dokładnie w momencie przejścia stoczni na małoseryjną produkcję statków o dużym tonażu. O ile bowiem dla długich serii

Foto 3. W miarę realizacji programu wyposażenia w sprzęt Zakładowych Ośrodków Przetwarzania Informacji, wzrastać będzie rola widocznej z lewej strony jednostki sterującej zdalnego dostępu. Na pierwszym planie czytnik taśmy dziurkowanej.

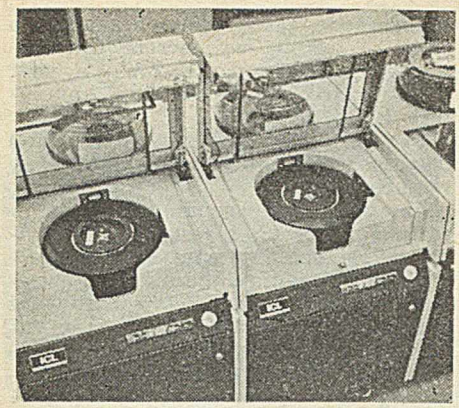
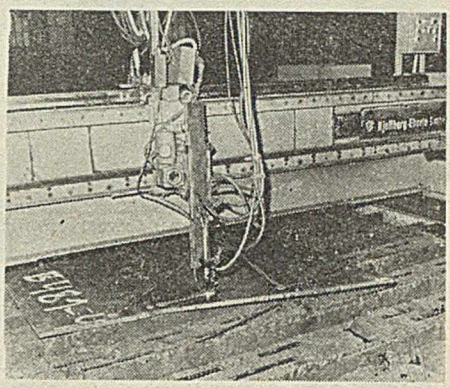


Foto 4. Jednostki pamięci dyskowej o pojemności 30 MB niebawem zostaną przekazane do Szczecina. Obecnie ZIPO dysponuje pięcioma jednostkami 30 MB, sześcioma o pojemności 8 MB i pięcioma o pojemności 60 MB.

obliczenia konstrukcyjne wykonuje się rzadziej — dla krótkich odwrotnie, zaś większe statki wymagają przeliczenia większej liczby segmentów.

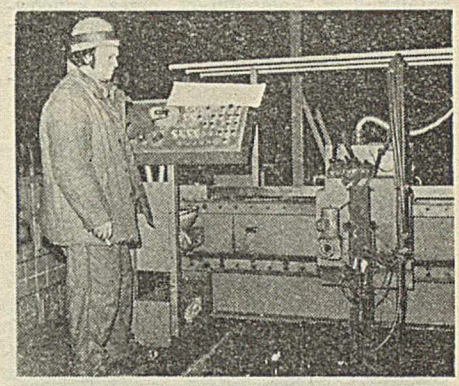
Początek dobrego okresu dla informatyki datuje się od roku 1967, gdy resort wydał dokument o rozwoju informatyki, precyzując obszar zastosowań informatyki: przetwarzanie danych do zarządzania, sterowanie numeryczne procesami technologicznymi i obliczenia konstrukcyjno-inżynierskie. Zatrudnienie wzrosło z 30 do 80 osób, w cztery lata później, gdy rodzaj, a zwłaszcza rozmiar prac przerósł możliwości ELLIOTTa, okazało się, że prace nad systemem ASTER wchodzą w okres finalny, usamodzielnione ZIPO otrzymuje komputer ICL 4/50; w półtora roku później — drugi o większej mocy obliczeniowej — ICL

Foto 6/7 Tu można stwierdzić naocznie przydatność komputera. Hala wydziału K-1 Gdańskiej Stoczni im. W. Lenina mieszcząca konwencjonalne stanowisko traserskie i 4 stanowiska do cięcia blach z urządzeniem sterowanym numerycznie dowodnie wskazuje na korzyści płynące ze stosowania komputera. Dawna, klasyczna metoda wymagała odwzorowania rysunku blach bezpośrednio na tworzywie i ręczne przycinanie blach palnikiem. System ASTER przygotowuje na taśmie papierowej zakodowane rozkazy dla sterowanego automatem palnika, który samoczynnie przykrawa blachy na poszycie kadłuba statku zgodnie z projektem. Jak zapewniano mnie na stoczni — pozwala to zaoszczędzić znacznie czas, poprawia dokładność cięcia i zapewnia istotne oszczędności materiału. Na foto 6 palnik sterowany numerycznie przycina blachę na foto 7 pracownik obsługujący urządzenie sterujące.



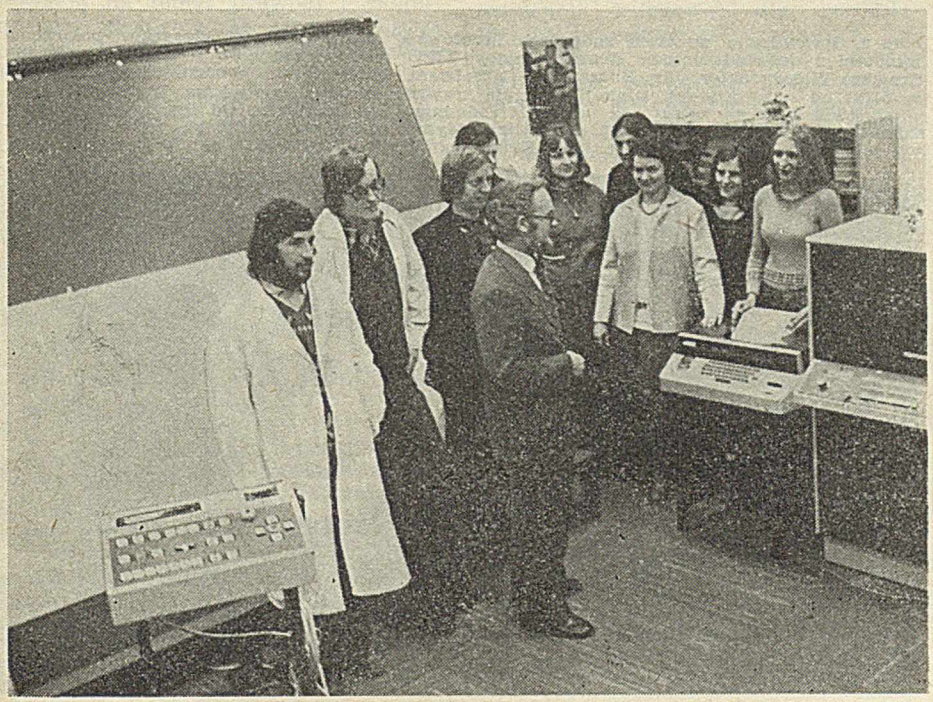
4/70. Rozwój parku maszynowego implikuje ilościowy i jakościowy wzrost kadry. W trzy miesiące po wizycie w maju 1975 r. — na park maszynowy ZIPO składają się 4 komputery w następujących konfiguracjach:

- ELLIOTT 803 — PAO 8 K słów, 3 jednostki pamięci taśmowej, czytnik kart 80-kolumnowych, 2 czytniki taśmy dziurkowanej, drukarka wierszowa, perforator taśmy papierowej, dalekopis.
- ICL 4/70 — PAO 512 kB, 5 jednostek pamięci taśmowej, 5 jednostek pamięci dyskowej o pojemności 60 MB każda, 3 jednostki po 8 MB, czytnik kart dziurkowanych, 2 czytniki taśmy dziurkowanej, drukarka 750 wierszy/min, drukarka 1350 wierszy/min, perforator taśmy papierowej, jednostka sterująca zdalny dostęp. Z komputerem aktualnie współpracuje w trybie zdalnego dostępu 10 urządzeń końcowych ekranowych oraz 10 dalekopisowych.
- ICL 4/50 — PAO 256 kB, 7 jednostek pamięci taśmowej, 3 jednostki pamięci dyskowej o pojemności 30 MB i trzy o pojemności 8 MB, dwie drukarki wierszowe, czytnik kart dziurkowanych, perforator taśmy papierowej, perforator kart, jednostka sterująca zdalny dostęp, 9 urządzeń końcowych ekranowych.
- ICL 2903 — PAO 48 kB, 4 jednostki pamięci taśmowej, 2 jednostki pamięci dyskowej o pojemności 60 MB, 2 urządzenia końcowe ekranowe. W skład wyposażenia ośrodka wchodzi również system kreślący norweskiej firmy KONGSBERG — DC 316/1830



z perforatorem taśmy papierowej, drukarką wierszową i stołem automatycznego kreślenia.

Foto 5. Od myśli do konkretnego wytworu. Zespół inż. Jana Juranka mgr Maria Najbar, mgr Maria Gałeczko, mgr Romualda Dojlido, mgr Anna Wróbel, mgr Anna Górska, mgr Mirosława Olejniczak, Wiesław Grzęda, Waldemar Przyłucki, Janusz Królikowski, opracował i wdrożył system ASTER, którego finalną funkcję pokazuje zdjęcie 6. Na zdjęciu — sala automatycznego kreślarza norweskiej firmy KONGSBERG — z prawej urządzenie sterujące.



Obsługę i eksploatację opisanego sprzętu wykonuje 300 osobowa załoga, której trzecia część zajmuje się projektowaniem i oprogramowaniem systemów. Ich liczba jest wystarczająco duża, aby nie ulec pokusie opisywania wszystkich. Należy więc skoncentrować się na najważniejszych.

System informatyczny zarządzania branżą SIZB, składający się z cząstkowych podsystemów opracowywanych dla poszczególnych dziedzin działalności i dla wszystkich jednostek organizacyjnych Zjednoczenia (Stocznia: gdańska, północna, gdyńska, szczecińska, przedsiębiorstwa wyposażenia okrętowego). Plan generalny obejmuje następujący kompleks podsystemów: zarządzanie, sterowanie numeryczne procesami technologicznymi budowy statków, projektowanie i konstruowanie mechanizmów i urządzeń okrętowych, projektowanie stoczni i zakładów. Te podsystemy realizowane są na dwóch poziomach: poziomie przedsiębiorstwa i poziomie branży. Zaspokajają one na poziomie

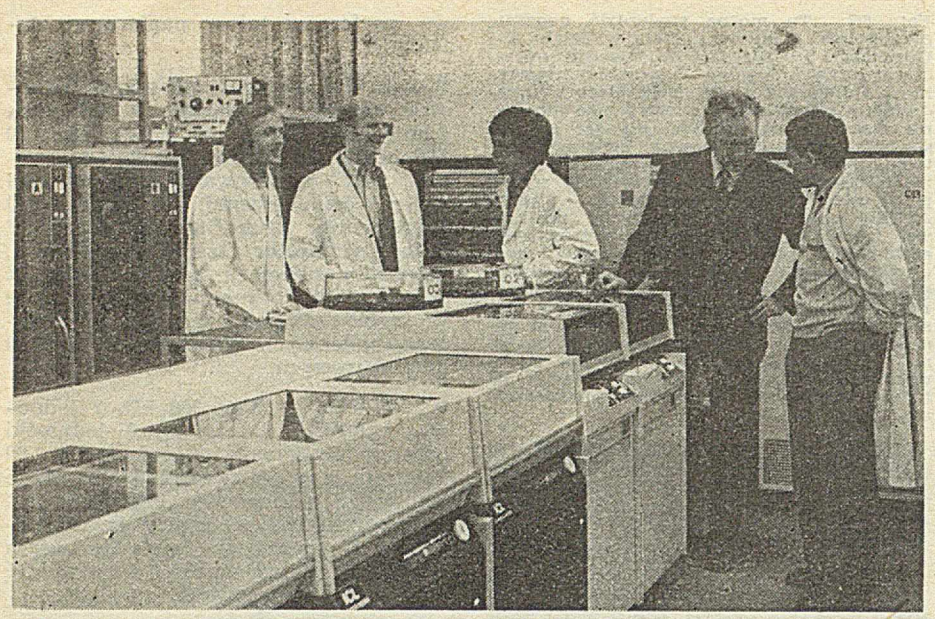


Foto 8. Zespół elektroników ZIPO. Od lewej główny inżynier Władysław Kostulak inż. Jan Dzierżanowski, inż. Andrzej Orszulak, mgr inż. Franciszek Thomas (szef inwestycji) inż. Henryk Richter.

Wymienione podsystemy pracują dla SIZB — metodą wyznaczoną funkcjonowaniem systemu TELZAR, a więc w zdalnym dostępie na bazie komputera ICL 4/50.

Nie wyczerpuje to oczywiście całej listy podsystemów uruchamianych w ramach SIZB.

Inna pokaźnia grupa podsystemów została wdrożona i jest eksploatowana w ramach Systemu Informatycznego Zarządzania Stoczniami. Poważnym osiągnięciem ZIPO jest również opracowanie i wdrożenie Systemu Informatycznego Sterowania Numerycznego ASTER. Aktualnie eksploatowane wydanie Systemu składa się z 11 programów i umożliwia generowanie kształtu kadłuba, odwzorowanie geometrii płatów konstrukcyjnych, generowanie linii konstrukcyjnych, konturowanie elementów konstrukcyjnych, rozwijanie płyt poszycia, standaryzację konstrukcji, opracowanie szablonów do gięcia płyt poszycia, odwzorowanie graficzne konturów, kompilacje kart wykrojów, program kreślący. System działa w oparciu o bazę danych w postaci zbiorów zawierających: analityczny zapis geometrii kadłuba, zbiór linii konstrukcyjnych i zbiór standardów. Dalszy rozwój systemu w dużym stopniu uwarunkowany jest dalszymi dostawami dla stoczni obrabiarek i urządzeń sterowanych numerycznie.

Rozmiar prac realizowanych przez ZIPO wskazuje na maksymalne wykorzystanie możliwości komputerów. Istotnie pracują one na trzy zmiany, ale nie jest pewne, czy to wystarczyłoby, gdyby nie wdrożenie i eksploatacja systemów operacyjnych ICL 4: system J *(Job)* i MJ *(Multi* Job) przy czym większe zastosowanie ma pierwszy — rządzący pracami wsadowymi (MJ przewidziany jest dla prac realizowanych w zdalnym dostępie), które jak dotąd stanowią znakomitą większość.

System operacyjny poziomu J jest dyskowym systemem operacyjnym umożliwiającym pracę wieloprogramo-

Foto 10. Zespół opracowujący system zdalnego dostępu dla celów informacji i zarządzania branżą TELZAR w trakcie komunikacji z komputerem przez końcówkę ekranową. Od lewej: inż. Ewaryst Nowak (kierownik zespołu) mgr Józef Matofi, mgr Mieczysław Antos, inż. Wiesław Patrzak, mgr Kazimierz Tworek, mgr Danuta Chludzińska.

- podsystem APIS 4 służący do wyszukiwania informacji naukowo-technicznej, ekonomicznej i patentowej
- podsystem APIS F — gromadzący i udostępniający informacje o aktualnie budowanych na świecie statkach
- podsystem K-71 — umożliwiający kalkulacje wynikowe kosztu budowy statków
- podsystem D-01 — zawierający kartotekę mechanizmów i urządzeń
- podsystem T-20 — umożliwiający kalkulację wstępną kosztu budowy statków
- podsystem P-90 — pozwalający sporządzać plany perspektywiczne
- podsystem M-61 Z — kontrolujący realizację zamówień
- podsystem P-95 — ujmujący wyniki produkcyjno-ekonomiczne branży.

pierwszym potrzeby informatyczne przedsiębiorstw i zabezpieczają dopływ danych do systemu, zaś na poziomie drugim — zaspokajają potrzeby informatyczne Centrali Zjednoczenia w oparciu o branżowe jednostki przetwarzania oraz dopływ danych z systemów przedsiębiorstw do jednostek przetwarzania Centrali.

Poszczególne podsystemy dzielone są na dziedziny tematyczne, a te z kolei są projektowane i wprowadzane do eksploatacji poprzez samodzielne jednostki informatyczne (jednostki przetwarzania danych, pakiety programów lub programy obliczeniowe).

