

6 1979

P.1877/79



informatyka

W NUMERZE

Strona

Druga generacja komputerów Jednolitego Systemu <i>Bolesław Gliksman</i>	1
Automatyczna odnowa pracy systemu czasu rzeczywistego dla zestawu ODRA 1325/SMA <i>Józef Lewoc, Marek Mlek, Janusz Nawojski, Mieczysław Rozent</i>	4
Modularyzacja. Część 1. Pojęcie modułu <i>Piotr Strzałkowski</i>	7
TEED — program do kontekstowe ¹ edycji zbiorów <i>Krzysztof Biesaga</i>	11
Systemy komputerowe ze strukturami nadmiarowymi <i>Janusz Biernat</i>	13
Czy oprogramowanie będzie prawnie chronione? <i>Jan Waluszczyński</i>	14
Informatyka w Hiszpanii <i>Witold Trylik</i>	16
RETD — ogólnodostępna sieć transmisji danych <i>Juliusz Kulikowski</i>	19
ETOB — CENTRUM	
O cenach za wdrażanie systemów użytkowych	20
PORTRETY ZAWODOWE	
Ewa Frąckowiak <i>Wincenty Łada</i>	22
Z KRAJU	
Klub Użytkowników Minikomputera MERA 400 (W.K.)	23
Udane, tyle że nie „targi” (K.B.)	24
Szkoła Naukowa „Sieci Komputerowe” (B.M.)	25
Z WIZYTA W EUROPEJSKICH OŚRODKACH	
Uczelniane ośrodki obliczeniowe <i>Janina Mincel</i>	26
Marzenia szalonego studenta <i>Jarosław Deminet</i>	27
ZE SWIATA	
EuroIFIP <i>Władysław Klepacz</i>	28
Największy na świecie <i>oprac. A. Nawrocki</i>	29
OŚRODKI INFORMATYKI PREZENTUJĄ	
Zastosowanie pakietu programów SOD w CPN Poznań <i>Rafał Łykowski</i>	31
ZE ZJEDNOCZENIA INFORMATYKI	
Tenis po raz drugi (K.B.)	31
NAUCZANIE I KSZTAŁCENIE	
Informatyka na Farmacji <i>Maciej Paprocki</i>	32
DYSKUJE, POLEMIKI, POGLĄDY	
Czekając na Napoleona <i>Aleksander Lesz</i>	33
NASZE RECENZJE	
Teleinformatyka na półkach księgarskich <i>Juliusz Kulikowski</i>	34
PROBLEMATYKA BAZY DANYCH	
Główne kierunki rozwoju zastosowań relacyjnego modelu baz danych <i>Bolesław Szymański</i>	36
GIEŁDA	
Systemy minikomputerowe	39
MERA ELWRO	40
Wskaźniki dla autorów	III str. okład



ul. Świętokrzyska 14a
00-950 Warszawa
skrytka pocztowa 1004

KOLEGIUM REDAKCYJNE

Redaktor naczelny: prof. dr hab. Leon ŁUKASZEWICZ
mgr Krystyn BERNATOWICZ, prof. dr hab. inż. Konrad FIAŁKOWSKI (zastępca redaktora
naczelnego), mgr Janusz GWIAZDA, dr inż. Marek HOLYŃSKI, mgr inż. Stanisław
JASKÓLSKI, Władysław KLEPACZ (zastępca redaktora naczelnego), mgr Stanisław MROZIK,
dr inż. Tomasz PAWLAK, Sekretarz redakcji: Anna GLUTH-NOWOWIEJSKA.
Red. techn.: EWA KAMIŃSKA

RADA PROGRAMOWA

Prof. dr hab. Tadeusz PECHE (przewodniczący), mgr inż. Tomasz BANKOWSKI (sekretarz),
mgr inż. Antoni BOSSOWSKI, mgr inż. Roman BURNO, prof. dr hab. Andrzej JANICKI,
mgr inż. Jan KRAMARCZUK, prof. dr hab. inż. Juliusz KULIKOWSKI, prof. dr hab. Leon
ŁUKASZEWICZ, gen. dr inż. Marian PASTERNAK, mgr inż. Bronisław PIWOWAR, mgr
Zbigniew SUBSTYK, mgr Jerzy TRYBULSKI, doc. dr hab. Tadeusz WALCZAK, dr inż.
Jan ŻYDOWO

Redakcja: 00-041 Warszawa, ul. Jasna 14/16, pokój 331, tel. 27-71-40 lub centrala 26-82-61 w. 285, dyżury redakcji 10.00—13.00

Zakł. Graf. „Tamka”. Zam. 195. Papier druk. sat V kl. 70 g. A1. Obj. 5 ark. druk. Nakład 7300. C-119.

Cena egzemplarza zł 25.—

INDEKS 36124

Prenumerata roczna zł 300.—



ORGAN KOMITETU INFORMATYKI, MINISTERSTWA NAUKI, SZKOLNICTWA WYŻSZEGO
I TECHNIKI ORAZ KOMITETU NAUKOWO-TECHNICZNEGO NOT DS. INFORMATYKI

BOLESŁAW GLIKSMAN
ZETO Katowice

Druga generacja komputerów Jednolitego Systemu

Komputery Jednolitego Systemu pracują już od pięciu lat. Kraje socjalistyczne, które podjęły współpracę w dziedzinie elektronicznej techniki obliczeniowej, stoją w obliczu zasadniczych jakościowych zmian w zastosowaniach informatyki, polegających na tworzeniu sieci obliczeniowych.

Przedsięwzięciami w zakresie rozwoju sprzętu i zastosowań kieruje Międzyrządowa Komisja ds. współpracy krajów socjalistycznych w dziedzinie elektronicznej techniki obliczeniowej. Ma ona swój stały organ w Moskwie w postaci Centrum Koordynacyjnego. Pracują tam wysoko wykwalifikowani specjaliści z 8 współpracujących krajów. Centrum Koordynacyjne — wspólnie z Radą Ekonomiczną, Radą Głównych Konstruktorów, Radą ds. Zastosowań ETO i Radą ds. Kompleksowej Obsługi Sprzętu ETO — przygotowuje plany współpracy i przedkłada je do zatwierdzenia przez Komisję Międzyrządową.

Funkcje stałego przewodniczącego tej Komisji sprawował od chwili jej powołania M. E. Rakowski — zastępca przewodniczącego Gosplanu ZSRR; ze strony Polski zasiada w Komisji S. Paszkowski, wiceminister Przemysłu Maszynowego, wraz z delegatami rządów Bułgarii, Czechosłowacji, Kuby, NRD, Rumunii i Węgier.

W Centrum Koordynacyjnym znajduje się komputer obsługujący bazę danych, zawierającą informacje o próbach przeprowadzanych na nowych modelach sprzętu JS i o niezawodności elektronicznych maszyn cyfrowych eksploatowanych w krajach uczestniczących w porozumieniu o współpracy w dziedzinie eto.

W ciągu minionych ośmiu lat, od chwili rozpoczęcia działalności Komisji Międzyrządowej, zanotowano znaczne osiągnięcia w postaci wytworzenia 11 typów komputerów i 160 urządzeń towarzyszących oraz opracowania 15 podstawowych systemów EPD, wśród których wymienić należy systemy:

- technicznego przygotowania produkcji (BRL)
- planowania prac i środków (CSRS)
- zarządzania bankiem danych (NRD)
- kosztów własnych wyrobów (PRL)
- kontroli jakości wyrobów (SRR)

- gospodarki środkami trwałymi (WRL)
- operatywnego zarządzania produkcją podstawowych wyrobów (ZSRR).

W formie typowych projektów opracowano systemy EPD dla zakładów przemysłu maszynowego i handlu wewnętrznego. Prowadzone z inicjatywy Komisji badania ekonomiczne dają podstawę do stwierdzenia, że zastosowanie informatyki pozwala na 3—5-krotne skrócenie czasu projektowania i konstruowania wyrobów przemysłu maszynowego, 5—10-krotne skrócenie czasu projektowania procesów technologicznych oraz zmniejszenie nakładów inwestycyjnych w budownictwie o 10 do 30% przy równoczesnej poprawie wskaźników techniczno-ekonomicznych projektów o 10 do 25%.

Rada Głównych Konstruktorów opracowała przed kilkoma laty program dalszego rozwoju sprzętu JS EMC. W wyniku realizacji programu powstała druga generacja tego sprzętu, który miał spełnić następujące podstawowe założenia:

- osiągnięcie lepszych wskaźników techniczno-ekonomicznych
- zapewnienie pełnej kompatybilności programowej i sprzętowej z pierwszą generacją
- rozszerzenie własności funkcjonalnych i polepszenie parametrów eksploatacyjnych podstawowych systemów operacyjnych.

Nowa rodzina maszyn, nazywana czasem generacją 3 1/2, potocznie jest określana RIAD 2. Wykazuje ona podobieństwo do komputerów serii IBM 370 w takim stopniu, w jakim RIAD 1 podobna jest do serii IBM 360.

Komputery rodziny RIAD 2 charakteryzują się większą pojemnością pamięci operacyjnej, wirtualną strukturą pamięci, wyposażeniem w kanały multipleksorowe blokowe i znacznie zmodernizowane urządzenia zewnętrzne, a także znacznie unowocześnioną technologią produkcji. Dla komputerów RIAD 2 przewiduje się również szerokie zastosowanie zdalnego przetwarzania danych.

W tabeli 1 przedstawiono podstawowe dane techniczne nowych jednostek centralnych.