Zakłada się, że moment opracowania wszystkich niezbędnych podsystemów i ich ostateczna integracja w Branżowy System Informatyczny Przemysłu Okrętowego przypadnie na rok 1980. Oceniając co już w tej materii zrobiono, należy mieć nadzieję, że jest to termin realny.

Jak dotychczas w ramach Systemu Informatycznego Zarządzania Branżą wdrożono i eksploatuje się:

Foto 9. Sprawdzona w wielu zadaniach „czołówka" ZIPO, stanowi klasyczne połączenie młodości z doświadczeniem. Reprezentujący ZIPO (pod nieobecność dyrektora mgr inż. Brewko) Wicedyrektor Jan Burzawa — absolwent Uniwersytetu Warszawskiego zaczynał w ZIPO od najprostszych funkcji, obecnie trzydziestolatek, przybył na Wybrzeże w towarzystwie czterech kolegów z uczelni i zadomowił się na dobre. Od lewej: mgr Bogdan Tomaszewski, dr Tadeusz Roth, mgr Alicja Myszor, Zdzisław Opaliński, mgr Jan Burzawa, mgr Zenon Chabielski, mgr Józef Kruk, mgr Marian Zieliński.

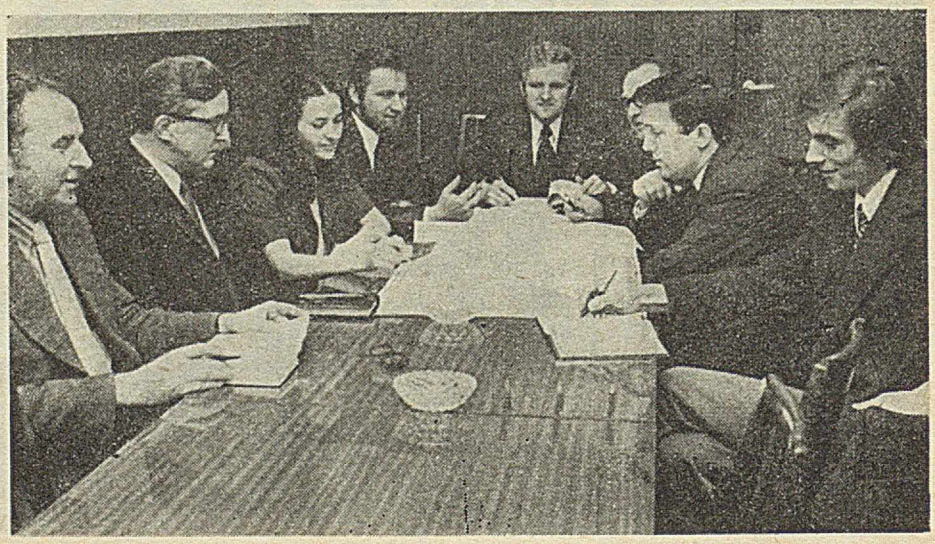


Foto 11. Zespół konserwacji i eksploatacji systemu operacyjnego: mgr Mieczysław Myszor, mgr Krzysztof Kwiek.

wą (maksimum 14 programów), dynamiczne planowanie przetwarzania i szybkie testowanie programów. Umożliwia programowanie w wielu językach. Stosowany do komputerów ICL system 4 (40, 50, 70) wymaga minimum 65 kB pamięci operacyjnej, jednostkę wymiennej pamięci dyskowej oraz podstawowe urządzenia WE/WY, czytnik taśmy dziurkowanej, czytnik kart dziurkowanych, drukarkę wierszowa, perforator taśmy.
Jak zwykle, tak i w tym przypadku nie sposób pominąć perspektywy rozwoju informatyki w branży okrętowej. Są one dyktowane rozwojem mocy wytwórczej stoczni i przedsiębiorstw wyposażenia okrętowego. Potrzeby floty krajowej i wyścig do zamówień zagranicznych powodują, że czas zaczyna się liczyć podwójnie. Armatorzy żądają coraz to większych jednostek. Te — tworzyć bez wprowadzenia nowoczesnych rozwiązań konstrukcyjnych i technologicznych — nie sposób.

To też informatycy z pewnością nie mogą być tu traktowani jako nieszkodliwi maniacy. Zakłada się, że jeszcze w tym roku należy wyposażyć pięć z pośród sześciu istniejących Zakładowych Ośrodków Przetwarzania Informacji w komputery, a więc poza Stocznią Szczecińską — posiadającą już komputer ICL-4-52, Stocznie: gdańska, północna oraz gdański HYDROSTER i bydgoski FAMOR powinny uczynić ze swoich ZOPI mocnych współpartnerów ZIPO.

Przewiduje się dalszy rozwój Branżowego Systemu Informatycznego dla potrzeb zarządzania i systemów przedsiębiorstw, a także konsekwentny rozwój systemów do numerycznego sterowania procesami technologicznymi i automatyzację prac projektowo-konstrukcyjnych.

Oceniając to co dotychczas już zrobiono można mieć nadzieję, że oczywisty wzrost zadań stojących przed przemysłem okrętowym nie zaskoczy informatyków spod znaku ZIPO.

Foto 12. Zamieszczając zdjęcie mgr Krystyny Kowalewskiej — absolwentki SGPiS, redakcja czyni to nie tylko z zamiarem fetowania Międzynarodowego Roku Kobiet, ale także z uwagi na jej odpowiedzialną funkcję głównego technologa EPD.

**Krystyn Bernatowicz**

foto M. Biernacki (CAF)

## Komunikat

W dniu 27 lutego 1975 r. przy współudziale Instytutu Problematyki Przestępczości, reprezentowanego przez Dyrektora, prof. dr hab. Brunona Hołysta odbyła się w Instytucie Matematyki Uniwersytetu Łódzkiego konferencja na temat prawnej i ekonomicznej problematyki rozwoju informatyki. Podstawą dyskusji były tezy referatów:
— mgr inż. A. Bossowskiego na temat problematyki zabezpieczenia danych, oraz
— mgr K. Prochaski i dr J. M. Szymańskiego na temat całokształtu problematyki.
W referacie, wygłoszonym przez drugiego ze współautorów poszerzono m. in. problematykę motywacji i warunków podejmowania decyzji o komputeryzacji, zagadnienia pracy w ośrodkach obliczeniowych, projektowanie systemów, problematykę gospodarności, zabezpieczenia danych i programów przed zniszczeniem, przestępczości komputerowej, zagadnienie wątpliwej celowości stosowania EMC do banalnych obliczeń rachunkowości gospodarczej za cenę znacznie przekraczającą koszt odpowiedniej pracy księgowych. W konkluzji sformułowano tezy stwierdzające konieczność:

- poprawy regulacji prawnej informatyki m.in. w zakresie przepisów inwestycyjnych, projektowania, cenników i warunków umów
- poprawy informacji w zakresie niezbędnym do podejmowania decyzji o wprowadzaniu SEPD;
- dokonania oceny przez naczelne organy państwowe i polityczne dotychczasowej praktyki zatrudnienia ośrodków obliczeniowych prostą pracą w zakresie sporządzania tabulogramów ewidencji księgowej dla przedsiębiorstw nieskomputeryzowanych, ewentualnie zorganizowanie dozoru naukowego nad pracą gospodarczych ośrodków obliczeniowych, analogicznie do istniejącej organizacji dozoru technicznego w przemyśle
- rozszerzenia programów szkolenia kadry Min. Sprawiedliwości, MSW, GKA, NRA, ZPP o tematykę: podstaw informatyki, problematyki prawno-ekonomicznej informatyki, informatyki prawniczej
- powołania organizacji rzeczoznawców w zakresie specjalistycznych zagadnień informatyki
- zachęcenia specjalistów od dogmatyki prawa cywilnego administracyjnego i karnego do zajęcia się problematyką prawną informatyki.

Wśród zabierających głos wystąpili: przedstawiciel Min. Sprawiedliwości Sędzia S. Chromicki, kierownik Państwowego Biura Notarialnego w Łodzi Notariusz S. Kurpiel, ekspert od organizacji pracy dr A. Wojciechowski, mgr J. Brzozowski, Dyr. II Oddz. NBP mgr B. Lelonkiewicz.

W toku dyskusji poparto tezy referatów, wyjaśniając jednocześnie pewne nieporozumienia i wskazując na trudności uregulowania prawnego działalności gospodarczej w zakresie informatyki. Zdaniem dyskutantów wprowadzanie informatyki jest procesem nieodwracalnym. Retrospektywnie należy stwierdzić, że postęp ilościowy w zakresie komputeryzacji w Polsce w ostatnich latach jest imponujący, należy więc zwrócić uwagę na aspekty jakościowe; w szczególności podniesiono następujące problemy:

- komputeryzacja jest akcją organizacyjną, wymagającą specjalistycznego przygotowania, tymczasem nie kształci się u nas kadry organizatorów, specjalistów od wprowadzania informatyki
- konieczne jest, aby przedstawiciele dogmatyki prawa cywilnego, administracyjnego i karnego zajęli się badaniem problematyki prawnej informatyki
- istnieje oczywista potrzeba wprowadzania informatyki w dziedzinie pracy, dysponujemy również kadrą zdolną do opracowywania systemów informatyki prawniczej. Tymczasem zarówno na uczelniach, jak i wśród praktyków prawa, informatyka oczekuje dopiero na zajęcie się nią. W tym celu należy:

— wprowadzić wzgl. rozszerzyć program studiów z zakresu informatyki

dla studentów prawa i adminnistracji — zaapelować do Zrzeszenia Prawników Polskich o przeprowadzenie akcji popularyzującej informatykę wśród praktyków prawa

— dążyć do integracji prawników-informatyków, zajmujących się opracowywaniem teoretycznych i praktycznych problemów informatyki. Prawnicy-informatycy pojawili się w Polsce, podobnie jak w innych krajach, na skutek praktycznej nierealności żądania od matematyków opanowania problematyki prawniczej, a bez metod matematycznych, zwłaszcza statystyki matematycznej, nie można sobie wyobrazić postępu w zakresie programów informatyki prawniczej. Prawnicy-informatycy pracują w zasadzie poza uczelniami i w rozproszeniu, co powoduje m.in. dublowanie prac. (kr)

## Informatycy nawiązują sąsiedzką współpracę

Coraz więcej toruńskich przedsiębiorstw pasługuje się nowoczesnym narzędziem organizacji zarządzania, planowania, sterowania procesami technologicznymi itp. — jakim jest informatyka. Instaluje się maszyny matematyczne, przeszkala lub zatrudnia fachowców z tej dziedziny. Zrodziła się myśl, aby zespolić wysiłki, stworzyć warunki do wymiany doświadczeń i doskonalenia kwalifikacji oraz upowszechniania problematyki w zakładach, które dopiero zamierzają to nowe narzędzie stosować.

Z inicjatywy OR NOT powołana została komisja informatyki, której zarząd tworzą: mgr Ryszard Piotrak z MERINOTEXU, mgr inż. Szczepan Holc z APATORA, mgr Waldemar Modrzyński z Instytutu Maszyn Matematycznych i mgr inż. Zbigniew Mówka z CHEMOAUTOMATYKI. Zainteresowanie inicjatywą okazały również: GEOFIZYKA, PP Cukrownie Kujawskie, Zakład Energetyczny, TOWIMOR, ELANA i TORAL.

Działalność swą komisja zamierza adresować do pracowników zakładowych komórek informatyki (wymiana doświadczeń) oraz do kadry kierowniczej i techniczno-ekonomicznej przedsiębiorstw (kształtowanie poglądów i pobudzanie do rozwijania informatyki). Nowo powołana komisja postawiła sobie również za cel: aktywny udział w przygotowaniach do tegorocznej Krajowej Konferencji Informatyków. (kr)

## Ośrodek Obliczeń Elektronicznych dla Stoczni w Ustce

W stoczni w Ustce rozpoczęto budowę obiektu, w którym znajdzie pomieszczenie Zakładowy Ośrodek Obliczeń Elektronicznych. Wyposażony on zostanie w maszynę cyfrową, która przetwarzać będzie dane w zakresie planowania produkcji, indeksu materiałów, zużycia materiałów, zamówień i ich realizacji oraz elementów kooperacyjnych. Ośrodek oddany zostanie do użytku w listopadzie br. (kr)

## Komunikat

Centrum Obliczeniowe PAN organizuje w dniach 24—26 września 1975 roku w Warszawie ogólnopolskie sympozjum n.t.:

„Organizacja maszyn cyfrowych i mikroprogramowanie"

W programie sympozjum przewiduje się następujące działy tematyczne:

- Architektura i organizacja logiczna maszyn i systemów cyfrowych
- Symulacja i modelowanie struktury maszyn cyfrowych
- Zagadnienia mikroprogramowego sterowania w maszynach i systemach cyfrowych
- Projektowanie oprogramowania i systemów operacyjnych
- Problemy niezawodności, testowania i naprawialności maszyn i systemów cyfrowych
- Automatyzacja projektowania sprzętu i oprogramowania.

Będzie to trzecie krajowe sympozjum poświęcone zagadnieniom organizacji maszyn cyfrowych. Celem konferencji jest dokonanie przeglądu prac prowadzonych w tej dziedzinie w kraju i umożliwienie szerokiej wymiany doświadczeń.

## Magazyn z komputerem

W Płockiej Fabryce Maszyn Żniwnych opracowano projekt elektronicznego systemu przetwarzania informacji dla potrzeb gospodarki magazynowej. System ten umożliwia prowadzenie racjonalnej gospodarki materiałowej i szybkie bilansowanie potrzeb produkcji z aktualnym stanem zapasów. Komputer zastąpi skomplikowany i pracochłonny system ewidencjonowania części oparty na drukach informacyjnych i tablicach dyspozycyjnych. Dane zawarte w pamięci komputera ułatwią pracę także zaopatrzeniowcom i kierownictwu fabryki. (kr)

## Informacje o normach z komputera

Możliwość szybkiego uzyskania potrzebnej informacji jest obecnie jednym z bardzo ważnych czynników rozwoju gospodarki a zwłaszcza nauki i techniki. Zapewnieniu tej możliwości służą m.in. rozbudowywane coraz bardziej systemy informatyczne krajowe i międzynarodowe. W Centralnym Ośrodku Badawczo-Rozwojowym Normalizacji w Warszawie opracowano ostatnio specjalistyczny komputerowy system informacyjny w zakresie normalizacji. W przyszłości powiązany on będzie z międzynarodowym systemem informacji o normach krajów RWPG i europejskim systemem ISO. Bank informacji tego systemu zawierać będzie charakterystyki około 0,5 mln dokumentów krajowych i zagranicznych, takich jak normy i ich projekty, zlecenia normalizacyjne, plany i dokumenty dotyczące unifikacji i typizacji wyrobów, różnego typu publikacji z tego zakresu itp.

Obecnie — w ramach uruchomionego podsystemu pilotowego wprowadzono do komputerowego „banku danych" informacje dotyczące wszystkich polskich norm z działu pod nazwą „przyrządy i przybory pomiarowe". (kr)

## Komunikat

Polski Komitet Automatycznego Przetwarzania Informacji — NOT w porozumieniu z Zarządem Głównym SARP, w związku z III Krajową Konferencją Informatyków organizuje:

### PRZEGLĄD PROJEKTÓW OBIEKTÓW OŚRODKÓW OBLICZENIOWYCH

Celem przeglądu jest wyłonienie szczególnie dobrych projektów i przedstawienie ich przyszłym inwestorom.

Dla najlepszych rozwiązań projektowych przewidziane są nagrody Ministra Nauki Szkolnictwa Wyższego i Techniki

| | |
|---|---|
| I nagroda | 50.000,— zł |
| II nagroda | 30.000,— zł |
| III nagroda | 20.000,— zł |

które przyznane zostaną na wniosek komisji powołanej przez Prezydium PKAPI w porozumieniu z ZG SARP.

**Założenia:**

Projektowane ośrodki informatyczne różnią się w wyniku odmiennych programów użytkowych.