Tabela 1. Dane techniczne jednostek centralnych drugiej generacji JS EMC

	EC 1025	EC 1035	EC 1045	EC 1055 bez pamięci buforowej	EC 1055 z pamięcią buforową	EC1065
Producent	CSRS	BRL—ZSRR	ZSRR	NRD	NRD	ZSRR
Szybkość (tysiące operacji/s)	30—40	100—140	400—500	450	750	4000—5000
Czasy wykonania operacji (μ s)						
— krótkie operacje	5—18	2,6—4,5	0,6—2,2	0,6—3,9	0,3—2,2	0,12
— dodawanie stałoprzecinkowe	50—66	9,7	1,9—2,3	1,6—3,6	1,3—1,6	0,24
— mnożenie stałoprzecinkowe	95—220	23	2,8—3,4	3,4—5,2	3,1	0,6
— dzielenie stałoprzecinkowe	225—235	32	8,4—11	4,1—6,0	3,9	1,2
Pojemność pamięci operacyjnej:						
— pamięć główna (K bajty)	128—256	256—512	256—3072	256—4096	256—4096	1024—16384
— pamięć wirtualna (do 16 M bajtów)	TAK	TAK	TAK	TAK	TAK	TAK
— pamięć buforowa (8 K bajtów)	NIE	NIE	TAK	NIE	TAK	TAK

CHARAKTERYSTYKA MODERNIZACJI

Najbardziej istotną cechą komputerów RIAD 2, odróżniającą je od komputerów RIAD 1, jest wirtualna struktura pamięci głównej. Zaletą tego rozwiązania jest znaczna elastyczność pojemności pamięci operacyjnej, którą reguluje system operacyjny i która zawiera się w granicach między fizyczną pojemnością tej pamięci i obszarem wyznaczonym przez adresy wskazane w rozkazach. Ta część pamięci, która przekracza jej fizyczną pojemność, znajduje się w pamięci zewnętrznej o dostępie swobodnym.

Znaczne rozszerzenie pojemności pamięci operacyjnej zapewnia efektywne równoczesne wykorzystywanie wszystkich zasobów jednego komputera przez większą liczbę użytkowników. Dla każdego użytkownika sporządza się model funkcjonalny przyporządkowanego mu systemu komputerowego, czyli tzw. „maszynę wirtualną”. Każda „maszyna wirtualna” może pracować niezależnie pod nadzorem innego systemu operacyjnego. Dzięki równoczesnej pracy kilku „maszyn wirtualnych” wzrasta znacznie wydajność komputera, zwłaszcza w przypadkach zdalnego przetwarzania w trybie konwersacyjnym.

W związku z rozwojem pamięci zewnętrznych wzrosły wymagania co do przepustowości kanałów selektorowych. Zmieniając ich organizację logiczną, wprowadzono kanały blokowo-multipleksorowe. Ich konstrukcja pozwala na równoczesną współpracę i bardzo szybką wymianę danych z wieloma urządzeniami pamięciowymi.

W przypadku kanałów blokowo-multipleksorowych szybkość wymiany znacznie wzrosła w porównaniu do szybkości kanałów selektorowych, osiągając 3000 K bajtów/s. Ich konstrukcja dopuszcza podział każdego z nich na 256 podkanałów.

Istotnym elementem modernizacji jest również rozszerzenie możliwości zastosowań dzięki wprowadzeniu nowych lub znacznie ulepszonych urządzeń zewnętrznych, rozbudowie oprogramowania podstawowego i użytkowego oraz położeniu akcentu na konwersacyjne oraz zdalne przetwarzanie danych.

JEDNOSTKI CENTRALNE

Obecnie oferowane są do sprzedaży trzy typy komputerów RIAD 2, a mianowicie: EC 1025 produkcji ZPA Cakowice (CSRS), EC 1035 produkcji Zakładu im. Ordżonikidze w Mińsku (ZSRR) oraz EC 1055 produkcji kombinatu ROBOTRON (NRD). Modele te wykazują znaczne różnice w stosunku do swoich poprzedników.

EC 1025 różni się zasadniczo od konstrukcji EC 1021 w wyniku wyprowadzenia struktury modułowej. Wszystkie układy sterujące modułów zostały skonstruowane w sposób jednolity. Zawierają one procesor sterowany mikroprogramowo.

EC 1025 składa się z 8 następujących modułów:

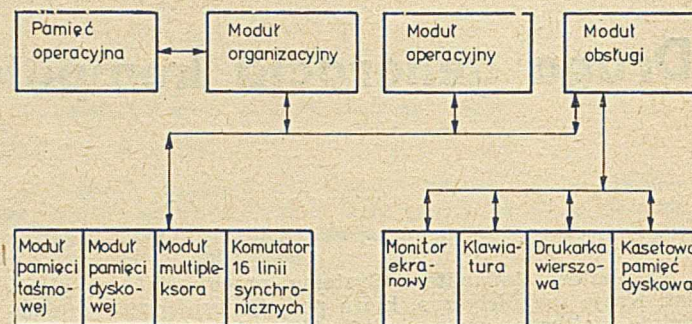
- 1) modułu operacyjnego (arytmometru) — do wykonywania operacji arytmetycznych i logicznych
- 2) modułu pamięci głównej — 2 bloki o pojemności po 128 K bajtów
- 3) modułu obsługi — do sprowadzenia mikroprogramów z pamięci dyskowej do pamięci sterującej poszczególnych modułów, sterowania urządzeniami peryferyjnymi oraz opracowania informacji o błędach
- 4) modułu organizacyjnego — do sterowania powiązaniem pomiędzy poszczególnymi modułami oraz pomiędzy modułami i pamięcią główną, a także do kontroli i eliminacji błędów w pamięci głównej

5) modułu pamięci dyskowej, która umożliwia bezpośrednio przyłączanie czterech jednostek tej pamięci o pojemności 100 M bajtów

6) modułu multipleksora, który łączy kanał standardowy z 36 podkanałami i 10 jednostkami sterującymi

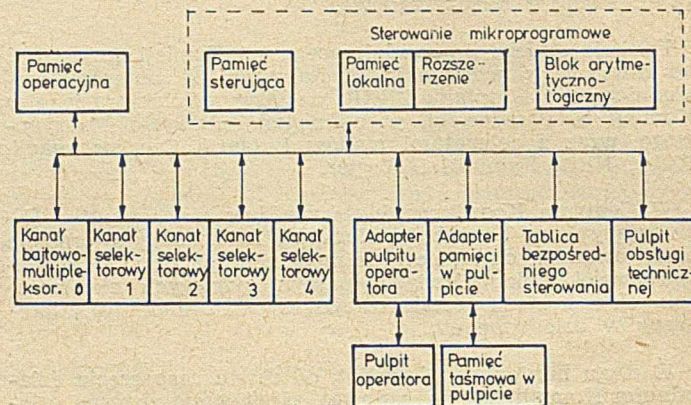
7) modułu pamięci taśmowej, który umożliwia bezpośrednio przyłączenie sześciu jednostek tej pamięci

8) modułu komunikacyjnego — do przyłączenia urządzeń zdalnego dostępu za pośrednictwem 16 linii działających w trybie pracy synchronicznej.



Rys. 1. Schemat struktury EC 1025

EC 1035 różni się od swego poprzednika poprzedniej generacji wprowadzeniem pamięci mikroprogramowej, z możliwością zmiany jej zawartości przez operatora, jak również możliwością automatycznej korekcji pojedynczych błędów w pamięci i automatycznego powtórzenia wykonania mikrorozkazu w przypadku błędnego zadziałania procesora. Komputer jest w pełni kompatybilny z maszynami pierwszej generacji oraz MINSKIEM-32.



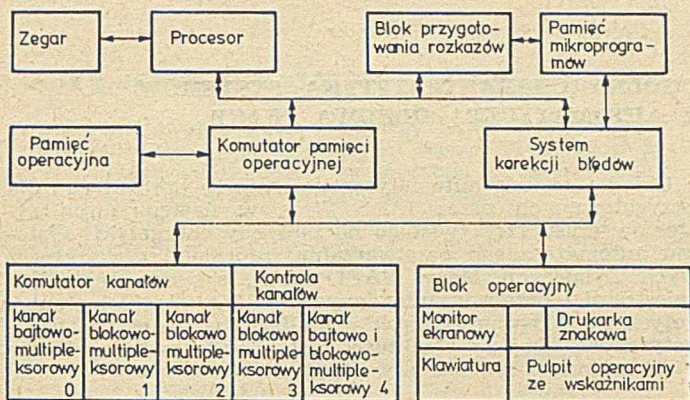
Rys. 2. Schemat struktury EC 1035

EC 1055 różni się od swego poprzednika (EC 1040):

- wprowadzeniem pamięci wirtualnej
- dwubajtowym połączeniem szyn i modułu sterowania
- monitorowym stanowiskiem operatora
- wprowadzeniem zegrowej kontroli przebiegu programów

— rejestracją wyników obliczeń
 — zwiększoną dokładnością operacji zmiennoprzecinkowych
 — rozszerzeniem repertuaru rozkazów
 — wprowadzeniem kanałów blokowo-multipleksorowych
 — rozszerzeniem systemu kontroli dzięki wprowadzeniu kodu korekcyjnego w pamięci operacyjnej.

Podobnie jak w pozostałych modelach drugiej generacji JS zastosowano sterowanie mikroprogramowe a procesor oraz pamięć operacyjna są zbudowane w technice obwodów scalonych wielkiej skali integracji.



Rys. 3. Schemat struktury EC 1055

Jednostka centralna EC 1055 (EC 2655) może być rozbudowana o następujące moduły:

● adapter kanał — kanał, umożliwiający połączenie dwóch jednostek centralnych EC 2655 lub jednostki EC 2655 z innymi jednostkami centralnymi rodziny RIAD 1 lub RIAD 2

● moduł macierzowy do wykonywania szybkich operacji w zmiennym przecinku.

URZĄDZENIA ZEWNĘTRZNE

I URZĄDZENIA DO REJESTRACJI DANYCH

Do parku urządzeń zewnętrznych wprowadzono nowe lub ulepszone rodzaje pamięci zewnętrznych (tab. 2) oraz urządzeń wejścia — wyjścia (tab. 3), a także skonstruowano nowe urządzenia do rejestracji danych (tab. 4). W urządzeniach tych znacznie zmodernizowano konstrukcję i technologię, co w istotny sposób wpłynęło na wzrost niezawodności eksploatacyjnej.

SYSTEMY OPERACYJNE

● System operacyjny OS/JS

Do dyspozycji użytkowników komputerów RIAD 2 przeznaczono system operacyjny OS/JS wydanie 6.0, w którym uwzględniono wymagania eksploatacyjne dostosowane do wymagań pamięci wirtualnej, kanałów blokowo-multipleksorowych oraz nowych typów urządzeń peryferyjnych.

Tabela 2. Pamięci zewnętrzne

Symbol	Rodzaj pamięci	Producent	Szybkość przesyłania (K bajty/s)	Pojemność (M bajty)
5001	Pamięć taśmowa	PRL	32/64	
5002	„	BRL, ZSRR	160/315	
5004	„	CSRS	64/126	
5025	„	ZSRR	64/126	
5061	Pamięć dyskowa z wymiennymi pakietami	BRL	312	29
5062	„	RRS	312	58
5066	„	ZSRR	806	100
5067	„	BRL	806	200
5091	Pamięć taśmowa kasetowa	PRL, CSRS		
5099	Pamięć dyskowa kasetowa	CSRS, PRL	90	1,5
5070	„	PRL	90	1,5
5075	Pamięć na dyskach elastycznych	CSRS	30	0,24
5064	Pamięć dyskowa ze stałymi głowicami	ZSRR	1600	11,2

Tabela 3. Urządzenia wejścia/wyjścia

Symbol	Rodzaj urządzenia	Producent	Cechy charakterystyczne
6019	Czytnik kart	ZSRR	1200 kart/min. (80- i 90-kol.)
6016	„	CSRS	1000 kart/min. (80- i 90-kol.)
6015	„	ZSRR	2 mechanizmy po 300 kart/min.
6113	„	NRD	160 kart/min. (dla terminali)
7904	Czytnik/perforator taśmy	CSRS	Czytanie 2000 zn/s, perforowanie 200 zn/s.
6041	Czytnik optyczny pisma	NRD	Formularze 210 × 297 mm, pismo OCR-B
6035	„	ZSRR	Szybkość czytania 500 zn/s
7941	Automatyczny stół kreślarski	CSRS	Pole 1189 × 1162 mm, szybkość 250 mm/s, krok 0,01 mm
6602	Wejście mikrofilmowe	NRD	Format 105 × 148 mm (72 klatki), repertuar 96 znaków, szybkość 1 KB/s
7602	Wejście mikrofilmowe	NRD	Reperuar 1920 znaków, szybkość zapisu 15 KB/s
7920	Alfaskop	współpraca	
7905	Grafoskop	ZSRR	Raster 1024 × 1024

Tabela 4. Urządzenia do rejestracji danych

Symbol	Rodzaj urządzenia	Producent	Cechy charakterystyczne
9080	Rejestrator danych na kartach	CSRS	Dziurkarka, sprawdzarka, opisywacz, szybkość opisywania 6 kolumn /s
9024	Rejestrator danych na taśmie papierowej	ZSRR	Dziurkarka, czytnik, maszyna do pisania 5 zn/s
9002	Rejestrator danych na taśmie magnetycznej	BRL	Możliwość wprowadzenia danych z kart, taśmy dziurkowanej lub klawiatury monitora ekranowego
9003	Wielostanowiskowy rejestrator danych na taśmie magnetycznej	BRL + CSRS	Centralna jednostka sterująca, 16 stanowisk
9150	Wielofunkcyjny rejestrator danych na taśmie magnetycznej	PRL	Minikomputer z kasetową pamięcią dyskową, 32 stanowiska
9110	Dwustanowiskowy rejestrator danych na dyskach elastycznych	BRL + CSRS	Wejście z klawiatury lub z czytnika taśmy z szybkością 20 zn/s blokami po 128 znaków, pojemność dysku ok. 3000 K bajtów

System operacyjny OS/JS 6.0 został znacznie rozszerzony w stosunku do poprzednich wydań. Jego nowe funkcje uwzględniają warunki pracy konwersacyjnej wieloprocesorowej oraz zdalnego dostępu.

● System operacyjny DOS/J

Modele (EC 1025, EC 1035 oraz EC 1045) wyposażone są w system operacyjny DOS/JS, wydanie 3. Uwzględniono w nim możliwość użycia danych i programów eksploatawanych pod nadzorem systemu operacyjnego DOS/JS komputerów poprzedniej generacji RIAD, a także przystosowanie do nowych rodzajów urządzeń zewnętrznych.

LITERATURA:

- [1] Computing Surveys, vol. 10, Nr 2, VI/78
- [2] Rechentechnik und Datenverarbeitung, Nr 8/78
- [3] Vyberinformaci z organizacni a wycetni techniky, Nr 3, 4, 5, 6/78

Automatyczna odnowa pracy systemu czasu

W Instytucie Automatyki Systemów Energetycznych (IASE) we Wrocławiu opracowano szereg systemów operacyjnych czasu rzeczywistego, pracujących na zestawach komputerowych ODRA 1325 i systemie modułów automatyzacji (SMA), przeznaczonych do wspomagania dyspozytorów podczas eksploatacji systemu elektroenergetycznego.

W trakcie uruchamiania i badań zestawów sprzętu stwierdzono dość wysoką awaryjność, spowodowaną w większości przypadków przez zatrzymanie jednostki centralnej po wykryciu błędu parzystości i przez utratę kontaktu SMA — jednostka centralna. Zatrzymania po błędzie parzystości bardzo często związane są z przekłamaniami transmisji lub pojedynczymi przekłamaniami w pamięci operacyjnej, spowodowanymi przez chwilowe zakłócenia. Stwierdzono również często, że po skasowaniu błędu parzystości, załadunku i ponownym uruchomieniu programów, system komputerowy pracuje poprawnie. W przypadkach utraty kontaktu SMA — jednostka centralna (co objawiało się zwykle zatrzymaniem bloku sterowania SMA w stanie niewłaściwym) wytworzenie sygnału zerowania SMA na ogół pozwalało na wznowienie poprawnej pracy zestawu.

Ręczne uruchamianie systemu komputerowego, dopuszczalne w systemach typu wsadowego, jest bardzo niewygodne dla systemów pracujących w czasie rzeczywistym. Bardzo często czas potrzebny na stwierdzenie błędu i ponowne uruchomienie systemu wynosi ponad 0,5 h.

Dla systemów komputerowych czasu rzeczywistego, przeznaczonych dla zastosowań przemysłowych, tak długie okresy nieoperatywności wywołanej przez dość częste zakłócenia przemijające są oczywiście niedopuszczalne. Traci się wtedy duże ilości informacji o ruchu obiektu, a efektywna dyspozycyjność systemu (określana jako stosunek czasu poprawnej pracy systemu do czasu pracy obiektu) znacznie maleje, mimo że zestaw sprzętu może pracować prawidłowo.

Ponadto konieczny jest ciągły nadzór nad pracą systemu komputerowego w celu szybkiego wykrywania stanów nieprawidłowych i ponownego uruchamiania (odnowy) systemu.

W celu zwiększenia dyspozycyjności systemu komputerowego w oddziale gdańskim IASE jednostkę centralną systemu wyposażono w układ automatycznego restartu. Układ i metodę realizacji automatycznej odnowy systemu opisano w [1]. Opierając się na tym rozwiązaniu, w Zespole Zakładów Automatyzacji Systemu Elektroenergetycznego IASE we Wrocławiu przygotowano własny układ i metodę odnowy pracy dla systemów komputerowych na zestawach ODRA 1325/SMA, pracujących pod nadzorem systemów operacyjnych, opracowanych przez IASE.

OGÓLNA CHARAKTERYSTYKA SYSTEMÓW Z AUTOMATYCZNĄ ODNOWĄ PRACY

Układ i metodę automatycznego restartu opracowano dla przygotowanych przez IASE Wrocław następujących systemów czasu rzeczywistego na potrzeby energetyki: systemu automatycznego przetwarzania informacji dla Okręgowych Dyspozycji Mocy (SAPI ODM), systemu centralnej rejestracji i przetwarzania danych (CRPD) dla bloku energetycznego, komputerowych symulatorów obiektów energetycznych. Ogólną strukturę systemów operacyjnych dla tych zastosowań opisano w [2, 3].

Przy budowie oprogramowania wykorzystano standardowy egzekutor EX2P Komputera ODRA 1325, uzupełniając go zorientowanym systemem operacyjnym [4]. Poszczególne człony takiego systemu w omawianym rozwiązaniu spełniają następujące funkcje:

- człon 2 (priorytetowy) realizuje podstawowe zadania związane ze współpracą z kanałem przemysłowym SMA
- człon 1 obsługuje punkt dyspozytorski oraz wstępnie przetwarza dane telemetryczne, a także koordynuje pracę systemu i przydziela zadania użytkowe
- człon 0 steruje i kontroluje pracę programów użytkowych.

W systemach operacyjnych SAPI i CRPD wykorzystano jedno pole dla programu pod kontrolą, natomiast w systemach operacyjnych symulatorów obiektów energetycznych zastosowano do tego celu dwa pola. We wszystkich tych systemach istnieje zewnętrzne urządzenie taktujące (zwykle zegar SMA), które inicjuje podstawowy cykl pracy co 5—10 sekund.

Projektując opisane wyżej systemy czasu rzeczywistego, przyjęto założenie, że należy stworzyć możliwość szybkiej odnowy ich pracy po awarii lub planowym wyłączeniu. Podstawowe informacje (telepomiar, telesygnal, czas, dane dotyczące struktury programowej) są cyklicznie zapisywane w pamięci zewnętrznej o bezpośrednim dostępie. Operator komputera, z uwagi na stosunkowo częste zakłócenia chwilowe, powinien mieć możliwość uruchomienia systemu na podstawie danych ściągniętych z pamięci o bezpośrednim dostępie. Opracowanie układu i metody automatycznego restartu pozwoliło na zautomatyzowanie, a w konsekwencji — znaczne przyspieszenie czynności operatorskich, co znacznie zwiększyło dyspozycyjność systemu.