Różnice wynikają głównie z następujących powodów:

— różna ilość i różne typy komputerów
— różna ilość i różne typy urządzeń do przygotowania danych
— odmienne zakresy i organizacja funkcji przygotowania systemów informatycznych (projektowanie i programowanie).

Z uwagi na fakt, że najczęściej występują ośrodki w których program zakłada instalowanie 2—3 komputerów (ODRA-1305, IS 1030) budynek ośrodka dla takich instalacji jest przedmiotem najpowszechniejszego zainteresowania inwestorów.

**Kryteria oceny projektów:**

— walory funkcjonalne
— walory estetyczne
— walory technologiczne (łatwość realizacji, zastosowanie elementów typowych, zastosowanie prefabrykatów)
— powtarzalność (możliwość wielokrotnego wykorzystania projektu lub jego części).

**Uczestnicy:**

W przeglądzie uczestniczyć mogą osoby fizyczne i zespoły osób oraz projektujące jednostki gospodarki uspołecznionej, zgłaszające do organizatora projekt techniczny obiektu bez względu na etap jego realizacji (projekt, projekt w realizacji, projekt zrealizowany).

**Termin:**

Projekty należy nadsyłać w terminie do 15 sierpnia 1975 r. na adres ZETO Wrocław ul. Ofiar Oświęcimskich 7/13.

Ogłoszenie wyników przeglądu nastąpi w czasie III Krajowej Konferencji Informatyków w październiku 1975 r.

# Analiza parku komputerowego zainstalowanego w Austrii

Całościowa analiza parku komputerowego polega na wykryciu wszystkich prawidłowości dotyczących jego rozwoju, ich interpretowaniu oraz na wyprowadzaniu użytecznych wniosków o jego dotychczasowym przebiegu i tendencjach na przyszłość. Stawia się tutaj tezę, że analiza taka powinna być przeprowadzona na tle rozwoju gospodarczego i społecznego danego kraju. Rozwój ten determinuje bowiem możliwość własnej produkcji maszyn cyfrowych, ich import oraz profil zastosowań, a także stopień ich wykorzystania. Znajomość zachodzących procesów gospodarczych umożliwia ocenę prawidłowości rozwoju parku komputerowego. Analiza tego ostatniego powinna bowiem odpowiedzieć na następujące pytania:

- Jaki jest jego skład i struktura oraz czy są one ekonomicznie uzasadnione?

Chodzi tu między innymi o ustalenie, czy:

— liczba komputerów znajdujących się w danym kraju zaspakaja potrzeby jego gospodarki

— prawidłowa jest struktura maszyn cyfrowych np. z punktu widzenia ich wielkości, zastosowań czy też ich pochodzenia (produkcja własna, import).

- W jakich działach gospodarki narodowej komputery są instalowane? Ważne jest tutaj stwierdzenie, czy komputeryzowane są działy, które decydują o rozwoju danego kraju.

- Do jakich celów komputery służą i w jakim stopniu są wykorzystywane?

- Jakie zachodzą zmiany ilościowe i jakościowe w odniesieniu do omawianego parku maszyn cyfrowych?

Zakres niniejszego artykułu jest poważnie zawężony w stosunku do przedstawionej powyżej problematyki. Skupiono uwagę jedynie na ilościowej charakterystyce składu i struktury komputerów uniwersalnych [1] zainstalowanych w Austrii wg stanu na 1.07.1973 r. i 1.07.1974 r. z punktu widzenia ich producentów oraz mocy obliczeniowej maszyn. Zakres ten został zdeterminowany dostępnością danych dotyczących interesującego nas problemu.

Ze względu na pierwsze z wymienionych kryteriów klasyfikacyjnych ustalono:

— liczbę komputerów pochodzących z importu

— ich podział na wyprodukowane w Europie Zachodniej i w USA.

Umożliwiło to określenie stopnia zależności rozwoju parku maszyn cyfrowych znajdujących się w Austrii od zagranicy. Uwzględniając drugie kryterium podano proporcje zachodzące pomiędzy liczbą komputerów o małej, średniej, dużej i bardzo dużej mocy obliczeniowej. Łączne rozpatrywanie obu wymienionych kryteriów klasyfikacyjnych pozwoliło na scharakteryzowanie każdego producenta, którego maszyny cyfrowe znajdują się w Austrii, z punktu widzenia klas wielkości komputerów. Dokonano również charakterystyki każdej z tych klas z punktu widzenia kształtowania ich liczebności przez poszczególnych producentów.

Ta część analizy miała charakter statyczny. Analiza dynamiczna która obejmowała okres dwóch lat pozwoliła natomiast określić:

- roczne tempo wzrostu w procentach liczby komputerów w wyszczególnionych poprzednio przekrojach.
- zmiany strukturalne, jakie miały miejsce na skutek nierównomiernego wzrostu poszczególnych grup komputerów.

Niniejsza analiza opiera się na opublikowanej po raz pierwszy w znanej periodycznej publikacji Diebold Management Report [2] statystyce dotyczącej komputerów uniwersalnych zainstalowanych w Austrii. Statystyka ta jest wynikiem badań firmy Diebold Parisini GmbH z Wiednia. Wynika z niej, że w Austrii, której ludność wynosi 7,5 mln, wg stanu na 1.07.1974 r. zainstalowanych było 755 komputerów. Oznacza to, że jedna maszyna cyfrowa przypada na 9801 mieszkańców co wskazuje na wysoki stopień komputeryzacji. Wspomniana liczba 755 komputerów pochodzi całkowicie z importu. Obejmuje ona 60 różnych modeli wyprodukowanych przez 6 firm amerykańskich i 7 firm zachodnioeuropejskich. Udział tych pierwszych w całym parku wynosi 53,3% liczby modeli oraz aż 71,3% ogólnej liczby komputerów. Wynika stąd, że struktura parku komputerów amerykańskich jest bardziej jednorodna niż w przypadku maszyn produkcji europejskiej.

Jak wynika z tabeli I, w okresie od 1.07.1973 r. do 1.07.1974 r. omawiany park zwiększył swoją liczebność o 96 komputerów, z których 53 zakupiono od producentów amerykańskich. Roczne tempo wzrostu liczby komputerów wyniosło więc 14,5%. Tempo wzrostu maszyn produkcji amerykańskiej było niższe od przeciętnego i ukształtowało się na poziomie 10,9%.

Konsekwencją było zmniejszenie udziału tych komputerów w ogólnej ich liczbie z 73,6% do 71,3%. Natomiast roczne tempo instalacji komputerów europejskich na rynku austriackim było wyższe od przecitęnego i ukształtowało się na poziomie 24,7%. Udział liczby tych komputerów zwiększył sięi z 26,4% w 1973 r. do 28,7% w 1974 r.

Interesująca jest również analiza rynku komputerowego z punktu widzenia mocy obliczeniowej komputerów. Podzielono je w omawianej statystyce na 4 kategorie (małe, średnie, duże i bardzo duże). W analizie wyodrębniono modele, które stanowią podstawową część wymienionych kategorii. Najczęściej stosowanymi w Austrii komputerami małej mocy obliczeniowej są maszyny serii IBM System 3 (179 egzemplarzy). Stanowią one 34,5% ogólnej liczby maszyn tej grupy. Popularne w tej grupie są również komputery: IBM 360/20 (50 szt.), IBM 1130/ /1620 (36 szt.), NIXDORF 880 (34 szt.),

[1] Komputery do zastosowań gospodarczych oraz naukowo-technicznych nie obejmują więc komputerów do sterowania procesami.

[2] Diebold Management Raport, September 1974, Wyd. Diebold Deutschaland GmbH, Frankfurt/Main

Tabela I

Charakterystyka parku komputerów uniwersalnych zainstalowanych w Austrii wg źródeł pochodzenia*)

| Wyszczególnienie | Liczba komputerów w sztukach wg stanu na | | Procentowy udział liczby komputerów wg stanu na | | Liczba modeli komputerów wg stanu na | |
|---|---|---|---|---|---|---|
| | 1.07. 1973 | 1.07. 1974 | 1.07. 1973 | 1.07. 1974 | 1.07. 1973 | 1.07. 1974 |
| komputery ogółem | 659 | 755 | 100,0 | 100,0 | 54 | 60 |
| z tego: | | | | | | |
| produkcji USA | 485 | 538 | 73,6 | 71,3 | 29 | 32 |
| produkcji zachodnioeurop. | 174 | 217 | 26,4 | 28,7 | 25 | 28 |

*) obliczenia własne na podstawie zestawień zawartych w cytowanym Diebold Management Report.

HONEYWELL BULL 561/58 (29 szt.), UNIVAC U 9300 I/II (22 szt.), oraz U 9200 (21 szt.). Podane w nawiasach liczebności dotyczą stanu na dzień 1.07.1974 r.

Komputery średniej mocy obliczeniowej reprezentowane są przede wszystkim przez modele: IBM 360/30 (31 szt.), IBM 360/25 (17 szt.), UNIVAC U 9400 (15 szt.) oraz IBM 370/125 (10 szt.). Komputery dużej mocy obliczeniowej w większości dotyczą następujących modeli: IBM 370/145 (25 szt.), IBM 370/135 (25 szt.), IBM 360/49/44 (22 szt.).

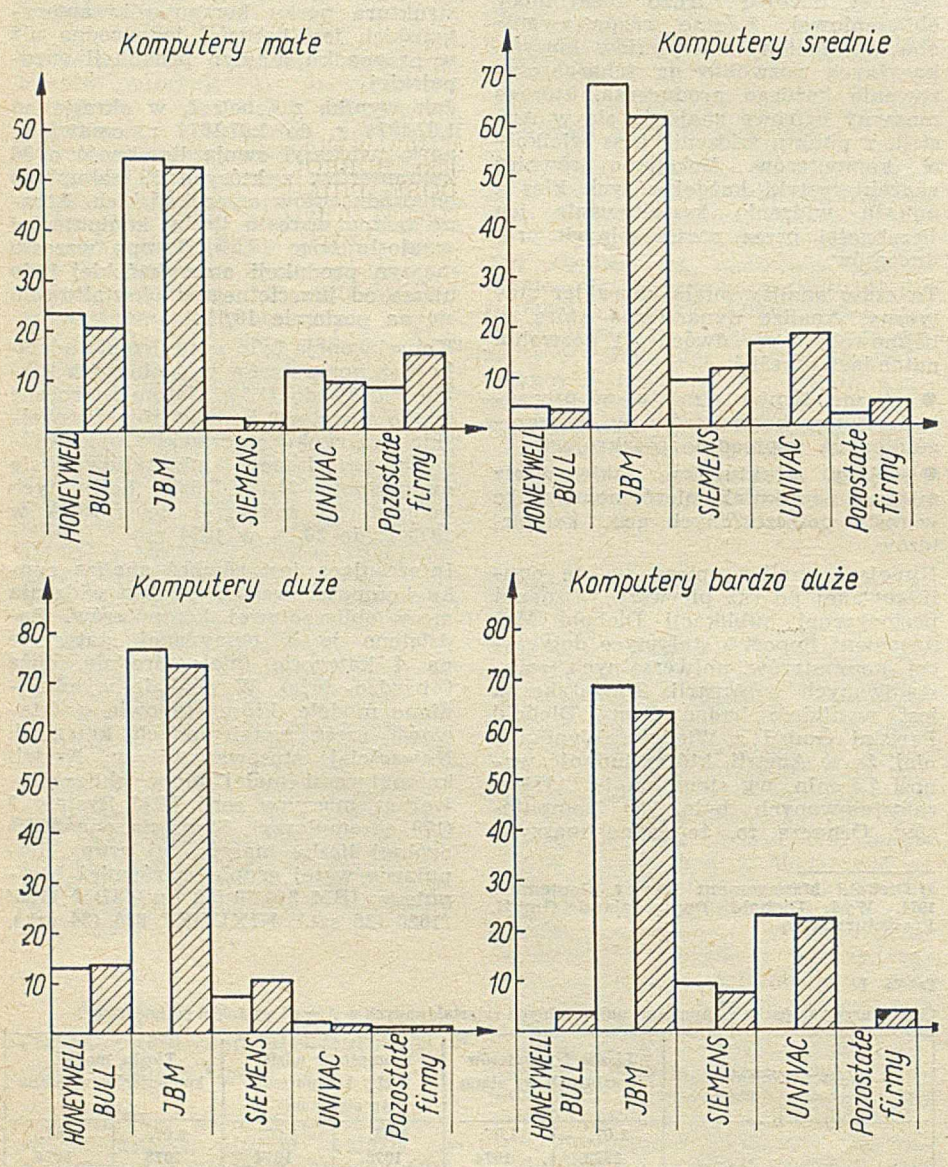
Ostatnią grupę wielkości stanowią komputery bardzo dużej mocy obliczeniowej. Do najczęściej stosowanych należą modele: IBM 370/155/158 (17 szt.) i UNIVAC U 1106 I/II (4 szt.). Sumaryczne liczebności oraz procentowy udział wymienionych kategorii wielkości komputerów uniwersalnych zawiera tabela II. Z tabeli tej wynika, że zdecydowaną przewagę mają komputery małe, natomiast procentowy udział komputerów średnich i dużych jest podobny i znacznie wyższy od tego, jaki osiągnęły maszyny bardzo duże. Również największy przyrost dotyczył komputerów małych. W omawianym okresie wyniósł on 78 instalacji stanowił ponad 81% ogólnej liczby nowo zainstalowanych komputerów. Również tempo wzrostu tych komputerów było wyższe od średniego i ukształtowało się na poziomie 17,7%, a ich procentowy udział wzrósł do 68,7%. Także tempo wzrostu liczby komputerów bardzo dużych było wyższe od przeciętnego i wynosiło 22,7%, należy jednak zauważyć, że odnosi się to do małych liczb bezwzględnych (wzrost z 22 do 27 instalacji). Ich udział w globalnej liczbie maszyn cyfrowych także uległ niewielkiemu zwiększeniu z 3,4 do 3,6%. Natomiast tempo wzrostu w kategoriach maszyn średnich i dużych było niższe od przeciętnego, a ich udział w globalnej liczbie maszyn zmalał. Tempo wzrostu maszyn średnich wyniosło mianowicie 7,2% a ich udział spadł z 14,7 do 13,8%. W kategorii komputerów dużych wskaźnik wzrostu wyniósł 6,1% a udział procentowy z 15,0% zmalał do 13,9%.

Tabela II

Charakterystyka parku komputerów uniwersalnych w Austrii (z punktu widzenia ich wielkości)*)

| Kategoria mocy obliczeniowej komputerów | Liczba komputerów w sztukach wg stanu na | | Procentowy udział liczby komputerów | |
|---|---|---|---|---|
| | 1.07.73 | 1.07.74 | 1.07.73 | 1.07.74 |
| małe | 441 | 519 | 66,9 | 68,7 |
| średnie | 97 | 104 | 14,7 | 13,8 |
| duże | 99 | 105 | 15,0 | 13,9 |
| b. duże | 22 | 27 | 3,4 | 3,6 |
| Ogółem | 659 | 755 | 100,0 | 100,0 |

*) Obliczenia własne na podstawie op. cit.




Rys. 1. Udział głównych producentów w kształtowaniu liczebności poszczególnych klas komputerów

Obecnie omówiony zostanie udział głównych producentów komputerów w kształtowaniu rynku maszyn cyfrowych w Austrii. Dane na ten temat zawiera tabela III.

Wynika z niej, że rynek komputerów w Austrii kształtują przede wszystkim 2 firmy amerykańskie — IBM i UNIVAC, oraz 2 firmy zachodnioeuropejskie — HONEYWELL BULL oraz SIEMENS.

Komputery ich produkcji stanowią łącznie 88,6% ilości i 94% ogólnej wartości parku maszyn cyfrowych.

W okresie od 1.07.1973 r. do 1.07.1974 r. uległ zmniejszeniu procentowy udział liczby zainstalowanych komputerów produkcji trzech głównych firm, a mianowicie IBM, HONEYWELL BULL i UNIVAC. W przypadku IBM mamy nawet do czynienia ze spadkiem dość znacznym, bo wynoszącym aż 2,8%.