Dr inż. Józef LEWOC dwukrotnie uzyskał tytuł magistra: w 1967 r. na Wydziale Elektrycznym Politechniki Wrocławskiej, w 1971 — na Uniwersytecie Wrocławskim. W 1973 r. uzyskał tytuł doktora na Politechnice Warszawskiej. W 1966 r. podjął pracę w WZE ELWRO, gdzie uczestniczył w realizacji kilku projektów dotyczących przemysłowych zastosowań komputerów. Od 1971 r. był zatrudniony w Instytucie Automatyki Systemów Energetycznych, obecnie pracuje w Centrum Obliczeniowym Politechniki Wrocławskiej.



Mgr inż. Marek MLEK w 1971 r. ukończył Wydział Elektroniki Politechniki Wrocławskiej i rozpoczął pracę w Instytucie Automatyki Systemów Energetycznych we Wrocławiu. Zajmuje się konstrukcją specjalizowanych urządzeń końcowych i pośredniczących dla maszyn cyfrowych.

rzeczywistego dla zestawu ODRA 1325/SMA

ZASADY INICJOWANIA AUTOMATYCZNEGO RESTARTU

Zasady inicjowania automatycznego restartu opracowano po przeanalizowaniu następujących typowych sytuacji zakłóceń przemijających, jakie stwierdzono podczas uruchamiania i badania systemów operacyjnych:

- błąd parzystości jednostki centralnej — powoduje całkowite zatrzymanie komputera, w związku z czym nie następuje generowanie sygnałów poprawnej pracy systemu
- „zatrzymanie” SMA (okresowe unieruchomienie bloku sterowania SMA) polega na braku przerw przychodzących do jednostki centralnej z SMA; w sytuacji tej nie są inicjowane podstawowe cykle pracy systemu
- „zatrzymanie” pracy układów współpracy z telemekanicą, pozwalające realizować podstawowe cykle pracy, lecz na ograniczonych strumieniach informacji wejściowych
- błędy programów użytkowych; w przypadku błędnych pętli lub zbyt długiego czasu wykonywania programu użytkowego następuje blokada komputera, polega na niedopuszczeniu do wykonania innych programów o tym samym priorytecie bezwzględny; sytuacje takie występują najczęściej w nowych programach użytkowych, które mają najniższy priorytet.

Powstanie każdej z wymienionych sytuacji prowadzi do zatrzymania całego systemu lub co najmniej do nieefektywnego jego działania. Dwa pierwsze zatrzymania są oczywiście najgroźniejsze — system traci kontakt z obiektem i w takim przypadku odnowa pracy jest niezbędna. W pozostałych przypadkach odnowa pracy systemu jest celowa (choćby częściowe informacje o stanie systemu są w dalszym ciągu rejestrowane).

Po przeanalizowaniu powyższych sytuacji przy realizacji układu i metody automatycznego restartu przyjęto następujące zasady:

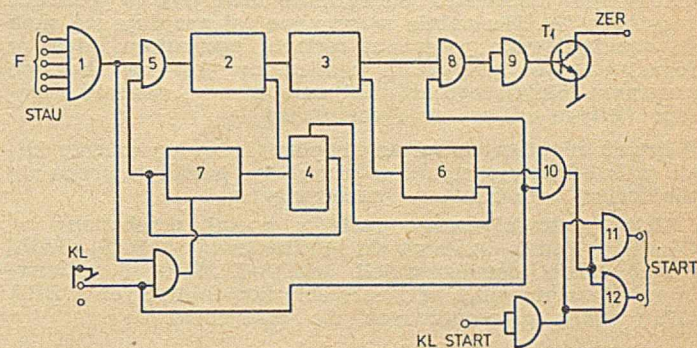
- w stanie pracy normalnej system operacyjny będzie cyklicznie wysyłał sygnał poprawnej pracy do układu automatycznego restartu
- niezbędne dane opisujące obiekt (najczęściej telepomiar, telesygnaly) oraz aktualna struktura programów użytkowych (tablice i wskaźniki programowe) będą zapisywane cyklicznie do pamięci zewnętrznej o bezpośrednim dostępie
- po restarcie zainicjowanym przez układ automatycznego restartu z pamięci zewnętrznej zostanie ściągnięty obraz pamięci operacyjnej w stanie normalnym (EX2P + system operacyjny), po czym system zostanie uruchomiony od wejścia restartowego (na danych zapisanych w pamięci zewnętrznej).

UKŁAD AUTOMATYCZNEGO RESTARTU

Przy projektowaniu układu automatycznego restartu przyjęto następujące założenia:

- układ powinien mieć prostą konstrukcję oraz powinien przyłączać się do jednostki centralnej
- do identyfikacji poprawnej pracy systemu należy wykorzystywać sygnały możliwe do wytworzenia programowo, ale nie generowane podczas normalnej pracy systemu
- w przypadku wykrycia przekłamania układ automatycznego restartu powinien wytwarzać sygnał zerujący układy jednostki centralnej oraz urządzenia zewnętrzne, a następnie sygnał uruchamiający pracę programu automatycznej odnowy w taki sposób, by wymagane zmiany w standardowym egzekutorze maszyny cyfrowej były minimalne
- układ powinien zawierać obwody ponawiające kilkakrotnie inicjowanie odnowy pracy systemu komputerowego jeśli pierwsza próba okaże się nieudana.

Założenie to zrealizowano w układzie, którego schemat logiczny przedstawiono na rysunku 1.



Rys. 1. Układ automatycznego restartu

Jako sygnał poprawnej pracy systemu przyjęto kod instrukcji 140, który nie jest wykorzystywany w żadnej maszynie cyfrowej serii ODRA 1300. Sygnał ten powinien pojawiać się w nastawianym okresie od 1 do kilkunastu sekund. Układ automatycznego restartu deszyfruje wspomniany kod w rejestrze F jednostki centralnej ODRA 1325 i strobuje go sygnałem STAU. Zdeszyfrowanie kodu 140 na elemencie 1 powoduje wytworzenie sygnału logicznej jedynki na wyjściu elementu 2. Jeśli kod 140 zostanie ponownie zdeszyfrowany przed upływem tego czasu, to wyjście elementu 2 utrzyma się w pozycji logicznej jedynki przez następny okres.



Mgr inż. Janusz NAWOJSKI w 1971 r. ukończył Wydział Elektryczny Politechniki Wrocławskiej i rozpoczął pracę w Instytucie Automatyki Systemów Energetycznych we Wrocławiu. Zajmuje się oprogramowaniem systemów komputerowych czasu rzeczywistego na potrzeby energetyki.



Mgr Mieczysław ROZENT w 1972 r. uzyskał tytuł magistra matematyki na Uniwersytecie Wrocławskim i rozpoczął pracę w Instytucie Automatyki Systemów Energetycznych. Specjalizuje się w oprogramowaniu systemów komputerowych czasu rzeczywistego.

W przypadku awarii systemu komputerowego przestaje pojawiać się kod 140. Po upływie zadanego czasu na wyjściu elementu 2 pojawia się zero logiczne, wyzwalając uniwibrator 3. Jednocześnie wyjście elementu 2 ustawia „0” logiczne na przerzutniku 4, co powoduje ponowne wyzwolenie układu 2 przez bramkę 5. Po zaniknięciu impulsu na wyjściu uniwibratora 3 (po ok. 1 s) zostaje wyzwolony uniwibrator 6 na ok. 1 s. Powoduje to zgaszenie przerzutnika 4 i zwiększenie zawartości licznika 7 o 1.

Po wyzwoleniu uniwibratora 3 przez bramkę 8 oraz 9 zostanieysterowany tranzystor T1, powodując wysłanie sygnału ZER do jednostki centralnej.

Wyzwolenie uniwibratora 6 powoduje wytworzenie sekwencji sygnałów przez bramki 10, 11, 12, analogicznej do sygnałów powstających po naciśnięciu klawisza START.

Jeśli po restarcie nie pojawi się kod 140, tzn. system nie zacznie poprawnie pracować, to po zwiększeniu zawartości licznika 7 o 1 wysłana zostanie ponownie sekwencja sygnałów ZER, START.

Układ wykonuje 9 prób inicjowania odnowy systemu i w przypadku niepowodzenia zatrzymuje się. Na płycie układu automatycznego restartu znajduje się klawisz KL, umożliwiający wyłączenie układu.

Omówiony powyżej układ automatycznego restartu spełnia wspomniane założenia projektowe, ponieważ:

- ma prostą konstrukcję (mieści się na jednej płycie drukowanej w jednostce centralnej i jest połączony z rejestrstem F)
- sygnał informujący o poprawnej pracy systemu jest łatwy do wytworzenia (użycie w programie nielegalnego kodu rozkazowego)
- wytworzona sekwencja sygnałów zerowania ogólnego maszyny (ZER) przyczynia się do zlikwidowania skutków przekłamań w urządzeniach, a sygnał START powoduje przekazanie sterowania do rozkazu zawartego w komórce w adresie *43. Normalnie w komórce tej znajduje się skok do standardowego programu restartu egzekutora, który w rozpatrywanym przypadku zamieniany jest na skok do programu inicjującego automatyczną odnowę systemu liczącego (jedyna zmiana w egzekutorze)

• licznik nieudanych restartów pozwala na wykonanie kilku prób odnowy systemu w przypadku przedłużającego się zakłócenia o charakterze przemijającym.

Układ automatycznego restartu został przebadany w Ośrodku Badań Automatyki Systemowej (OBAS) IASE przy udziale przedstawicieli MERA-ELWRO i zatwierdzony do stosowania w systemach komputerowych ODRA 1325 + SMA, przeznaczonych dla energetyki.

METODA AUTOMATYCZNEGO RESTARTU

Składowanie informacji opisujących obiekt

Podstawowymi informacjami opisującymi stan obiektu są zbiory aktualnych i uśrednionych wartości pomiarowych, zbiory stanów elementów dwustanowych (topologia obiektu) oraz zbiory masek sygnałów dwustanowych (opisujących sprawne nadajniki sygnałów dwustanowych). Przy realizacji systemów przyjęto, że informacje te będą dostępne dla programów użytkowych w określonych miejscach pamięci operacyjnej. Założenie to miało na celu poprawę efektywności realizacji procedur użytkowych (eliminacja zbędnych transmisji z/do pamięci zewnętrznych). Taki układ informacji w pamięci operacyjnej pozwala na łatwe ich przechowywanie dla celów odnowy pracy systemu.