W dalszym jednak ciągu pozycja tej firmy jest dominująca, zarówno pod względem liczby zainstalowanych komputerów, jak też ich wartości (57% ilości oraz 70% globalnej wartości wszystkich zainstalowanych w Austrii komputerów). Wśród komputerów wyprodukowanych przez IBM najbardziej popularne w Austrii są małe maszyny serii SYSTEM 3 (pierwsze instalacje w 1970 r.). W okresie od 1.07.1973 do 1.07.1974 r. nastąpił przyrost liczby tych maszyn o 45 egzemplarzy i obecnie stanowią one 41,5% wszystkich maszyn

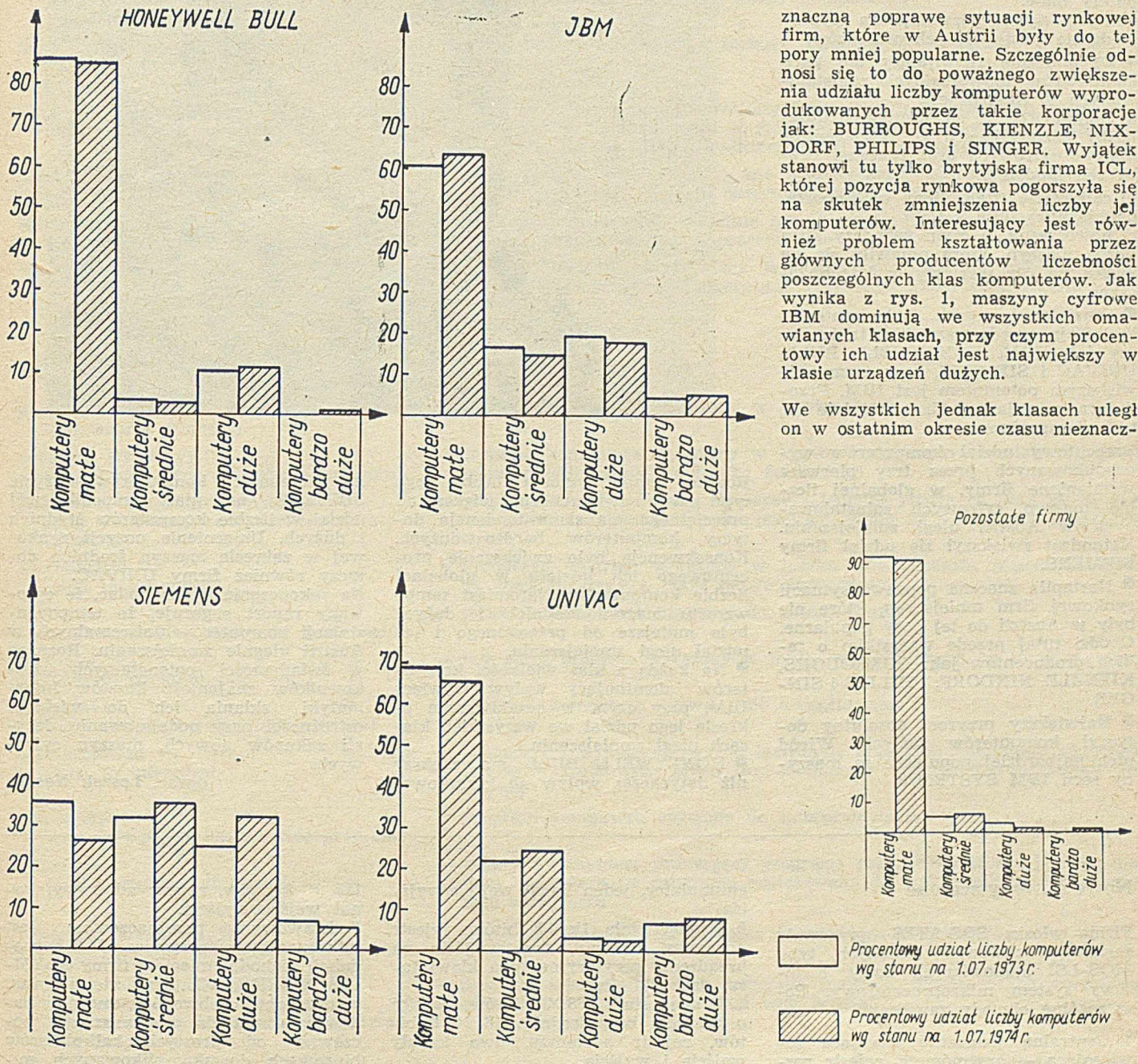
IBM, zainstalowanych w tym kraju. Jest to najwyższy wskaźnik wzrostu wśród wszystkich maszyn. Również serię IBM 370 charakteryzują tendencje wzrostowe. Udział tych maszyn w ogólnej liczbie komputerów firmy IBM stanowi 17,8%, a w ciągu omawianego okresu nastąpił ich przyrost o 18,4% (pierwsze dostawy komputerów tej serii nastąpiły w 1971 r.). Należy zaznaczyć, że **liczba komputerów** najpopularniejszej dotąd serii IBM 360 uległa w ciągu ostatniego roku zmniejszeniu o 14 maszyn i obecnie wynosi 133 ma wydajność czterokrotnie większą w porównaniu z modelem 360/50. Zastępowane są one bardziej wydajnymi **komputerami serii** IBM 370, z których przykładowo modelem 370/155. Najbardziej popularne z serii IBM 360 są małe komputery model 360/20. Liczba ich uległa jednak zmniejszeniu i wynosi obecnie tylko 50 instalacji. Odnotować należy również

**Tabela III**

Udział głównych producentów komputerów w kształtowaniu rynku maszyn cyfrowych w Austrii w 1974 r.*)

| Producent | Liczba zainstalowanych komputerów wg stanu na dzień 1.07.1974 r. | | | | | Procentowy udział liczby komputerów | Procentowy udział wartości komputerów |
|---|---|---|---|---|---|---|---|
| | małych | średnich | dużych | bardzo dużych | ogółem | | |
| 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 |
| IBM**) | 273 | 64 | 77 | 17 | 431 | 57,0 | 70,0 |
| HONEYWELL**) | | | | | | | |
| BULL | 107 | 4 | 14 | 1 | 126 | 16,7 | 9,0 |
| UNIVAC | 51 | 19 | 2 | 6 | 78 | 10,3 | 9,0 |
| SIEMENS | 9 | 12 | 11 | 2 | 34 | 4,6 | 6,0 |
| Pozostali | 79 | 5 | 1 | 1 | 86 | 11,4 | 6,0 |
| Razem | 519 | 104 | 105 | 27 | 755 | 100,0 | 100,0 |

**) Dane liczbowe nie potwierdzone przez producenta
*) Wielkości zawarte w kolumnie 8 pochodzą z op.cit.
Wielkości w kolumnach 2—7 zostały ustalone na podstawie obliczeń własnych




Rys. 2. Struktura zainstalowanych w Austrii komputerów w podziale wg głównych ich producentów

znaczną poprawę sytuacji rynkowej firm, które w Austrii były do tej pory mniej popularne. Szczególnie odnosi się to do poważnego zwiększenia udziału liczby komputerów wyprodukowanych przez takie korporacje jak: BURROUGHS, KIENZLE, NIXDORF, PHILIPS i SINGER. Wyjątek stanowi tu tylko brytyjska firma ICL, której pozycja rynkowa pogorszyła się na skutek zmniejszenia liczby jej komputerów. Interesujący jest również problem kształtowania przez głównych producentów liczebności poszczególnych klas komputerów. Jak wynika z rys. 1, maszyny cyfrowe IBM dominują we wszystkich omawianych klasach, przy czym procentowy ich udział jest największy w klasie urządzeń dużych.

We wszystkich jednak klasach uległ on w ostatnim okresie czasu nieznacz-

nemu zmniejszeniu. Jednocześnie umocniła się pozycja rynkowa firm SIEMENS i UNIVAC w klasie komputerów średnich. SIEMENS zwiększył również swój procentowy udział w globalnej liczbie komputerów dużych To samo odnosi się do firmy HONEYWELL BULL.

Reasumując stwierdzamy, że pozycja firmy IBM we wszystkich klasach wielkości komputerów jest niezagrożona. W dziedzinie komputerów dużych i bardzo dużych ostro z nią konkurują HONEYWELL BULL oraz SIEMENS. Pozostałe firmy odgrywają najważniejszą rolę w kształtowaniu liczby komputerów małych. W ostatnim czasie zaczęły się one również liczyć w dziedzinie komputerów średnich i bardzo dużych.

Ostatnim punktem analizy jest charakterystyka głównych producentów z punktu widzenia klas wielkości wytwarzanych przez nich komputerów. Z rys. 2 wynika, że wszyscy producenci, z wyjątkiem SIEMENSA sprzedają w Austrii przede wszystkim komputery małe.

Fakty powyższe pozwalają na wyprowadzenie następujących wniosków:

● Dynamicznie rozwijający się park komputerów uniwersalnych zainstalowanych w Austrii jest:

a) niejednorodny, co wynika z dużej liczby kształtujących go producentów maszyn cyfrowych oraz znacznej liczby ich modeli

b) całkowicie uzależniony od importu, przede wszystkim z USA

c) nowoczesny, o czym świadczy duża liczba maszyn, które pojawiły się na rynku po raz pierwszy w latach 70-tych.

● Omawiany park komputerów kształtują przede wszystkim takie firmy jak: IBM, HONEYWELL BULL, UNIVAC i SIEMENS, przy czym największym potentatem jest IBM. Przodująca pozycja tej firmy wydaje się być niezagrożona.

Procentowy udział komputerów wyprodukowanych przez trzy pierwsze wymienione firmy, w globalnej liczbie maszyn cyfrowych zainstalowanych w Austrii uległ zmniejszeniu. Natomiast zwiększył się udział firmy SIEMENS.

● Nastąpiła znaczna poprawa sytuacji rynkowej firm mniejszych, które nie były w Austrii do tej pory popularne. Chodzi tutaj przede wszystkim o takich producentów jak: BURROUGHS, KIENZLE, NIXDORF, PHILIPS i SINGER.

● Największy przyrost absolutny dotyczył komputerów małych. Wśród nich najbardziej popularne są maszyny serii IBM SYSTEM/3.

Roczne tempo wzrostu liczby tego typu maszyn było znacznie większe od przeciętnego. Ta sama tendencja dotyczy komputerów bardzo dużych. Konsekwencją było zwiększenie procentowego ich udziału w globalnej liczbie komputerów. Natomiast tempo wzrostu maszyn średnich i dużych było mniejsze od przeciętnego i ich udział uległ zmniejszeniu.

● Na każdą z klas wielkości komputerów dominujący wpływ wywiera IBM, przy czym w oczekiwanym okresie jego udział we wszystkich klasach uległ zmniejszeniu.

● HONEYWELL BULL ma większy niż dotychczas wpływ na kształtowanie liczebności komputerów dużych. SIEMENS natomiast umocnił swój udział w liczbie komputerów średnich i dużych. Umocnienie pozycji rynkowej w zakresie maszyn średnich dotyczy również firmy UNIVAC.

Na zakończenie warto podać, że cytowany raport sugeruje, że tempo instalacji komputerów uniwersalnych w Austrii ulegnie zmniejszeniu. Rosnąca w świadomości potencjalnych użytkowników znajomość kosztów informatyki skłania ich do większej ostrożności przy podejmowaniu decyzji zakupów nowych, maszyn cyfrowych.

oprac. **Leszek Nowak**

## Nowe obwody scalone

Firma włoska SGS/ATES opracowała zestaw 4 obwodów scalonych typu MOS-LSI stanowiących nowy, 8 bitowy system mikroprocesorowy. Poszczególne obwody zawierają:

1. Centralną jednostkę sterującą zawierającą — arytmometr, rejestr rozkazów, 48 rejestrów uniwersalnych, akumulator, jeden kanał wejścia/wyjścia.
2. Pamięć stałą 1k × 8 bitów, rejestr adresowy stosowy, dwa wyjścia na urządzenia peryferyjne typu klawiatura, drukarka itp.
3. Pamięć stałą 768 × 8 bitów, pamięć o dostępie bezpośrednim 18 × 8 bitów, rejestr adresowy, dwa kanały wejścia i wyjścia.
4. Pamięć o dostępie bezpośrednim 128 × 8 bitów, rejestr adresowy, kanał wejścia/wyjścia.

Podstawowym przeznaczeniem jest mikroprocesorowy system CP-3-F produkcji zachodniemieckiej firmy OLYMPIA WERKE AG. Przy zastosowaniu wymienionych obwodów można budować urządzenia przetwarzające począwszy od prostych kalkulatorów biurowych do skomplikowanych minikomputerów. (lj)

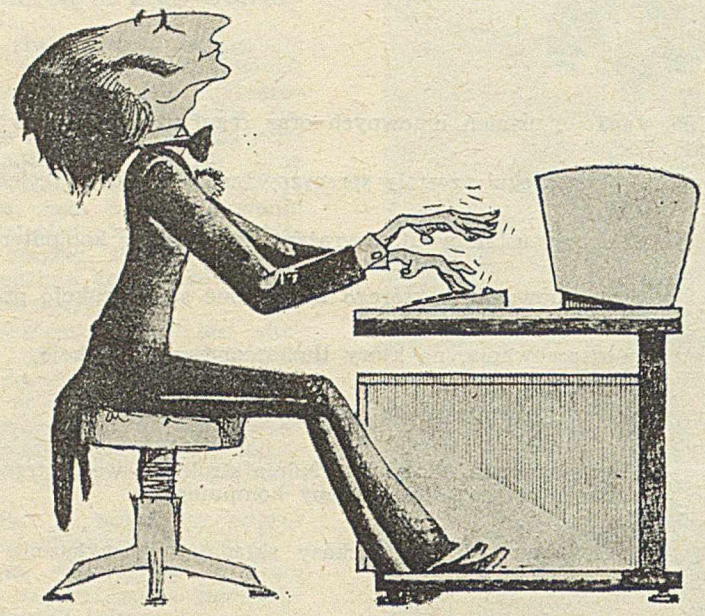
## PROGRAMOWANIE — JĘZYKI grupa 07.02

| | |
|---|---|
| język (ang. *language*) | — zbiór znaków, ustaleń umownych oraz reguł używanych do przenoszenia informacji |
| język sztuczny (ang. *artificial language*) | — język, którego reguły zostały sprecyzowane przed rozpoczęciem jego użytkowania |
| język programowania (ang. *programming language*) | — język sztuczny, ustalony dla wyrażania programów komputerowych |
| język źródłowy (ang. *source language*) | — język programowania, z którego tłumaczone są instrukcje programu |
| język wynikowy, język docelowy (ang. *target language, object language*) | — język programowania, na który tłumaczone są instrukcje |
| język algorytmiczny (ang. *algorythmic language*) | — język sztuczny, ustalony dla wyrażania algorytmów |
| język maszynowy, język niskiego szczebla (ang. *computer-oriented language low-level language*) | — język programowania, który uwzględnia strukturę wewnątrzną określonego komputera lub określonej grupy komputerów |
| język komputera, język maszyny (ang. *computer language, machine language*) | — język maszynowy, którego rozkazy składają się wyłącznie z rozkazów komputera |
| język symboliczny, język assemblerowy (ang. *assembley language*) | — język maszynowy, którego rozkazy korespondują z rozkazami komputera zazwyczaj w stosunku jeden do jednego i który może zawierać takie udogodnienia, jak np. użycie makroinstrukcji (makrorozkazów) |
| język wysokiego szczebla (ang. *high-level language*) | — język programowania, który nie odzwierciedla struktury wewnętrznej określonego komputera lub określonej grupy komputerów |
| język problemowy (ang. *problem-oriented language*) | — język programowania, specjalnie przystosowany do określonej klasy problemów. Przykłady: języki proceduralne |
| język proceduralny (ang. *procedure-oriented language, procedural language*) | — język problemowy ułatwiający wyrażanie procedury w postaci jednoznacznie sprecyzowanego algorytmu. Przykłady: FORTRAN, ALGOL, COBOL, PL/1 |

## PROGRAMOWANIE — METODY TRANSAKCJI grupa 07.03

| | |
|---|---|
| program źródłowy (ang. *source program*) | — program komputera wyrażony w języku źródłowym |
| program wynikowy (ang. *target program, object program*) | — program komputera wyrażony w języku wynikowym (docelowym), który został przetłumaczony z języka źródłowego |
| translator, program tłumaczący (ang. *translator, translater, translating program*) | — program komputera, który tłumaczy z jednego języka na inny język, a w szczególności z jednego języka programowania na inny język programowania |
| asemblować (ang. *to assemble*) | — tłumaczyć program wyrażony w języku symbolicznym (assemblerowym) na język maszyny a także łączyć podprogramy |
| asembler (ang. *assembler, assembly program*) | — program komputera używany do asemblowania |
| kompilować (ang. *to compile*) | — tłumaczyć program wyrażony w języku programowym na język maszyny |
| kompilator (ang. *compiler, compiling program*) | — program komputera używany do kompilowania |
| interpretować (ang. *to interpret*) | — tłumaczyć i wykonywać każdą instrukcję programu komputera określoną w języku źródłowym przed tłumaczeniem i wykonaniem instrukcji następnej |
| interpreter (ang. *interpreter, interpretive program*) | — program komputera używany do interpretowania |
| symulator (ang. *computer simulator*) | — program komputera tłumaczący programy przygotowania dla jednego modelu komputera na programy dostosowane do wykonania na innym modelu komputera |
| generator kompilatorów (ang. *compiler generator*) | — translator lub interpeter używany do tworzenia kompilatorów |
| generator makroinstrukcji, generator makrorozkazów (ang. *macro generator, macro generating program*) | — program komputera, który zastępuje makroinstrukcje języka źródłowego określoną sekwencją instrukcji tego języka. |

**Władysław Klepacz**



*Możliwości tworzenia sztuki za pomocą komputerów traktowane są przez ludzi ze środowiska informatycznego z dużą dozą pobłażania. Uznajemy to zastosowanie maszyn cyfrowych za jedną z marginesowych ciekawostek, powstających w chwilach odprężenia; będącą raczej zabawą niż poważną działalnością.*

*Okazało się jednak, że sztuka komputerowa w ciągu ostatnich lat zrobiła wiele i osiągnęła obecnie poziom, który nie pozwala na jej dalsze lekceważenie. Co więcej, dziedzina ta stała się, jak twierdzą jej kreatorzy, jedną z rokujących najpoważniejsze nadzieje w zastępowaniu ludzi przez maszyny w czynnościach czysto intelektualnych.*

*Jak przedstawia się rzeczywista sytuacja sztuki komputerowej? Jaką rolę odgrywały i mogą odgrywać maszyny w tworzeniu muzyki, plastyki, literatury czy filmu? Próbując dać odpowiedź na te pytania, poprosiliśmy mgr inż. Marka Hołyńskiego, jednego z redaktorów INFORMATYKI o zorientowanie się w temacie i przygotowanie czegoś na kształt ogólnego raportu o stanie sztuki komputerowej.*
*W paru numerach naszego pisma mamy zamiar przedstawić wyniki owego raportu. Jednocześnie zapraszamy do współpracy wszystkich Czytelników zainteresowanych tymi problemami, posiadających wszelkie związane z tematem dokumenty lub dzieła sztuki komputerowej.*

*Będziemy je publikować w dziale SZTUKA KOMPUTEROWA, którego utrzymanie na dłużej, uzależniamy właśnie od nadsyłanych materiałów.*

*Redakcja*

**Muzyka była pierwszą dziedziną sztuki, do której wkroczyły komputery i w której, przynajmniej na razie, odnoszą największe sukcesy. Wpływ techniki na muzykę jest zresztą oczywisty, tak jak uchwytny jest dla każdego laika dystans między wierzbową fujarką, a organami elektrycznymi. Kompozytorzy i wykonawcy z zaufaniem i nadzieją spoglądali zawsze w kierunku techniki, chętnie przystając na pronowane przez nią eksperymenty. Ponadto dokonywała ona pewnych nieodwracalnych przeobrażeń i powrót do uboższej techniki ograniczał sztukę. Naturalnie, i w tej regule są wyjątki, ale przecież nie sposób oddać pełnego brzmienia utworu Chopina zastępując fortepian klawesynem.**

Wydaje się przy tym, że ze wszystkich rodzajów sztuki muzyka jest tym, który łatwiej trafia do umysłów zaporządtnięty ch sprawami nauki i techniki. Wybitni matematycy twierdzą, że odbierając muzykę czują się podobnie jak wówczas, gdy intensywnie pracują nad rozwiązaniem jakiegoś problemu. Wśród naukowców spotykamy wielu melomanów próbujących nawet własnych sił w tej dyscyplinie. Na swoim ulubionym instrumencie — skrzypcach — chętnie grywał Einstein. Niemiecki fizyk Herman von Helmholtz twórca opisu zasady zachowania energii, określił ową intuicyjną bliskość nauk ścisłych i muzyki następująco: „Matematyka i muzyka! Najbardziej rażące z możliwych przeciwieństw ludzkiej myśli! A jednak powiązane, odbijające się w sobie wzajemnie!"

Panująca między muzyką i techniką harmonia została zakłócona wskutek potężnego skoku techniki w ostatnich dziesiątkach lat. Należałoby przypuszczać, że pozostająca coraz bardziej w tyle muzyka nigdy już nie nawiąże kontaktu z najnowszymi odkryciami, bo i cóż mogłaby mieć wspólnego z rakietą kosmiczną, laserem czy komputerem? Puszczając wodze fantazji rozważano co prawda możliwości związania tych wynalazków z muzyką, były to jednak wizje nieśmiałe, ograniczające się do czynczynności prozaicznych i traktowane jako niezbyt poważne ciekawostki. Jeśli chodzi o komputery, to projektowano tylko powierzenie im prac, do których przywykły: różnych obliczeń, zbierania i przechowywania informacji mniej lub więcej z muzyką związanych, ale dla istoty tej sztuki raczej marginalnych.

**Początek nawet został już zrobiony. Zaprzęgnięto je do zbierania danych o historii muzyki, o kompozytorach, termianach imprez itp. Biblioteki przy uczelniach muzycznych — idąc śla-**

**dem innych szkół wyższych — powierzały maszynom pieczę nad swoimi księgozbiorami, zmuszone zalewem bieżących informacji, lawiną nowych książek i artykułów.** Komputery wzięły na siebie rolę rejestratorów i klasyfikatorów; na żądanie znajdowały potrzebne materiały tematyczne szeregując je według hierarchii ważności, uwzględniając objętość tekstów, czy datę publikacji. Podobne funkcje zaczęły spełniać w taśmotekach i archiwach radiowych — zbliżając się coraz bardziej do samej muzyki. **Okazały się też wielce przydatne w obliczaniu danych technicznych dla instrumentów, sal koncertowych i aparatury elektroakustycznej.** Przykład: znakomity system głośników BOSE 901, którego model powstał przez symulowanie w komputerze idealnej czaszy głośnikowej. Autor pomysłu, amerykański profesor elektroakustyki Harald Bose, ustalił także za pomocą maszyny optymalną proporcję między dźwiękiem odbitym i bezpośrednim trafiającym do słuchacza. Wyniki wykorzystał do takiego rozstawienia głośników w kolumnie, by audytorium miało wrażenie, że przebywa w sali koncertowej.

Powodzenie zatwierdziło awans maszyn o szczebel wyżej. Użyto ich przy analizowaniu utworów muzycznych. Do statystycznych obliczeń liczby taktów w utworach tego samego rodzaju, do oceny najczęściej spotykanych cech dźwięku: czasu jego trwania okresu narastania i zanikania, wysokości, natężenia czy barwy. Dzięki tym badaniom pogłębiono dotychczasową wiedzę o „fizyce" muzyki, a także uzyskano przesłanki dla precyzyjnych analiz muzykologicznych. Statystyka ułatwia wychwytywanie typowych, powtarzających się zależności. Zestawienia robione przez komputer pozwalały zatem ustalać cechy charakterystyczne dzieł danego kompozytora, bądź też kompozytorów należących do jednej szkoły lub okresu historycznego. Na podstawie obliczeń i porównań można było określić wykonawców grających tę samą melodię. Ustalanie zbiorów cech stało się podstawą do identyfikacji anonimowych kompozycji dawnych mistrzów — albo oceny doskonałości naśladowców.

**Na tym jednak kończą się podobieństwa między losami komputerów w muzyce i technice. Okazało się bowiem, że lepiej zejść z utartego lecz prowadzącego przez peryferie szlaku, gdyż istnieje prostsza droga. Wiodła ona przez elektryczną i elektroniczną aparautrę muzyczną, nad której wykorzystaniem myśleli awangardowi kompozytorzy od początku naszego wieku. To właśnie oni otworzyli komputerom główną bramę fortecy, jaką stanowił do niedawna świat dźwięków; trudno się zatem dziwić, że przestały one szturmować boczne umocnienia.**

Twórcy muzyki nigdy nie rezygnowali z żadnego ze środków mogących wzbogacić ich sztukę, łowili nowe dźwięki i szukali odmiennych sposobów wykonania, sprawniejszych metod kompozycji i zapisu. Jakże by więc mogli przegapić szansę jaką dawał im gwałtowny rozwój techniki? Szczególną uwagę zwrócili na te wynalazki, które wynikały z zastosowania energii elektrycznej.

Telegraf, telefon i radio przełamały niepokonaną — zdawałoby się — dotychczas barierę odległości zamykającą dźwięk w promieniu kilkunastu metrów. Gramofon i magnetofon zaiściły marzenie pokoleń o zapisie idealnym, który przenosiłby niemożliwe do sformalizowania niuanse kompozycji i finezyjne detale interpretacji. Przestało odtąd obowiązywać aktualne przez tysiąc trzysta lat powiedzenie świętego Izydora z Sewilli: „Dźwięki giną bo są niepowtarzalne". Kolecjonowanie dźwięków na celuloidowej płycie lub magnetycznej taśmie umożliwiło natychmiastowe odtworzenie ich dowolnego fragmentu, dokonywanie zmian, wzmacnianie, łączenie i przesuwanie w czasie. I, co równie ważne, dało to nareszcie okazję do pozbycia się kłopotliwego pośrednictwa wykonawcy w dialogu kompozytor — słuchacz. Twórca w zaciszu własnej pracowni mógł doprowadzić nagranie dzieła do końca, do postaci dokładnie takiej, w jakiej powinno zostać odebrane. Jednocześnie ogromnie wzrosło zapotrzebowanie na muzykę. Wytwórnie płytowe, radio i telewizja zalały świat muzyką, wzbudziły głód wciąż nowych dźwięków, dźwięk stał się wszechobecny.

**Elektroniczny sprzęt muzyczny pojawił się we właściwym momencie. Kompozytorom przestały wystarczać dawne kanony — zaczęli wprowadzać dysonanse, stonalności, formy nieokreślone, rytmy asymetryczne. Włączyli do swych dzieł skrzypienie drzwi i stukot maszyny do pisania. W obowiązującej skali dźwiękowej — oktawie rozdzielonej na dwanaście półtonów — muzyka awangardowa nie znajdowała możliwości utrwalenia się. Kompozytorzy zastosowali wówczas własne notacje, najczęściej w formie diagramów opatrzonych komentarzami. Ogromnym zmianom zaczęła ulegać technika wykonania, której pozwolono na znacznie więcej swobody (na wdzieranie się np. do wnętrza fortepianu). W tej rewii nowości nie można było dopuścić, by unikalne zalety urządzeń elektronicznych pozostały niezauważone.**

Zrozumiał to Edgar Varese, Francuz przebywający od 1915 roku w Stanach Zjednoczonych. Muzyk z wykształcenia — absolwent paryskiego konserwatorium, inżynier z rodzinnej tradycji — niejako dziedzicznie obciążony ideą połączenia muzyki z techniką — jest autorem hasła, które wzbudziło tyleż uznania, co zaciekłych sprzeciwów: „Sztuka musi dotrzymać kroku nauce".

Varese dochowywał wierności tej zasadzie. W Nowym Jorku urządził sobie odpowiednie studio i przystąpił do komponowania muzyki posługując się metodami stosowanymi dotychczas przez fizyków i matematyków. „Muzyka jest to dźwięk zorganizowany" — twierdził Varese, tytułując siebie samego „rzemieślnikiem rytmu, częstotliwości i natężenia". Przez kilkadziesiąt lat nie mógł jednak tą drogą osiągnąć rezultatów, które pretendowałyby do miana prawdziwej sztuki. Wreszcie w 1954 roku odniósł zwycięstwo: utwór „Deserta" zawdzięcza wiele fragmentów elektronice. Sukces zachęcił Varese'a do dalszych eksperymentów nad konstrukcją elektronicznych zestawów, wydających sygnały, których nie udałoby się w żaden inny sposób powołać do życia. „Potrzebny nam jest instrument zdolny do tworzenia ciągłego dźwięku w dowolnej tonacji. Żeby zaś zbudować taki instrument, konieczna jest współpraca kompozytora i elektronika".

**W niespełna wiek po odkryciu przez Edisona fonografu muzyka elektroniczna, traktowana w latach dwudziestych jako zabawna ciekawostka, zyskała sobie pełne prawa obywatelskie. Niemal każdy z twórców młodego pokolenia jest autorem kompozycji elektronicznych lub takich, które godzą urządzenia elektroniczne z instrumentami konwencjonalnymi. Pracownie dźwięku elektronicznego znajdują się już prawie we wszystkich konserwatoriach, a w wielu średnich szkołach muzycznych na Zachodzie muzyka elektroniczna stanowi odrębny przedmiot nauczania.**

Drugą — po magnetofonowej — rewolucję w muzyce wywołało zbudowanie przenośnych zestawów do syntezy dźwięku zdolnych do bezpośredniej współpracy z komputerami. Zestawy te, zwane popularnie syntezatorami, przypominały skrzyżowanie centrali telefonicznej z fortepianem. Malowane zazwyczaj na czarno, kształtem i wielkością podobne do kompletu szaf ubraniowych, z przednią ścianą przeznaczoną na pulpit sterowniczy, z dziesiątkami pokrętel, lampek, przełączników i gniazdek wtykowych. Syntezatory miały też niewielką tradycyjną klawiaturę, na której grało się, jak na zwykłym instrumencie. Z tym tylko, że uderzenie w klawisz nie powodowało wibracji powietrza w rurach czy drgań pobudzonej struny, lecz przepływ prądu w obwodach elektrycznych wybranych przez odpowiednie nastawy na pulpicie sterującym. Pobudzenie tych obwodów wytwarzało strumień dźwięków zgodnych z intencją operatora, a nie pochodzących z instrumentów, przetworzenia taśmy lub dźwięków fizycznych.

**Syntezator może więc naśladować brzmienia tradycyjnych instrumentów muzycznych, powtarzać głosy ludzkie, wszystkie dźwięki dobiegające z otoczenia. Potrafi także tworzyć efekty akustyczne, które nigdy przedtem na świecie nie mogły istnieć.**

Idea jest niby prosta, ale jej techniczna realizacja dużo trudniejsza. Pierwszy syntezator, wykonany za ćwierć miliona dolarów na zlecenie RADIO CORPORATION of AMERICA, rozpoczął pracę na Uniwersytecie Columbia w 1958 roku. Było to jednak urządzenie dość pokaźnych rozmiarów, jakby całe studio wciśnięte w obręb jednej maszyny. Konstruktorem, któremu udało się zrobić model syntezatora w miarę wszechstronnego, a zarazem niezbyt trudnego do wykonania i użytkowania, jest czterdziestoletni obecnie dr Robert A. Moog. Odwrotnie niż Varese z wykształcenia inżynier, pasjonował się od dziecka muzyką. W latach studenckich próbował połączyć oba kierunki swoich zainteresowań, budując układy elektryczne wydające dźwięki — ilustrację muzyczną do filmów. Nieco później zetknął się z kompozytorami, którzy podejmowali identyczne prace, ale nie mając technologicznego przygotowania budowali urządzenia złożone i mało efektywne. Moog służył im swoją wiedzą techniczną, a w zmian, dowiadywał się czego muzyka najbardziej potrzebuje. Minisyntezatory o wymiarach mniej więcej metr na metr

powstające pod fachowym okiem Mooga w kształcie gabinetowych sekretarzyków, odpowiadały w pełni wymaganiom twórców i natychmiast znajdowały nabywców. Nazwisko konstruktora tak nierozerwalnie wiąże się z syntezatorami, że nawet te produkowane gdzie indziej określane są jako „moogi" (przy tym zapomina się często o flamandzkim rodowodzie dra Mooga i na ogół wymawia się tę nazwę z amerykańska: „mugi").