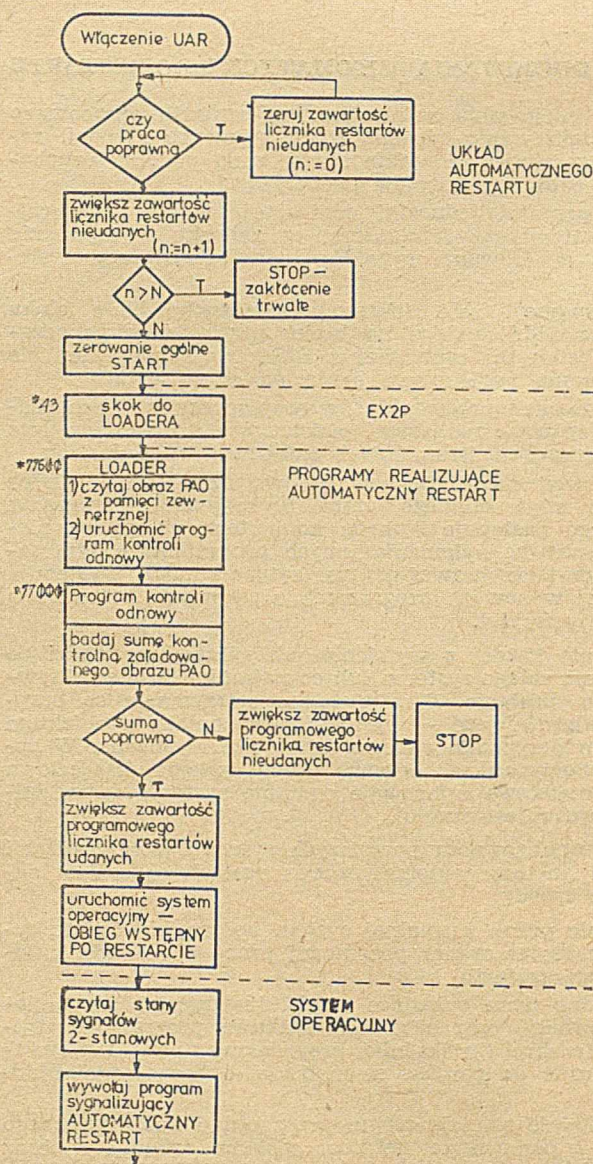
Przy projektowaniu automatycznego restartu częstotliwości przechowywania dobrano w taki sposób, aby z jednej strony restart wprowadzał jak najmniejsze zakłócenia w jakości pracy systemu, a z drugiej strony samo przechowywanie informacji nie powodowało nadmiernego ograniczenia mocy obliczeniowej. Decydującym czynnikiem w zasadzie było to, aby wartości uśrednione wielkości pomiarowych nie wykazywały po restarcie zbyt dużych skoków. W omawianych systemach wartości te są wartościami średnimi za okres 1 lub 2 minut. Stosownie do tego przyjmuje się okresy przechowywania 1 lub 2 minuty. Wielkości cyfrowe są mniej istotne, ponieważ po odnowie pracy systemu zwykle możliwe jest odzyskanie ich z obiektu w stosunkowo krótkim czasie.

Omawiane systemy pozwalają na dość elastyczne tworzenie różnych powiązań między programami użytkowymi. Powiązania te mogą się zmieniać zależnie od stanu obiektu

(np. przy pracy bloku energetycznego w stanie niestabilnym wywołuje się inne ciągi programów niż przy pracy w warunkach normalnych). Informacje definiujące powiązania między programami są zawarte w pamięci operacyjnej i są przechowywane wraz z informacjami opisującymi stan obiektu.

Sposób odnowy pracy systemu liczącego

Schemat metody automatycznego restartu przedstawiono na rysunku 2.



Rys. 2. Schemat metody automatycznego restartu

Po wytworzeniu sygnałów ZER i START sterowanie przekazane zostaje do rozkazu w komórce o adresie *43 (standardowy restart EX2P). W wersji z automatycznym restarcem do komórki tej wpisuje się rozkaz skoku do programu ładującego, znajdującego się w ostatnim bloku pamięci operacyjnej (blok ten może być zabezpieczony przed zniszczeniem). Program ten wprowadza obraz pamięci operacyjnej z pamięci zewnętrznej i uruchamia program badania poprawności wprowadzonego obrazu. Program kontroli bada sumę kontrolną informacji stałych (EX2P, system operacyjny), nie sprowadza natomiast sumy kontrolnej danych zmiennych (nie jest to możliwe, gdyż zmiana informacji może nastąpić w trakcie przechowywania). Błąd sumy kontrolnej powoduje zatrzymanie jednostki centralnej w wyniku niewytworzenia sygnału poprawnej pracy systemu. Przy prawidłowej sumie kontrolnej następuje uruchomienie systemu operacyjnego od wejścia restartowego. Egzekutor był przechowywany w stanie aktywnym i dlatego nie dostrzega przerw w pracy po automatycznej odnowie pracy systemu (w ten sposób uniknięto konieczności modyfikacji EX2P).

System operacyjny uzupełnia informacje o obiekcie o aktualny stan sygnałów dwustanowych, po czym przechodzi do pracy normalnej, podczas której generowany jest sygnał poprawnej pracy. Po restarcie, jako pierwszy z programów użytkowych wywoływany jest program rejestrujący fakt odnowy systemu i wysyłający odpowiednie informacje dla obsługi systemu.

Restarty udane i nieudane są rejestrowane w pamięci operacyjnej jako informacje dla konserwatorów sprzętu.

ZASTOSOWANIE I UZYSKANE EFEKTY

Opisano układ i metodę automatycznego restartu zastosowane dotychczas w systemach operacyjnych:

- SAPI ODM Warszawa (wersja bębnowa)
- SAPI ODM Poznań (wersja dyskowa)
- CRPD Turów (wersja bębnowa).

Komputer ODRA 1325, wykorzystywany w OBAS IASE do uruchamiania różnych wersji systemów (bębnowych i dyskowych), również wyposażono w układ automatycznego restartu. Zarówno w czasie badań systemów w OBAS IASE, jak i w poszczególnych obiektach, stwierdzono bardzo dużą przydatność rozwiązania.

Pierwsza wersja systemu operacyjnego, uruchomiona w ODM Warszawa w grudniu 1977 roku, pracuje bez przerwy do chwili obecnej. Automatyczny system występuje do kilku razy na dobę, co przy pracy ciągłej zmuszałoby do stałego nadzoru operatorskiego i istotnie obniżyłoby dyspozycyjność systemu. Dzięki automatycznej odnowie pracy systemu użytkownicy (dyspozytorzy) praktycznie nie odczuwają przerw w pracy systemu.

Podobnie podczas długotrwałych prób systemu CRPD Tu, rów, przeprowadzonych w OBAS IASE, stwierdzono wy-

stępowanie restartów głównie z powodu „zatrzymań” SMA. Powodowano także w sposób sztuczny automatyczny restart przez zatrzymanie jednostki centralnej i podczas tych prób nie stwierdzono poważniejszych zakłóceń w działaniu systemu komputerowego.

Dodatkowe korzyści wynikające z zastosowania automatycznego restartu to łatwość ładowania egzekutora i systemu po normalnym zatrzymaniu. Po wprowadzeniu EX2P i systemu operacyjnego do pamięci zewnętrznej uruchomienie systemu jest operacją kilkusekundową. Korzystają z tego również programiści uruchamiający nowe wersje systemów i oprogramowania użytkowego.

Próby w OBAS IASE i eksploatacja w obiektach potwierdziły, że dla systemów czasu rzeczywistego, budowanych w oparciu o komputer ODRA 1325 oraz SMA, możliwe jest zastosowanie automatycznej odnowy pracy. Wykazały one również całkowitą poprawność działania zaprojektowanego układu i użytej metody odnowy pracy.

LITERATURA:

- [1] Czesnowski R.: Automatyczny restart komputerowego systemu sterowania i kontroli. Prace IASE, zeszyt 31, Wrocław 1977
- [2] Praca zbiorowa: „System operacyjny dla maszyny cyfrowej ODRA 1325, pracującej w systemie automatycznego przetwarzania informacji okręgowej dyspozycji mocy”. Prace IASE, zeszyt 27, Wrocław 1974. Prace IV Krajowej Konferencji Automatyki, Poznań 1974
- [3] Praca zbiorowa: „Opis ogólny systemu operacyjnego SOCER”. IASE, nr ewid. 63454, Wrocław 1975
- [4] „Egzekutory dla m.c. ODRA 1325”. Publikacja nr 1320101, ELWRO, Wrocław 1974

PIOTR STRZAŁKOWSKI

Instytut Badań Jądrowych (CYFRONET)
Świerk

Modularyzacja. Część 1

Pojęcie modułu

Przy tworzeniu dużych systemów oprogramowania powstaje konieczność ich podziału na części. Przyczynami takiego postępowania mogą być m.in.:

- parametry sprzętu; typowym przykładem jest tu ograniczona pojemność pamięci operacyjnej, określającą maksymalne rozmiary programów; w przypadku maszyny wieloprotocowej lub wieloprogramowej również należy uwzględniać współbieżność wykonywania zadań, co wymaga operowania mniejszymi jednostkami programowymi
- względy organizacyjne; w fazach projektowania, uruchamiania i eksploatacji oprogramowania podstawowymi wymaganiami są: przejrzystość i zrozumiałość dokumentacji, łatwość modyfikowania i konserwowania programów oraz możliwość podziału zadań pomiędzy różnych wykonawców; wymagania takie można spełnić dzieląc całe oprogramowanie na stosunkowo małe części
- predyspozycje umysłowe człowieka, który bardziej efektywnie rozwiązuje problemy mniej złożone.