Syntezator Mooga, podobnie jak studio muzyki elektronicznej, składa się z urządzeń do wytwarzania sygnału elektrycznego i takich, które ten sygnał modyfikują. Zmodyfikowane sygnały elektryczne sterują instrumentami elektronicznymi powodując wytwarzanie się dźwięków. Moog zawiera pewną liczbę odrębnych bloków, są to: różne generatory przebiegów piłokształtnych, prostokątnych i szumów, zasilacze, oscylatory, filtry, modulatory kołowe i przesuwniki fazy. Sterowanie blokami instrumentów odbywa się tu na skutek wahań napięcia, które uzyskujemy z układów elektronicznych lub programujemy z możliwością dokonywania zmiany tegoż napięcia ręcznie za pośrednictwem pulpitu operatorskiego. Wszystkie bloki syntezatora są projektowane w ten sposób, by nadawały się do jak największej liczby zastosowań. Dają się one w dowolny sposób łączyć z innymi urządzeniami studia muzycznego, ale mogą się bez nich obejść. Żeby uzyskać pełny zestaw do eksperymentów, wystarczy dodać do mooga wzmacniacz, mikser, głośnik i dwa lub trzy magnetofony. Zakłady w Trumansburgu wraz z wytwórnią organów w Chicago produkują syntezatory o różnych wymiarach i zdolnościach. Najmniejsze, ważące niewiele ponad 10 kilogramów, kosztują 1200 dolarów. Najbardziej złożone osiągają ceny rzędu 20 tysięcy dolarów. Przy fabryce w Trumansburgu znajduje się znakomicie wyposażone studio wzorcowe, otwarte dla klientów, którzy chcą zapoznać się z walorami nowego sprzętu i mogą na miejscu przeprowadzać własne próby syntezy.

**Nagrania dokonane z udziałem moogów udały się niespodziewanie. Pierwszy longplay z utworami Jana Sebastiana Bacha (m.in. II Koncert Brandenburski) nagrano w wytwórni COLUMBIA w 1968 roku. Autor, aranżer (choć może poprawniej byłoby go nazwać operatorem mooga czy syntezystą), Walter Carlos, ochrzcił swoje dzieło „Switched-On Bach" Tytuł trafia w sedno — „Włączenie" albo raczej „Przełączenie Bacha" dobrze określa czynności programowania syntezatora na grę Bacha. Płyta, zaskakująca świeżością brzmienia i perfekcją wykonania, pobiła rekord sprzedaży — natychmiast pojawiły się naśladownictwa. Można by pomyśleć, że to triumf Bacha, że to jego geniusz przezwycięża najwymyślniejszą adaptację. Ale nie — po baroku przyszła kolej na muzykę klasyczną i romantyczną.** Rynek wchłaniał równie szybko następne „syntetyczne" płyty. Jedna z nich — nazwana z lekką megalomanią „The Moog strikes Bach... to say nothing of Chopin, Mozart, Rachmaniniff, Paganini and Prokofieff" („Moog odkrywa Bacha... nie mówiąc już o Chopinie, Mozarcie, Rachmaninowe, Paganinim i Prokofiewie") — pozwala na stwierdzenie, że moogi „nie kłócą się" z duchem polskiej muzyki. Oto wyjątki z recenzji zamieszczonej w czasopiśmie „Ruch Muzyczny": „Na pierwszy ogień idzie Chopin Etiuda op. 10 nr 5 (na czarnych klawiszach) i ... robi dobre wrażenie. Czyste, bogate brzmienie, niewiarygodnie precyzyjne, miękkie staccato, — to zasługa instrumentu nie syntezysty! — doskonałe tempo". Łatwość sukcesu na nowym polu spowodowała wezbranie fali nagrań, Austriacki inżynier elektroakustyk Hans Wurman, realizator „The Moog Strikes Bach...", wzbudził zachwyt melomanów dalszymi opracowaniami, za rewelacyjne zwłaszcza uznano „Eine Kleine Nacht Musik" i „Marsz Turecki" Mozarta. Syntezowano już nie tylko muzykę poważną, ale i lekką, czego dowodem płyty „Moog Espana" i „Moog Power" (RCA), „Swithed-On Moog" (ATHENA), „Age of Electronicus" i „Moog-Electis" (COMMAND).

Nie pominięto również muzyki młodzieżowej —np. „Swithed-On Rock" (COLUMBIA), przy czy znów wyróżnił się tu Wurman syntezą znanej melodii „Jumpin' Jack Flesh" napinanej przez Micka Jaggera z „Rolling Stonesów".

**Moogi przyjęły się na stałe i mają swój udział w sukcesach najsłynniejszych zespołów młodzieżowych. Często, słuchając skrzypiec, gitar, organów elektrycznych nie podejrzewamy nawet obecności w utworze imitującego je mooga.** Ale chociaż nagrania moogów pojawiają się w Polskim Radio, wiedza o syntezatorach nie wykracza na ogół poza kawiarniane ogólniki. Krytyk muzyczny Mirosław Kondracki, dowiedział się — jak sam pisze — podczas jednej z audycji, że „Muk (sic!) to myśląca maszyna, która gra, kiedy włoży się w nią program", a z drugiej audycji, że „że muk to mózg zaprogramowany" i zagroził, że po trzeciej audycji na podobnym poziomie „pojawi się w redakcji z wiatrówką".

Chociaż syntetyzatory można uznać za dość złożone maszyny analogowe, to trudno jeszcze uważać je za komputery. O muzyce tworzonej przez maszyny cyfrowe pisać będziemy za miesiąc.

**Marek Hołyński**

---

## Eurocomp 75

Tegoroczna konferencja EUROCOMP 75 odbywająca się tradycyjnie w Londynie planowana była na kwiecień. Pierwotny termin uległ jednak zmianie i EUROCOMP 75 odbędzie się ostatecznie w dniach 23—25 września. Planowane są posiedzenia w dwu sekcjach:

- Komunikacja i sieci komputerowe
- Systemy współdziałające i automatyzacja projektowania.

Posiedzenia odbywać się mają w nowozbudowanym hotelu Heathrow specjalnie przeznaczonym dla celów konfrencji, położonym w pobliżu głównego lotniska międzynarodowego. Organitaorzy starają się, aby była to największa europejska konferencja komputerowa. Warto przy okazji przypomnieć, że na ubiegłorocznej majowej EUROCOMP 74 odbyło się 31 sesji, na których wygłoszono ponad 90 referatów.

Brało w niej udział 800 uczestników z 20 krajów, a w wystawach i prezentacjach sprzętu uczestniczyło ponad 100 firm komputerowych. (mh).

## Komputer w roli astronoma

W wielu obserwatoriach astronomicznych na świecie używa się już komputerów m.in. do naprowadzania teleskopów na wybrane do obserwacji ciała niebieskie. Ostatnio w amerykańskiej uczelni Masschausetts Institute of Technology oddano do eksploatacji pierwszy na świecie całkowicie skomputeryzowany teleskop. Maszyna w mariażu z teleskopem zastępuje astronoma przyjmując „zamówienia" na żądane obserwacje, listę gwiazd do przejrzenia, zlecenia na wykonanie obliczeń, zdjęć analiz itp. Jak twierdzą amerykańscy specjaliści, skomputeryzowany teleskop pracuje dwa razy efektywniej, czyli wykonuje pracę, która wymagałaby zaangażowania dwóch teleskopów — urządzeń — jak wiadomo — bardzo drogich. (kr)

## Rozstrzeliwanie chmur

Chmury gęstnieją. Jeszcze parę minut i na zasiewy runie ulewa z gradem. Zaczynają działać rakietowe stanowiska. Salwa jedna i jeszcze jedna, Chmury bledną, rozpraszają się. Niebezpieczeństwo minęło.

\ wszystko odbywa się bez udziału ludzi. Komputer skonstruowany przez uczonych laboratorium teoretycznej cybernetyki Uniwersytetu Leningradzkiego, sam rozróżnia grożące gradobiciem chmury i wydaje rozkaz rozstrzelania ich.

Maszynę można „uczyć". „Pokazuje się" jej zdjęcia chmur, które należy odróżniać od siebie. Istotne charakterystyczne oznaki chmur sortuje analizator, po czym przechodzą one do aparatu klasyfikacyjnego. Kiedy aparat „zauważy" na niebie chmurę, porównuje jej charakterystykę z tym, co przechowane jest w elektronicznej pamięci i podejmuje decyzję. (kr)

# Propozycje Komitetu CODASYL w dziedzinie systemów zarządzania bazą danych

Komitet CODASYL (Conference on Data Systems Languages) zrzesza przedstawicieli około 60 organizacji w tym wszystkich poważnych producentów komputerów, szeregu agencji rządowych oraz przedsiębiorstw będących dużymi użytkownikami sprzętu informatyki. Prace prowadzone w ramach CODASYLU są realizowane przez następujące zespoły specjalistów:

- Systems Committe prowadzący prace w zakresie automatyzacji projektowania i wdrażania systemów informatycznych
- Planning Committe spełniający rolę organizacji pośredniczącej pomiędzy producentami oprogramowania a grupami użytkowników
- Programming Lnaguage Committee pracujący nad rozwojem języków programowania. Podstawowym kierunkiem prac jest rozwój języka COBOL
- Data Description Language Comittee został zorganizowany we wrześniu 1971 z zadaniem opracowania specjalizacji uniwersalnego Języka Opisu Danych *(Data Desciption Language)*, niezależnego od istniejących obecnie języków programowania.

Data Base Task Group stanowiła jedną z sześciu grup roboczych PLC koncentrującą się nad problematyką systemów zarządzania bazą danych.
Efektem prac DBTG była publikacja dwóch raportów [1] i [2] stanowiących propozycję w zakresie systemów zarządzenia bazą danych. DBTG zakończyła swoją działalność po ogłoszeniu drugiego raportu stanowiącego ostateczną wersję propozycji uniwersalnego systemu zarządzania bazą danych. Prace nad Językiem Opisu Danych są kontynuowane w ramach DDLC, który ogłosił w czerwcu 1973 następną wersję tego języka [3]. Wersja DDLC wprowadza szereg drobnych zmian (składnia, dodatkowe opcje) do propozycji DBTG opierając się całkowicie o zaproponowaną w raporcie z kwietnia 1971 architekturę uniwersalnego systemu zarządzania bazą danych.

Język Manipulacji Danymi dla bazowego języka programowania COBOL jest przedmiotem prac PLC ogłoszonych w marcu 1973 [4].

Podobnie, jak w przypadku JOD ogłoszona wersja języka nie stanowi widocznego postępu w zakresie prac nad językami manipulacji danymi wprowadzając jedynie zmiany składni poszczególnych komend JMD wynikających z rozwoju języka COBOL. Poważnym mankamentem prac prowadzonych w ramach BLC jest koncentracja nad zagadnieniami wynikającymi z rozszerzenia możliwości języka COBOL, ponieważ zarówno Język Opisu Danych dla Schematu bazy danych w wersjach DBTH i DDLC, jak i Język Manipulacji Danymi w wersji DBTG są wystarczająco niezależne dla zastosowania ich w połączeniu z innymi bazowymi językami programowania takimi, jak na przykład PL/1 i FORTRAN.

Propozycja Data Base Task Gropu doczekała się już szeregu implamentacji wykonanych dla komputerów amerykańskich oraz kilku zachodnioeuropejskich.

Do najbardziej znanych implementacji raportu DBTG należą:

- Data Management System 1100 (UNIVAC 1100)
- Integrated Data Management System (IBM 360/370)
- Data Base Management System-10 (PDP-10)
- Data Management System (DATA SAAB)
- PHOLAS (PHILIPS)
- Data Base Management System (ICL New Range)

Zaawansowane prace nad implementacją raportu DBTG są prowadzone przez firmy NCR i CONTROL DATA. Firma HONEYWELL dysponuje systemem IDS (Integrated Data Store) przejętym z GENERAL ELECTRIC i stanowiącym prototyp dla prac DBTG. Wszystkie dotychczasowe implementacje raportu DBTG stanowią podzbiory Języków Opisu Danych (schemat, subschemat-COBOL) i stosunkowo pełne wersje Języka Manipulacji Danymi. Językiem bazowym jest przede wszystkim COBOL, ze względu na jego największą popularność w dziedzinie systemów przetwarzania danych do zarządzania. Obszar zastosowań tych systemów obejmuje szeroki zakres dziedzin, między innymi systemy przetwarzania danych w przedsiębiorstwach przemysłowych, organizacjach handlowych i usługowych oraz w wyższych uczelniach.

Prace DBTG opierały się o następujący zestaw kryteriów uznanych przez zespół za podstawowe wymagania użytkowe w stosunku do proponowanego rozwiązania:

— możliwość elastycznego konstruowania logicznej struktury danych optymalnej z punktu widzenia wielu różnych zastosowań
— równoległa obsługa dostępu do bazy danych dla wielu zadań
— możliwość stosowania różnych strategii wyszukiwania w bazie danych lub częściach bazy danych
— zabezpieczenie przed nielegalnym dostępem do bazy danych oraz wzajemnym zakłócaniem pracy przez programy użytkowe
— centralne sterowanie fizycznym rozmieszczeniem danych w bazie danych
— możliwość opisu różnorodnych klas struktur logicznych — od niepowiązanych grup danych do struktur sieciowych
— manipulacja elementów struktury logicznej bez potrzeby utrzymywania zadeklarowanych powiązań
— uzyskanie możliwie maksymalnej niezależności programów od danych
— podział pomiędzy opisem struktury danych w bazie danych a strukturą danych w programie
— mechanizm opisu bazy danych, niezależny od istniejących języków programowania
— architektura systemu umożliwiająca wykorzystanie bazy danych przez różne języki programowania.

Spełnienie powyższych założeń prowadzi do uzyskania wystarczającego stopnia uniwersalności i elastyczności rozwiązania poprzez stworzenie możliwości opisywania i manipulacji dowolnie złożonych struktur danych.

Przyjęte rozwiązanie nie wyklucza możliwości zastosowania języka bezpośredniego dostępu, tym niemniej taki język nie stanowił przedmiotu prac DBTG. Raport wskazuje jedynie na konieczność rozwinięcia prezentowanej koncepcji o mechanizm pośredniczący dla użytkownika bezpośredniego (rys. 1).

## Języki opisu danych *(Data Description Languages)*

Języki służące do opisu bazy danych lub części bazy danych znajdującej się w polu widzenia programu użytkowego. Języki opisu danych stanowią mechanizm pośredniczący pomiędzy Administratorem Bazy Danych a procedurami zarządzania bazy danych, służący do zakładania i utrzymania bazy danych.

**Opis struktury bazy danych obejmuje nazwy i atrybuty:** danych, danych zbiorowych, zapisów (rekordów), obszarów, zbiorów zawartych w bazie danych.

**Języki opisu danych operują następującymi podstawowymi pojęciami:**

— **dana:** najmniejszy, nazwany element struktury logicznej bazy danych

— **dana zbiorowa:** nazwana grupa danych występująca w ramach zapisu (rekordu)

— **wektor jednowymiarowy:** uporządkowany ciąg danych o tej samej charakterystyce

— **powtarzalna grupa:** ciąg danych występujący dowolną liczbę razy w ramach danego wystąpienia zapisu (rekordu). Grupa może składać się z danych, wektorów i powtarzalnych grup

— **zapis** (rekord): nazwany ciąg zera, jednej lub więcej danych, lub danych agregowanych

— **zbiór:** nazwana grupa typów zapisów (rekordów). Każdy zbiór zdefiniowany w opisie danych musi mieć jeden typ zapisu (rekordu) określony jako zapis (rekord) — właściciel i jeden lub więcej typów zapisów (rekordów) określonych jako zapis (rekord) — członek. Każde wystąpienie zbioru musi zawierać jedno wystąpienie zapisu (rekordu) właściciela, oraz dowolną liczbę wystąpień zapisów (rekordów) — członków

— **obszar:** nazwana część wynikająca z podziału adresowalnego przedziału bazy danych, zawierająca wystąpienia zapisów (rekordów) i zbiorów lub części zbiorów różnych typów

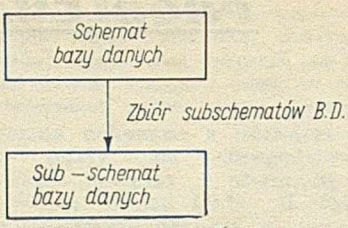
— **baza danych:** wszystkie wystąpienia zapisów (rekordów), zbiorów i obszarów zdefiniowanych w opisie bazy danych. (Schemat bazy danych)

— **schemat bazy danych:** sekwencja zdań języka opisu danych stanowiąca kompletny opis struktury bazy danych

— **subschemat bazy danych:** sekwencja zdań języka opisu danych stanowiąca opis tych obszarów, zbiorów, zapisów (rekordów) danych i danych agregowanych, które znajdują się w polu widzenia wybranego programu użytkowego lub grupy wybranych programów użytkowych.

**Języki opisu danych zawierają:** język opisu schematu bazy danych i języki opisu subschematu dla języka COBOL (rys. 2).

Wprowadzenie odrębnych Języków Opisu Danych dla schematu i subschematów bazy danych pozwoliło na




Rys. 2. Struktura opisu bazy danych

uzyskanie — ze względu na dopuszczalne różnice pomiędzy schematem a subschematem — pewnej niezależności programów użytkowych od struktury danych bazy danych.

Zmiana logiki programu użytkowego jest konieczna jedynie w przypadku wystąpienia zmian struktury danych opisującej fragment bazy danych objęty subschematem wykorzystywanym przez dany program, jeżeli zmiany wykraczają poza dopuszczalne różnice pomiędzy schematem a subschematem. Dopuszczalne są następujące różnice pomiędzy schematem a subschematem na poziomie:

- **danej elementarnej:**
— nazwa danej elementarnej
— charakterystyka danej elementarnej
— wskaźnik blokady dostępu
— kolejność danych elementarnych
— pominięcie opisu danej elementarnej
- **danej zbiorowej:**
— nazwa danej zbiorowej
— pominięcie opisu danej zbiorowej
— wskaźnik blokady dostępu
— kolejność danych zbiorowych
— redefinicja wektora lub tablicy
— nowe struktury wynikające z charakterystyki danego Języka Opisu Danych dla subschematu
- **zapisu (rekordu):**
— nazwa zapisu (rekordu)
— pominięcie opisu zapisu (rekordu)
— wskaźnik blokady dostępu
— struktura danych zapisu (rekordu)
- **zbioru:**
— nazwa zbioru
— pominięcie opisu zbioru
— wskaźnik blokady dostępu
— zasady selekcji wystąpienia zbioru.

Możliwość uzyskania niezależności Języka Opisu Danych dla schematu od przyjętych w systemie bazowych języków programowania wynika z założenia, że każdy z tych języków programowania wymaga indywidualnej wersji Języka Opisu Danych dla subschematu.

Schemat bazy danych napisany w JOD zawiera cztery podstawowe typy zdań spełniające funkcje:

- Identyfikacji schematu (zdanie schemat)
- Definicja obszarów (zdanie obszar)
- Definicja zapisów (rekordów) (zdanie zapis (rekord))
- Definicja zbiorów (zdanie zbiór)

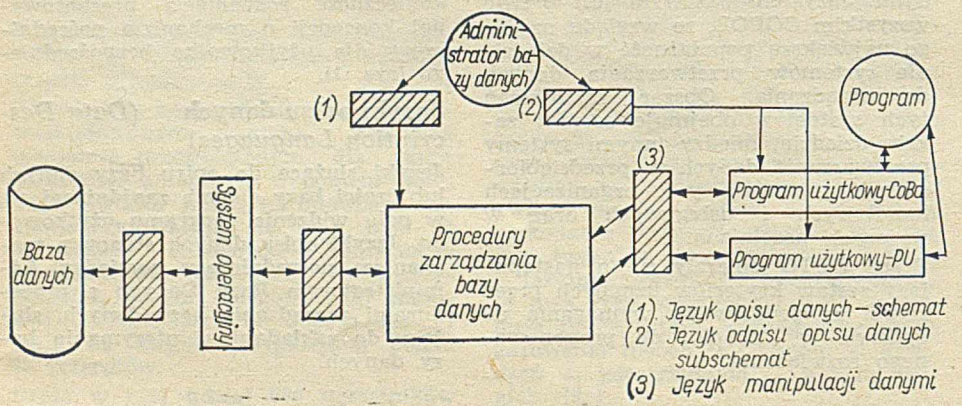
Osobne zdanie JOD jest wymagane dla każdego opisywanego w schemacie typu obszaru, zapisu (rekordu) i zbioru.

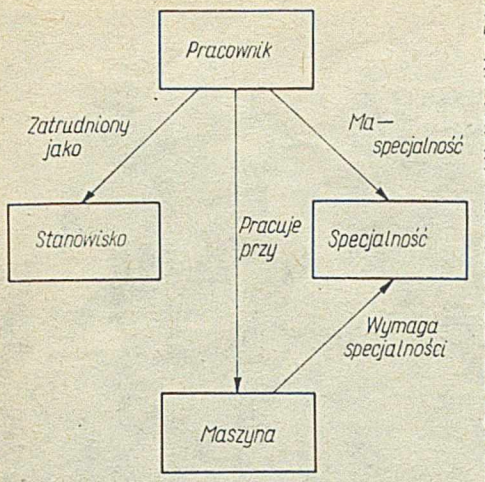
Zdanie języka składa się z jednej lub więcej klauzul, które definiują jego atrybuty. Zasady budowania schematu bazy danych w oparciu o Język Opisu Danych ilustruje przykład.

**Definicja fragmentu schematu bazy danych w Języku Opisu Danych.**

```
SCHEMA NAME IS BAZA — PRACOWNIKÓW;
PRIVACY LOCK IS HASŁO:
AREA NAME IS ZATRUDNIENIE; COMMET „DEFINICJA OBSZARÓW"
    .
    .
    .
    .
    .
    .
    .
    .
RECORD NAME IS PRACOWNIK COMMENT „DEFINICJA REKORDÓW"
LOCATION MODE IS CLAC PRAC—HASH
USING NUMER—PRACOWNIKA;
WITHIN ZATRUDNIENIE;
ON DELETE CALL ZAPIS—MIKROFILMOWY;
1 NAZWISKO PICTURE IS A(30)
    .
    .
    .
    .
    .
    .
RECORD NAME IS STANOWISKO
    .
    .
    .
    .
    .
RECORD NAME IS SPECJALNOŚĆ
    .
    .
    .
    .
    .
RECORD NAME IS MASZYNA
    .
    .
    .
    .
    .
SET NAME IS ZATRUDNIONY—JAKO
COMMENT „DEFINICJA ZBIORÓW"
ORDER IS NEXT
OWNER IS PRACOWNIK
NUMBER IS STANOWISKO
SET OCCURENCE SELECTION IS THRU
CURRENT OF SET
    .
    .
    .
    .
    .
SET NAME IS PRACUJE — PRZY
ORDER IS SORTED
OWNER IS PRACOWNIK
NUMBER IS MASZYNA
LINKED TO OWNER
ASCENDING KEY IS NUMER—MASZYNY
```

Rys. 1. Schemat architektury według propozycji DBTG






Rys. 3. Graficzna reprezentacja schematu bazy danych

## Język manipulacji danymi *(Data Manipulation Language*

Język obsługujący funkcje przesyłania danych pomiędzy programem użytkowym a bazą danych.
JMD posiada następujące cechy charakterystyczne:

— nie jest pełnym językiem programowania

— jest zawarty w bazowym języku programowania obsługującym funkcje proceduralne

— zdania JMD są umieszczone dowolnie w sekwencji zdań języka bazowego

— z punktu widzenia programowania JMD i język bazowy stanowią jeden język programowania — relacja między JMD a językami opisu danych jest relacją typu procedura—deklaracja

— relacja między JMD a językiem bazowym jest oparta o założenie, że wszystkie odwołania do bazy danych typu: wybierz dane, dodaj nowe dane, modyfikuj istniejące dane lub powiązania danych są zapisywane w języku manipulacji danymi.

Przyjęty w raporcie DBTG format komend JMD nie jest uzależniony od przyjętego w konkretnej implementacji systemu bazowego języka programowania.

Mimo niezależnego charakteru Języka Manipulacji Danymi raport DBTG jest zorientowany przede wszystkim na język programowania COBOL.
Podobnie, jak COBOL, JMD składa się z trzech części:

- *Identification Division*
- *Data Division*
- *Procedure Division.*

*Identification Division* programu COBOLowego jest wykorzystywany dla przekazywania kluczy dostępu (PRIVACY KEY) dla zdefiniowanych w schemacie (subschemacie) blokad wykonania operacji JMD.

*Data Division* stanowi obszar programu użytkowego, w którym są generowane opisy danych i zapisów (rekordów), zawartych w subschemacie oraz opis obszaru komunikacji systemu.

Numerowanie strzałki na schemacie ilustrują realizację współpracy poszczególnych elementów systemu:
1 — odwołanie programu użytkowego do USZBD za pośrednictwem komendy JMD
2 — analiza odwołania i zastąpienia jego parametrów wartościami zawartymi w wersji wykonawczej schematu bazy danych i w wersji wykonawczej subschematu wykorzystywanego przez dany program użytkowy
3 — procedury zarządzania bazy danych inicjują operacje wejścia/wyjścia na podstawie odwołania programu użytkowego i informacji zawartych w wersajch wykonawczych schematu i subschematu
4 — system operacyjny obsługuje fizyczny transfer danych pomiędzy urządzeniem pamięci zewnętrznej, a pamięcią operacyjną
5 — fizyczny transfer pomiędzy urządzeniem pamięci zewnętrznej a buforami systemu
6 — procedury zarządzania bazą danych wykonują przesunięcie danych zgodnie z odwołaniem programu użytkowego pomiędzy buforem systemu a obszarem roboczym użytkownika zawartym w tym programie użytkowym
7 — procedury zarządzania bazą danych przechowują informacje o wykonaniu odwołania. Informacje te zawierają: wskaźnik bieżącej selekcji, kody błędów, nazwę zapisu (rekordu), nazwę — obszaru
8 — wartości zawarte w obszarze roboczym użytkownika są przetwarzane w oparciu o funkcje bazowego języka programowania
9 — procedury zarządzania bazą danych sterują wykorzystaniem buforów systemu

Wywołanie odpowiedniego subschematu następuje za pośrednictwem komendy INVOKE.

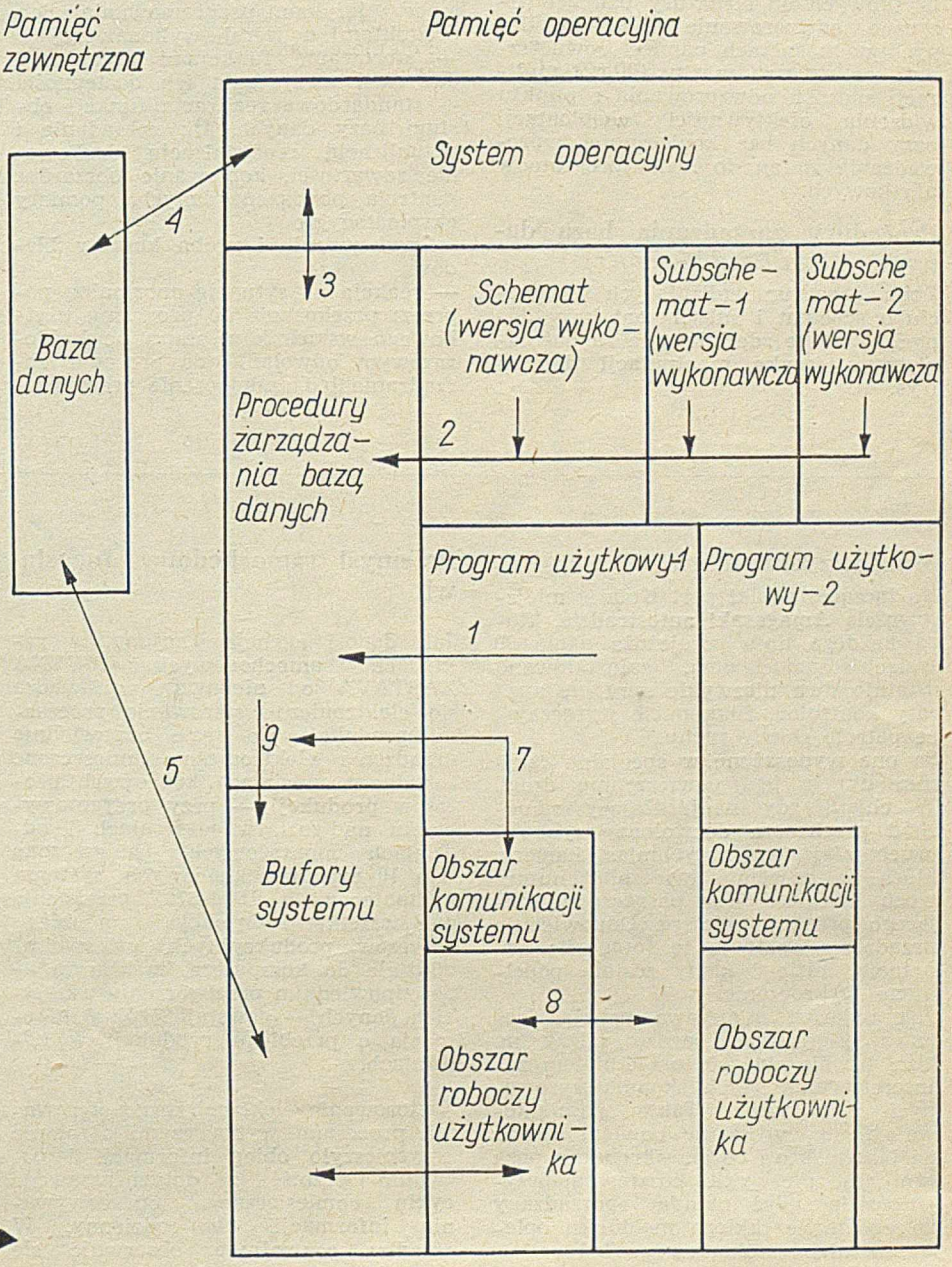
Wszystkie pozostałe komendy Języka Manipulacji Danych mogą być umieszczane w dowolnych miejscach *Procedure Division* programu użytkowego.
Działanie na strukturach danych deklarowanych w schemacie/subschemacie bazy danych jest realizowane za pośrednictwem następujących imperatywnych komend JMD:

- **komendy sterujące**

OPEN — otwarcie obszaru(ów)
CLOSE — zamknięcie obszaru(ów)
IF — zdanie warunkowe
KEEP — blokada zapisu (rekordu)
FREE — zwolnienie zapisu (rekordu)
USE — sterowanie obsługą błędów

- **Komendy działania na rekordach**

FIND — wyszukiwanie zapisu (rekordu)
GET — przesunięcie zapisu (rekordu) lub danych
STORE — zapisanie zapisu (rekordu)
DELETE — skreślenie zapisu (rekordu)




Rys. 4. Schemat funkcjonalny oprogramowania według propozycji DBTG

● **Komendy działania na zbiorach**
INSERT — włączenie zapisu (rekordu) do zapisu (ów)
REMOVE — usunięcie zapisu (rekordu) ze zbioru (ów)
ORDER — uporządkowanie zbioru
● **Komenda komunikacji z systemem**
MOVE — udostępnienie pól komunikacji z systemem

Repertuar komend JMD umożliwia pełne wykorzystanie złożonych struktur danych możliwych do zadeklarowania za pośrednictwem Języka Opisu Danych.