Zagadnienie nie jest nowe i znalazło swój wyraz np. w językach programowania, umożliwiając podział programu na program główny i podprogramy. W ostatnich latach pojawiło się kilka nowych idei związanych z tą problematyką. Wynika to ze wzrostu zainteresowania częściowym i metodycznym ujęciem procesów projektowania, uruchamiania i eksploatacji oprogramowania. Spróbujmy więc spojrzeć bliżej na proces podziału systemu na części — nazwiemy go ogólnie modularyzacją — i przedstawić (w sposób raczej intuicyjny) wybrane zagadnienia związane z tą problematyką.

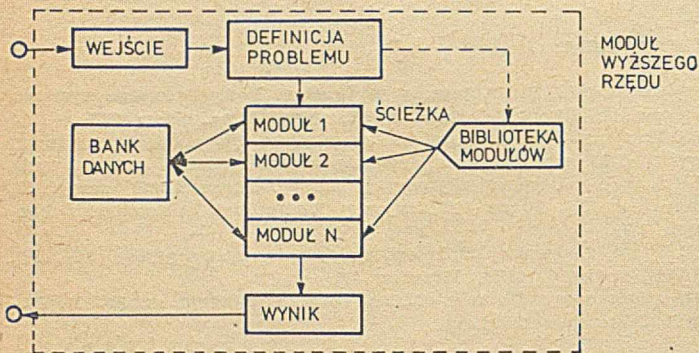
W niniejszym numerze INFORMATYKI przekażę kilka refleksji na temat kryteriów tworzenia modułów (części systemu) i definicji modułu, natomiast w następnym numerze postaram się omówić niekóre języki „modularne” i problemy z nimi związane.

Zauważmy przede wszystkim, że przy tak ogólnej definicji modularyzacji jaką podaliśmy wyżej („podział systemu na części”) słowo moduł nie ma jednoznacznej interpretacji. Pytanie „co to jest moduł?” wiąże się ściśle z pytaniem „jak podzielić projektowany (programowany) system na części?” Rozpatrzymy kilka możliwości.



Mgr Piotr Strzałkowski ukończył Wydział Matematyki i Mechaniki UW (sekcja numeryczna) w 1973 roku. Pracuje w Zakładzie Informatyki SCO CYFRONET w Świerku jako starszy asystent w pracowni języków programowania i metod numerycznych. Interesuje się metodologią i językami programowania.

Modularyzacja jest bardzo łatwa: piszę po prostu program o 3000 instrukcji, robię ciach? ciach! — i mam sześć modułów po 500 instrukcji — podejście to (zaprezentowane przez jednego z programistów US Air Force), a przytoczone przez Yourdanha [1] wyraża pogląd, że podział powinien być taki, by poszczególne części systemu nie przekraczały z góry ustalonej liczby instrukcji, a mianowicie takiej, która pozwala łatwo zrozumieć istotę działania programu. W przytoczonym przykładzie programista nie odwołał się do logicznej struktury programu, lecz — jak twierdzi — podzielił program mechanicznie. Nietrudno przewidzieć kłopoty wynikające z takiego podejścia, zwłaszcza przy opracowywaniu dużych, skomplikowanych systemów oprogramowania należy stosować podejście bardziej usystematyzowane. Stosunkowo dawno znane jest rozwiązanie polegające na podziale „logicznym”. Przedstawimy je według pracy [2], wychodząc od ogólnego schematu organizacji systemu modularnego (rys. 1).



Rys. 1. Ogólny schemat organizacji systemu modularnego

Moduł składa się z kilku elementów: wejścia i definicji problemu, biblioteki modułów niższego rzędu, banku danych oraz wyników wyprowadzanych na zewnątrz. Na podstawie danych wejściowych blok definicji problemu wybiera z biblioteki modułów moduły niższego rzędu i zestawia je w postaci tzw. „ścieżki” (za „ścieżkę” można uważać ustalenie przepływu sterowania między modułami). Określa również format i sposób korzystania z dodatkowych danych wejściowych, niezbędnych do wykonania obliczeń danego problemu. Moduły niższego rzędu kontaktują się poprzez bank danych.

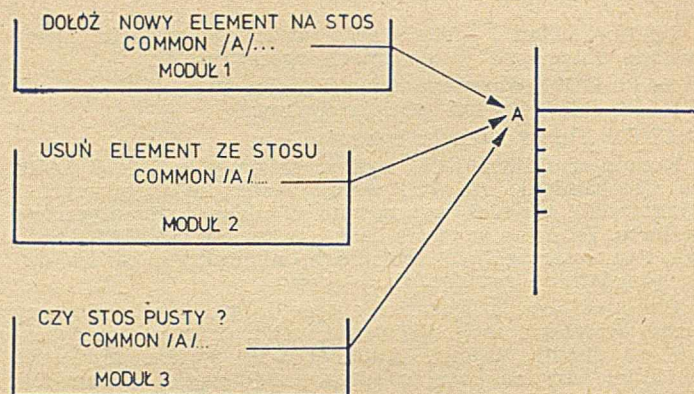
Rozpatrzmy teraz, jak może wyglądać taki moduł, gdy do jego realizacji jako język programowania wybierzemy FORTRAN. W przypadku najprostszym dla definiowania problemu posłużymy się programem głównym. Tworzy on „ścieżki” poprzez ciąg wywołań swoich podprogramów na podstawie danych wejściowych. Modułami są tu właśnie podprogramy, do których zaliczamy także podprogramy biblioteczne i funkcje standardowe, których dostarcza kompilator. Bankiem danych będą wspólne pola programu (COMMON), zbiór parametrów wywoływania podprogramów w pamięci operacyjnej, zbiór danych zawartych na dyskach, taśmach, kartach itp. Zauważmy, że również program główny jest modułem, lecz szczególnego rodzaju.

W wypadku FORTRANU modularyzację można więc sprowadzić do podziału programu na program główny i podprogramy, z uwzględnieniem informacji dostarczanych przez dane wejściowe i bank danych. Określenie „moduł niższego rzędu” sugeruje strukturę hierarchiczną modułów. Nie będziemy się tym teraz zajmować, zwrócimy jedynie uwagę, że jako podstawowy można przyjąć podział, które dokonaliśmy przed chwilą, oraz że „moduły wyższego rzędu” mogą składać się z programów głównych wraz z zestawem ich podprogramów (nie uwzględniamy tu niektórych elementów banku danych).

Pozostanmy jeszcze przy modularyzacji związanej z FORTRANEM. Wiadome jest, że wybór języka programowania oddziałuje na strukturę „ideową” projektu rozwiązania problemu, narzucając „punkt widzenia” wybranego języka.

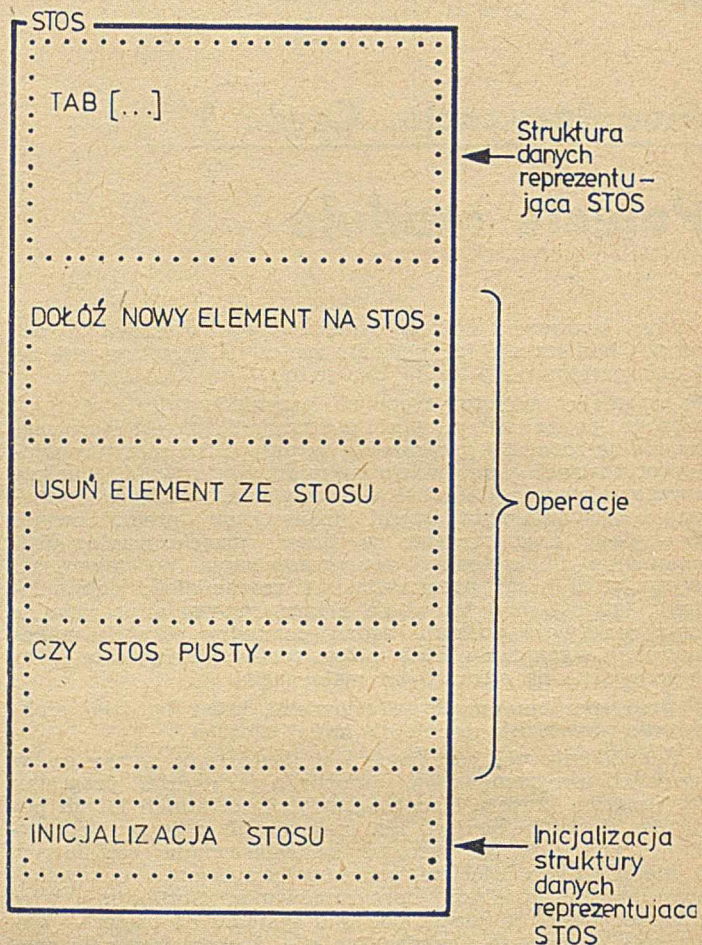
Struktura programu w FORTRANIE odpowiada strukturze schematu blokowego. Podprogramy reprezentują ciągi czynności, które można pogrupować w odpowiednich blokach schematu. Wykonanie takiego podprogramu odpowiada „wejściu” do bloku i wykonaniu czynności tam opi-

sanych. Moduły, będące programami, kontaktują się ze sobą za pomocą banku danych, w skład którego wchodzi na poziomie struktur programowych dane zawarte w wspólnych polach i parametry przekazywane przy wywołaniach. Moduły odwołują się do nich przez odpowiednie odnośniki w tekście. Przypuśćmy, że chcemy opisać stos i operacje na nim. Stos będzie reprezentować tablica we wspólnym polu o nazwie A. Odpowiednie moduły przedstawia rysunek 2.



Rys. 2. Opis stosu w FORTRANIE. Moduły reprezentują operacje na stosie. Tablica reprezentująca stos i odpowiednie wskaźniki znajdują się we wspólnym polu A. W każdym z modułów znajduje się odniesienie do tego pola (do tablicy i wskaźników)

Tak więc „punkt widzenia” FORTRANU wygląda następująco: moduły reprezentują operacje, ciąg czynności. Struktura danych, na której wykonywane są te operacje, w zasadzie znajduje się na zewnątrz modułu i konieczne są odwołania do niej w tekście modułu.



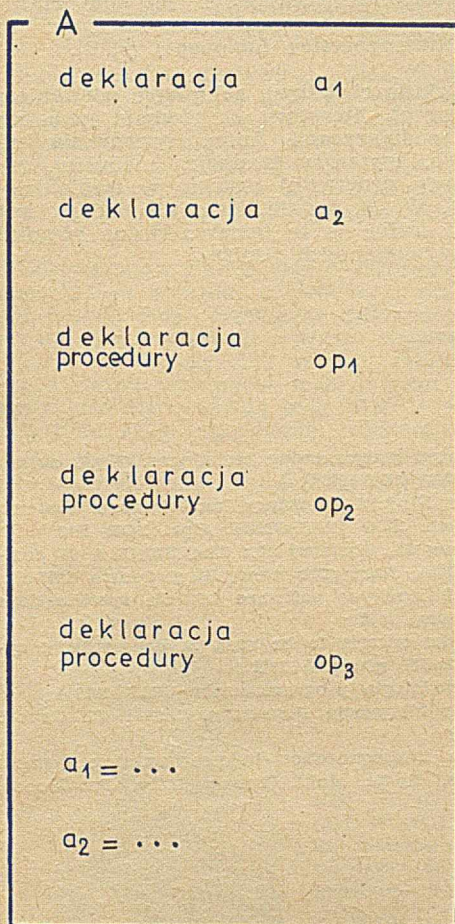
Rys. 3. Moduł „wyższego rzędu” reprezentujący stos

Sytuacja taka ma pewne wady. Wyobraźmy sobie, że chcemy zmienić zakres tablicy (co jest szczególnie niezbędne w przypadku FORTRANU), reprezentującej w naszym przykładzie stos, bądź zmienić nazwę wspólnego pola. Zmianą objęte zostaną wszystkie zainteresowane moduły. Biorąc pod uwagę rozwiązanie narzucone przez FORTRAN, spróbujmy spojrzeć na zagadnienie nieco inaczej. Czy istnieje np. język, w którego jednej jednostce programowej (jednym module) można by ująć łącznie strukturę danych i operacje na niej (ewentualnie także inicjalizację)? W FORTRANIE byłby to moduł „wyższego rzędu”, ale czy możliwa jest taka modularyzacja od razu na poziomie języka? W przypadku stosu moduł taki wyglądałby tak, jak to zostało przedstawione na rysunku 3.

Używając nomenklatury Naura, można powiedzieć, że w jednej jednostce programowej umieściliby strukturę danych i wiązkę czynności z nią związanych (czynności takie w przypadku FORTRANU były traktowane jako moduły). Obecnie interesować nas będą przede wszystkim moduły tego właśnie rodzaju. Przeprowadzając ponownie porównanie z FORTRANEM, zauważmy, że realizacja takiego modułu poprzez podprogram FORTRANU jest niemożliwa: powijając już brak możliwości zagnieżdżania tekstowego podprogramów, należy wziąć pod uwagę fakt, że w nowo skonstruowanym modelu interesuje nas dostępność struktury danych oraz wykonywanych na tej strukturze operacji po uprzedniej inicjalizacji, która jest przecież częścią składową modułu. Inaczej mówiąc, chcemy mieć dostęp do podprogramu po jego wykonaniu. Jest to niemożliwe w FORTRANIE.

Przypuśćmy, że mamy moduł o nazwie A oraz zadeklarowane w nim zmienne a_1 , a_2 (struktura danych) i procedury op_1 , op_2 , op_3 (operacje). W części inicjalizującej znajdują się instrukcje podstawienia wartości początkowych na zmienne a_1 i a_2 (rys. 4).

W FORTRANIE podprogram możemy jedynie wywołać, powodując jego wykonanie. Musimy tu rozgraniczyć dwie czynności: inicjalizację modułu oraz wykorzystanie którejsz z jego zmiennych czy procedur po tej inicjalizacji. Oznacza to więc, że potrzebne będą dwa typy instrukcji. Pierwsza z nich to instrukcja inicjalizacji, np. „inicjalizuj A”.



Rys. 4. Przykładowy moduł o nazwie A

Po jej wykonaniu zmienne a_1 i a_2 będą miały wartości początkowe i wraz z procedurami staną się dostępne w programie. Drugi typ instrukcji określa wykorzystanie tych zmiennych i procedur na zewnątrz modułu. Zwykle instrukcje takie odwołują się do nazwy modułu, np. $A.op_1$ oznacza wywołanie procedury op_1 z modułu A, $A.a_1$ oznacza zmienną a_1 z modułu A, np. w instrukcji „ $i - A.a_1 + 10$ ”, gdzie i jest zmienną spoza modułu. Zauważmy, że rzeczywiście sytuacja taka nie może zaistnieć w FORTRANIE.

Operację otrzymania przedstawionej jednostki programowej z elementów dostępnych w takim języku, jak FORTRAN, można porównać do „strukturalizacji” instrukcji (zresztą w przypadku skrajnym instrukcję można też uważać za moduł). Rzeczywiście, modularyzacja tego typu nazywana jest „strukturalizacją na poziomie makro” („inaczej wielką strukturalizacją”). Przypomnijmy sobie sposób utworzenia instrukcji iteracji *repeat* czy *while*. Zostały one zbudowane z instrukcji podstawienia, warunkowej i skoku. Istnieją jeszcze języki, w których są dostępne wszystkie wymienione typy instrukcji (choć skok stracił rację bytu) i często, w odczuciu ich użytkowników, są one równoprawne, choć iteracja jest instrukcją „na wyższym poziomie”. Tak też i w stosunku do modułów będziemy dopuszczać istnienie modułów w sensie FORTRANU (procedury), jak i modułów „wyższego rzędu”.

Przeprowadźmy jeszcze jedną analogię. Obserwując języki programowania pod kątem ich „stosunku” do struktury danych i operacji, można zauważyć, że w językach numerycznych typu ALGOL 60 czy FORTRAN przeprowadza się złożone operacje na stosunkowo prostych strukturach danych. Z drugiej strony, w językach baz danych proste operacje dotyczą złożonych struktur danych. COBOL można umieścić mniej więcej pośrodku. Przechodząc od koncepcji modułu typu fortranowskiego do postulowanej postaci modułu „wyższego rzędu”, przenosi się jednocześnie akcent z operacji — bliżej danych. Będzie to miało swoje konsekwencje w sposobach jego używania (np. fakt, że moduł można potraktować jako typ z określonymi operacjami — w przypadku modułu STOS będą to operacje DOŁÓŻ NOWY ELEMENT NA STOS itd. — oraz deklarować i używać zmienne tego typu — np. zmienne STOS 1 i STOS 2 typu STOS; korzystanie z operacji na stosie będzie mogło wyglądać następująco: np. STOS 1. DOŁÓŻ NOWY ELEMENT NA STOS po inicjalizacji zmiennej STOS 1).

Zanim przejdziemy do przykładu ilustrującego ideę modularyzacji, odpowiemy na pytanie, w jakim języku programowania można zrealizować moduł „wyższego rzędu” na poziomie języka. Jest nim SIMULA 67 wraz ze swoją koncepcją klasy [3] i większość języków modularnych (zostaną one przedstawione w następnym numerze INFORMATYKI) w większym lub mniejszym stopniu oparta jest na tym języku.

Dla ilustracji modularyzacji przedstawimy przykład, który zaczerpnięty jest z artykułów Parnasa [4, 5]. Dla danego zbioru zdań należy stworzyć zbiór ich przesunięć cyklicznych (wszystkich możliwych zdań utworzonych ze zdań oryginalnych przez uporządkowane — cykliczne — przeniesienie słów z początku na koniec tych zdań) i uporządkować go według alfabety.

Następujące moduły otrzymamy w przypadku metody „klasycznej” (według schematu blokowego):

1. Wejście. Wczytanie zdań i zapamiętanie ich dla przetwarzania przez nine moduły. Pamięta się miejsce, w których znajdują się początki zdań.
2. Przesunięcie cykliczne. Moduł jest wywoływany po zakończeniu wykonania modułu 1. Przygotowuje się informacje dotyczące początków przesunięć cyklicznych wszystkich zdań oraz zapamiętuje się je wraz z numerami zdań, którym przesunięcia odpowiadają.
3. Porządkowanie według alfabety. Moduł, wykorzystując tablice utworzone przez poprzednie moduły, tworzy tablicę podobną do tablicy z modułu 2. Odpowiednie początki (indeksy lub adresy) zdań są w niej uporządkowane według kolejności alfabetycznej słów tworzących zdania.
4. Wyjście. Moduł wypisuje przesunięcia cykliczne w porządku alfabetycznym, w oparciu o tablice modułu 1 i 3.
5. Sterowanie. Moduł steruje kolejnością wykonania poprzednich modułów (kolejność ta będzie odpowiadała kolejności, w jakiej zostały one opisane).

W przypadku metody „funkcjonalnej” (struktury danych i operacje grupuje się według ich funkcji w systemie w modułach „wyższego rzędu”) moduły będą wyglądały następująco:

1. Moduł przechowujący zdania. Danymi w tym module będą zdania, natomiast operacjami — dostęp do konkretnego znaku, podstawienie znaku w konkretne miejsce, usunięcie zdania, usunięcie słowa. Dostępna będzie informacja nt. liczby słów w zdaniu i znaków w słowie.

2. Moduł wczytujący. Czyta zdania i wykorzystuje operacje modułu 1 do ich zapamiętania. W tym celu dysponuje on operacją wykonującą te czynności.

3. Moduł tworzący przesunięcie cykliczne. Informacje i operacje są tu podobne do tych, które można uzyskać w module 1; powstaje wrażenie, że mamy do czynienia nie ze zbiorem zdań, lecz ze zbiorem wszystkich przesunięć cyklicznych.