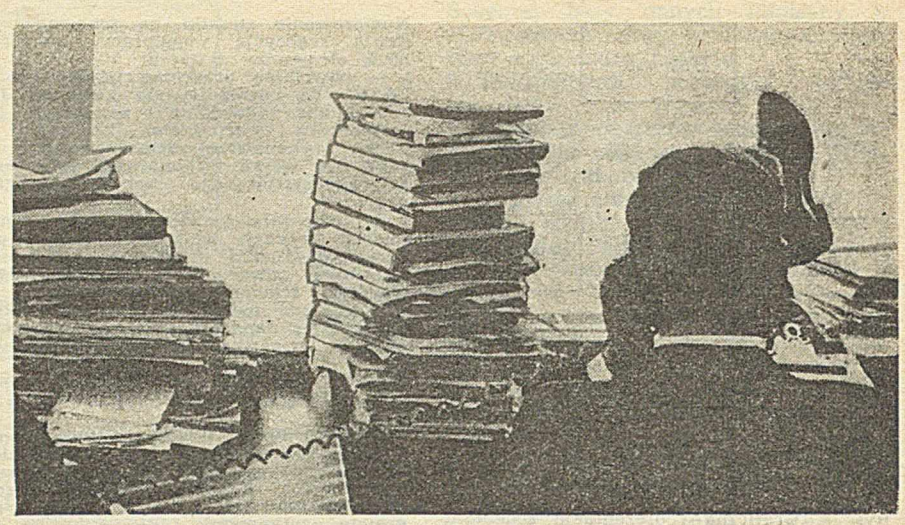
Wykorzystanie tych komend wymaga jednak od programisty dobrej znajomości struktury danych objętych w subschemacie wykorzystywanym przez jego program.

Mocne związanie Języka Manipulacji Danymi z opisem struktury danych stanowi z pewnością pewien mankament przyjętego rozwiązania, umożliwiając równocześnie znacznie pełniejsze wykorzystanie potencjału tych struktur niż w przypadku innych współcześnie proponowanych rozwiązań w dziedzinie systemów zarządzania bazą danych.

Ważną cechą rozwiązania DBTG jest stosunkowo dobra izolacja programów użytkowych od struktury pamięci (fizyczne odwzorowanie struktury danych na nośnikach pamięci zewnętrznej), co stwarza możliwość optymalizacji sposobu odwzorowania z punktu widzenia efektywności eksploatacji bazy danych bez konieczności wprowadzania zmian do logiki programów użytkowych.

### Procedury zarządzania bazą danych *(Data Base Manager)*

Zbiór procedur realizujących w oparciu o schemat i subschemat bazy danych, funkcje zdefiniowane za pośrednictwem języka manipulacji danymi (rys. 4).



Oprogramowanie procedur zarządzania bazą danych realizuje następujące funkcje:
— manipulacja danymi na poziomie struktury danych i struktury pamięci zgodnie z zasadami zdefiniowanymi w jej opisie
— utrzymywanie komunikacji poprzez standardowe mechanizmy pośredniczące z językami programowania i ich produktami w wersji wykonawczej
— sterowanie funkcjami organizacji danych i utrzymania ich dostępności
— standardowa realizacja funkcji obsługi bazy danych, tj.: zakładanie i aktualizacją, reorganizacja, drukowanie zawartości, kopiowanie obszarów, kontrola powiązań danych, pomiary eksploatacyjne
— wykrywanie i próba korekty błędów
— reakcja na sytuację anormalną poprzez: przekazanie do programu użytkowego wskaźników stanu po każdorazowym odwołaniu do procedur zarządzania b.d., zakończenie pracy programu użytkowego, odcięcie dostępu do części bazy danych, przywrócenie bazy danych do poprzedniego stanu
— obsługa sytuacji konfliktowych
— ochrona dostępu do bazy danych.

**Władysław Bogucki**
**Witold Staniszkis**

Literatura


[1] Report of the Data Base Task Group, October 1969, ACM 1969
[2] Data Base Task Group, April 71 Report, ACM 1971
[3] Codasyl Data Description Language — JOURNAL OF DEVELOPMENT, June 1973, National Bureau of Standards
[4] CODASYL COBOL Data Base Facility Proposal, March 1973, Specifications Board Canadian Government
[5] Data Base Task Group, — CODASYL, BCS October 71, Conference on April 71, REPORT BRITISH COMPUTER SOCIETY 1971


---

## Mandat z komputera

Na łączącej wielki port Kobe z miejscowością Amagasaki autostradzie, którą każdego dnia przejeżdża około 90 tysięcy samochodów, zainstalowano ostatnio dwa niezwykle sprawne aparaty kontrolne, znakomicie wyręczające patrole służby ruchu.

Są one wyposażone w specjalne czujniki ukryte pod powierzchnią drogi. W chwili gdy nadjeżdżający samochód przekracza dozwoloną szybkość, uruchamiają one natychmiast kamerę, która na zdjęciu „uwiecznia" numer wozu kierowcę oraz pasażera zajmującego przednie miejsce. Odpowiednie urządzenia opatrują tę fotografię adnotacją, gdzie i kiedy zostało popełnione wykroczenie.

Jak twierdzi miejscowa policja drogowa, kierowcom bardzo zależy na tym, by nie stać się przedmiotem zainteresowania tych skomputeryzowanych inspektorów. Takie „pamiątkowe zdjęcie" wykazuje bowiem czarno na białym kto i co przeskrobał, a poza tym jest niezwykle kosztowną przyjemnością, gdyż mandat sporządzany na podstawie takiego meldunku opiewa na niebagatelną sumę. (kr)

## Przemysł samochodowy Jugosławii

Jak donosi agencja Tanjung, w zakładach samochodowych CRVENA ZASTAVA od niedawna wprowadza się elektroniczne sterowanie procesami produkcji. Na razie odpowiednie urządzenia elektroniczne umieszczone zostały w oddziałach końcowych procesów produkcji — przy przygotowywaniu nadwozi, w lakierniach i oddziałach montażowych. Umieszczono tam 10 urządzeń końcowych, systemu komputerowego IBM-370, przekazują one wszelkie informacje o przebiegu procesów produkcyjnych w „swoim" oddziale do komputera, w którym — po odpowiednim przetworzeniu wszystkich danych — powstaje ogólna informacja o przebiegu produkcji w oddziałach.

Zastosowanie elektronicznego sterowania procesami wytwórczymi ogromnie przyśpieszyło obieg informacji i pozwoliło przejść z dotychczasowego cyklu comiesięcznego opracowywania informacji, na codzienny. W ośrodku centralnym obserwuje się obecnie przebieg końcowych stadiów produkcji samochodów, aż do chwili zejścia z taśmy gotowych pojazdów. W dowolnym momencie można sprawdzić, gdzie znajduje się obserwowany samochód, jaki posiada numer nadwozia i silnika, typ każdej części wmontowanej w ten egzemplarz wozu, kolor karoserii i nawet kolor tapicerki.

Komputer oprócz liczby wyprodukowanych samochodów — rejestruje także inne dane. Na przykład — na podstawie danych o częściach wmontowanych w samochody — informuje on o stanie zapasów magazynowych poszczególnych zespołów i części oraz sygnalizuje wyczerpywanie się zapasów, co przeciwdziała powstawaniu przestojów produkcyjnych.

Kierownictwo zakładów CRVENA ZASTAVA twierdzi, że jest to dopiero początek zastosowania elektronicznego systemu przetwarzania danych w fabryce. W najbliższej przyszłości instalowane będą sukcesywnie w następnych oddziałach produkcyjnych i w krótce system IBM-370 obejmie większość oddziałów i procesów wytwórczych (kr)

# Recenzja książki Z. Gackowskiego: Projektowanie systemów informacyjnych zarządzania*)

Na półkach księgarskich ukazała się książka Zbigniewa Gackowskiego pt. „Projektowanie systemów informacyjnych zarządzania". Książka wywołała duże zainteresowanie wśród fniormatyków ze względu na brak tego rodzaju publikacji książkowych polskich autorów, jak również rzadkie pozycje książkowe autorów zagranicznych.

Szybko rozwijające się Ośrodki Elektronicznego Przetwarzania Danych (OEPD) w zakładach przemysłowych i instytucjach nadrzędnych dotychczas pracowały w zasadzie na własnych, niezależnych opracowaniach projektowych, korzystając czasem z usług instytucji specjalistycznych.

Po książce tej spodziewano się stworzenia jednolitych zasad projektowania systemów informatycznych, jak również metod prowadzenia prac projektowych. Oczekiwania te nie zostały w pełni zaspokojone, ponieważ Autor podszedł do zagadnienia zbyt ogólnie i teoretycznie.

Treść książki można umownie podzielić na dwie logicznie powiązane części. Pierwsze trzy rozdziały książki obejmujące podstawowe problemy, pojęcia, oraz podstawy systemów informatycznych, miały na celu zaznajomienie czytelnika z podstawami informatyki, natomiast pozostałe rozdziały dotyczą problematyki projektowania systemów informatycznych.

Projektant systemów informatycznych, do którego książka ta jest kierowana, jeżeli nie zna dobrze podstaw teorii informacji, teorii wielkich systemów, teorii gier czy teorii grafów, nie jest w stanie po przeczytaniu prezentowanych przez Autora teoretycznych wywodów, zrozumieć tych zagadnień, nie mówiąc już o jakiejkolwiek możliwości skorzystania z nich w praktyce zawodowej.

Prezentowanie pojęć bez przykładów ich zastosowania, staje się dla projektanta SI, działającego w rzeczywistych warunkach funkcjonującego przedsiębiorstwa, zupełnie nieprzydatne. Jaskrawym przykładem niezmiernie ważnego pojęcia, którego wyjaśnienie jest niedostateczne i niejasne, to pojęcie „cenności" informacji (R2 s. 41). W rozdziale pierwszym „Podstawowe problemy" Autor krótko charakteryzuje niezbędne elementy prac przygotowawczych do kompleksowej automatyzacji przetwarzania danych, przedstawiając całe przedsięwzięcie w postaci harmonogramu. Szczegółowe omówienie elementów prac jest treścią dalszych rozdziałów książki.

Z harmonogramu wynika, że wdrożenie kompleksowego systemu APD dokonane zostanie w ciągu 6 lat.

Autor natomiast nie wspomina o wpływie czasu opracowywania na całokształt systemu. Projektanci-analitycy muszą wziąć pod uwagę wpływ czasu i warunki, w jakich system będzie działał w przyszłości, tzn. w chwili wdrożenia. Niebędne jest zatem określenie możliwych zmian w przyszłości, już w momencie przystąpienia do projektowania systemu APD. Wymaga to dokładnego rozpoznania planów wieloletnich rozwoju przedsiębiorstwa pod kątem nowych inwestycji, wzrostu produkcji, ilości asortymentów wyrobów itd.

Autor poza tym tylko sygnalizuje cele, nie omawiając szczegółowo (właśnie w tym rozdziale) przygotowania organizacyjnego przedsiębiorstwa przez samo przedsiębiorstwo, które najbardziej powinno być zainteresowane we wdrożeniu systemu APD.

W rozdziale drugim godnym uwagi i rozpowszechnienia jest zaprezentowanie i szczegółowy opis wykorzystania szablonu graficznych symboli (rozbudowany standard ECMA) do opisu procesów przetwarzania danych. To samo dotyczy prezentacji odwzorowywania danych w języku COBOL.

Ujednolicenie metod przedstawiania wszelkiego rodzaju opisów jest jedną z najważniejszych i najpotrzebniejszych

*) Warszawa, PWN, s. 354, ark. wyd. 25,2

Cena 15 zł.–


PRODUKCJA
SPRZEDAŻ
SERWIS

MERA·ELWRO

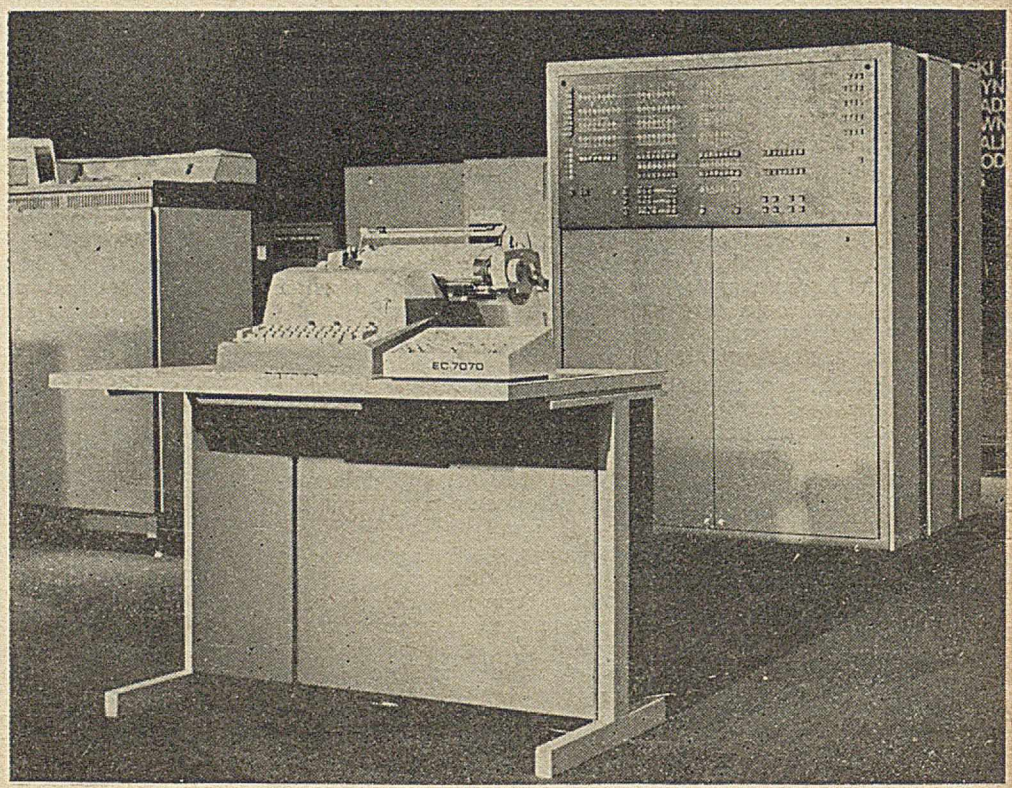
# WROCŁAWSKIE ZAKŁADY ELEKTRONICZNE „MERA ELWRO"

produkują nowoczesne systemy komputerowe EC-1030 ODRA 1305 ODRA 1325 usprawniające

- obliczenia naukowo-techniczne
- przetwarzanie danych
- sterowanie procesami technologicznymi

oraz dostarczają je na zasadzie generalnych dostaw, które obejmują:

- projektowanie i kompletowanie systemów cyfrowych
- projekty budowlano-organizacyjne ośrodków
- szkolenie personelu użytkowników m.c.
- serwis oprogramowania
- dostawę i uruchamianie zestawów komputerowych
- opiekę gwarancyjną i pogwarancyjną
- konsultacje w zakresie problemów związanych z tworzeniem i funkcjonowaniem ośrodków obliczeniowych

ul. Ostrowskiego 32, 53-238 Wrocław, tel. 478-27, 477-41, teleks 034518

*WCT/500/K/75*