4. Moduł porządkujący alfabetycznie. Moduł ten dostarcza uporządkowanych wskaźników przesunięć. Jego operacje porządkują je korzystając z informacji modułu 3.

5. Moduł wypisujący. Powoduje wydruk uporządkowanych przesunięć.

6. Moduł sterujący. Wyznacza kolejność wykonywanych operacji.

Gdyby wykonanie tego programu przedstawić w postaci graficznej, to w przypadku pierwszym strzałki idące z modułu do modułu oznaczałyby przejście sterowania (jak w schemacie blokowym). Moduły tego typu można utworzyć w FORTRANIE przez podział na program główny (moduł 5) i podprogramy. W przypadku drugim wygodniej byłoby przedstawić wykres powiązań modułów, gdzie strzałki oznaczałyby korzystanie z informacji (i operacji) [7].

Odtąd używając słowa „moduł”, będziemy mieć na myśli moduł zbudowany według podziału „funkcjonalnego”.

W trakcie projektowania, uruchomienia i użytkowania modułów pozostaje istotne, co jest dostępne na zewnątrz poszczególnych modułów, jakich operacji i informacji mogą one nam dostarczyć, oraz w jaki sposób można je identyfikować. Interesuje to zarówno projektanta, jak i użytkownika. Wracając do przykładu stosu, dla osoby wykorzystującej moduł opisujący stos może być całkowicie nieistotne jak jest on zrealizowany (za pomocą tablicy czy też listy). Chce ona tylko wiedzieć, jak może się do niego odwoływać oraz jakiego typu operacje może na nim wykonywać. Można więc przeprowadzić podział struktury danych i operacji modułu na te, które są dostępne na „zewnątrz”, i te, które są istotne dla implementacji modułu. Powstaną więc informacje czy operacje o nierównym dostępie (nazywa się to czasem „zasłanianiem informacji”) i, jak zobaczymy, koncepcja ta wywarła pewien wpływ na strukturę tworzonych języków modularnych.

W fazie projektowania interesuje nas najczęściej sformułowanie tego co program lub poszczególne części systemu mają realizować. Tego typu sformułowania znajdują wyraz w tzw. specyfikacjach, które pisze projektant. Dopiero na podstawie tych specyfikacji realizator systemu pisze programy, decydując tym samym, jak mają być zaimplementowane i wykonane idee projektanta. Interesować nas będzie szczególnie tzw. „specyfikacja niezupełna” modułu [6], tj. definicja modułu wyrażona w terminach pojęć dostępnych poza modułem (definicja, która nie odwołuje się do żadnych szczegółów ukrytych przez moduł).

Wspomniany już Parnas [4, 5] dla definiowania modułów wprowadza specyfikacje, które nazywa „funkcjami”¹⁾. Mogą one reprezentować zarówno informacje, jak i operacje dostępne na zewnątrz modułu. Każdy moduł składa się z zestawu „funkcji” o postaci następującej:

„Funkcja”:
Możliwe wartości:
Wartość początkowa:
Parametry:
Efekty:

¹⁾ Ponieważ nazwa ta jest nieco myląca, będziemy używać jej w cudzysłowie.

W punkcie pierwszym podaje się nazwę „funkcji”, w drugim zakres wartości, które „funkcja” może przyjmować. „Efekty” reprezentują żądanie tego, co „funkcja” ma wykonywać (jest to opis efektów jej wykonania). „Efekty” mogą zawierać tzw. „pułapki na błędy”, np. sprawdzenie czy parametry mają poprawne wartości.

W przykładzie o porządkowaniu alfabetycznym przesunięć cyklicznych zdań „Moduł przechowujący zdania” może składać się z następujących „funkcji”:

ZNAK — dostęp do konkretnego znaku
PODSTAWZNAK — podstawienie znaku w konkretne miejsce

USUNZD — usunięcie zdania
USUNSL — usunięcie słowa
ILSLOW — ilość słów w zdaniu
ILZNAKOW — ilość znaków w słowie

LINIEMAX — maksymalna ilość zdań (dopuszczalna).

„Funkcja” ILSLOW może wyglądać tak:
„Funkcja” ILSLOW
Możliwe wartości: całkowite, nieujemne
Wartość początkowa: 0

parametry: 1 — typu całkowitego (numer zdania)
Efekty: jeśli $1 \leq 0$ lub $1 > \text{LINIEMAX}$ to PISZ („BLAD”).

„Efekty” nie zawierają tu wyliczenia ilości słów w 1-tym zdaniu. „Funkcje” można wykorzystywać w innych „funkcjach” (np. ILSLOW wykorzystuje „funkcję” LINIEMAX). Można też na zewnątrz „funkcji” podstawiać na nie wartości. Tak więc w tym, że „efekty” zawierają jedynie „pułapkę na błędy”, a „funkcja” dostarcza nam informacji o ilości słów w zdaniu, nie ma paradoksu. Podstawienie następuje gdzie indziej (w innej „funkcji”) i w momencie gdy informacji potrzebujemy, jest ona „na swoim miejscu”.

Należy wspomnieć, że „funkcje” mogą korzystać także z „funkcji” innych modułów. Wtedy nazywa się je „odwozującymi”.

„Funkcje” Parnasa są jedną z wczesnych prób formalizacji specyfikacji dotyczących modułów. Inne próby doprowadziły do utworzenia specjalnych języków opisujących moduły i ich wzajemne związki. Przykładem może być MIL (*Module Interconnection Language*) [6, 8].

Pierwszą część artykułu chciałbym zakończyć odesłaniem Czytelników do szerszej literatury na temat modułów. Wiele problemów modularyzacji (m.in. rozważania nt. zmiennych globalnych, kryteriów tworzenia modułów, czy transformacji struktury programu) omawia w swojej nowej książce [6] prof. W. M. Turski. Tam też podana jest bardzo szeroka bibliografia na te tematy. Można również zajrzeć do pozycji wymienionych poniżej.

LITERATURA:

- [1] Yourdan E.: *Advanced programming techniques*. Wyd. autora z 1972 roku (I wyd. IAT CDC-1970)
- [2] Arkuszewski (red.): *Projekt struktury modularnego systemu programów reaktorowych. SCO CYFRONET*. Opr. ZOR IBJ
- [3] Dahl O.-J., Myhrhaug B., Nygaard K.: *The SIMULA 67 common base language*. Oslo, Norwegian Computing Centre 1967
- [4] Parnas D. L.: *A technique for software module specifications with example*. CACM maj 1972
- [5] Parnas D. L.: *On the criteria to be used in decomposing systems into modules*. CACM grudzień 1972
- [6] Turski W. M.: *Metodologia programowania WNT 1978*
- [7] Strzałkowski P.: *Modularyzacja Parnasa w Simuli 67*. Raport IBJ (w przygotowaniu)
- [8] Thomas J.: *Module interconnection in programming systems supporting abstractions*. Rozpr. dokt. Brown Univ. Providence, maj 1976

TEED – program do kontekstowej edycji zbiorów

Podstawowymi pamięciami masowymi dla maszyn cyfrowych ODRA, obok pamięci dyskowych, są przede wszystkim taśmy magnetyczne. Wśród bogatego asortymentu ich oprogramowania ważną rolę odgrywają programy zwane edytorami. Ułatwiają one użytkownikowi otrzymywanie zbiorów znakowych o żądanej strukturze i zawartości. Najbardziej popularnym edytorem maszyn cyfrowych ODRA serii 1300 jest program xKYA [6]. Daje on użytkownikowi wiele możliwości w zakresie tworzenia i poprawiania zawartości zbiorów zapisanych na taśmach magnetycznych. Edytor ten jednak pozwala jedynie na swobodne operowanie podzbiorem i rekordami, natomiast możliwości jego ingerencji w znakową strukturę zbiorów są bardzo małe.

Powstała w tym miejscu luka w oprogramowaniu maszyn cyfrowych ODRA nie jest wypełniona przez inne programy z biblioteki standardowej. Stąd też celowe było stworzenie narzędzia programowego, uzupełniającego funkcje programu XKYA w zakresie kontekstowej edycji zbiorów. Powstały program został nazwany edytorem kontekstowym TEED (TEExt EDitor).

Powstanie TEED poprzedziła dokładna analiza, w której wzięto pod uwagę potrzeby użytkowników, zakres funkcji udostępnianych przez rozbudowane edytory znanych firm [3, 4, 5] i wreszcie możliwości sprzętu, dla którego program miał być zaprojektowany.

Specyfika taśm magnetycznych sprawia, że zapisywane na nim zbiory można podzielić na trzy hierarchiczne poziomy: 1) zbiorów i podzbiorów, 2) rekordów, 3) znaków.

Zgodnie z tym podziałem, zbiór funkcji wszystkich analizowanych edytorów daje się również podzielić na trzy rozłączne klasy. Niezależnie jednak od różnorodności wykonywanych operacji, funkcje nawet bardzo rozbudowanych wersji edytorów można zawsze sprowadzić do tworzenia nowych zbiorów oraz usuwania i dopisywania nowych elementów na każdym z poziomów w już istniejących zbiorach.

Sekwencyjna forma zapisu informacji na taśmach magnetycznych sprawia, że realizacja przedstawionych powyżej funkcji może przebiegać tylko w jeden sposób. Polega on na selektywnym kopiowaniu wskazanych taśm i plików kart w celu stworzenia nowych zbiorów o żądanej strukturze. Jeżeli w trakcie tego procesu okaże się, że parametry sterujące edycją były błędnie podane, powinna być ona przerwana i ponownie uruchomiona.

Przedstawione powyżej fakty stanowiły punkt wyjścia do sformułowania założeń projektowych edytora TEED.

Został on zaprojektowany jako swoisty rodzaj języka do edycji zbiorów zapisanych na taśmach magnetycznych. Jego instrukcje pozwalają na warunkową zmianę kolejności wykonywanych czynności, umożliwiając tym samym reakcje na wszystkie możliwe do przewidzenia, a dynamicznie powstające sytuacje wyjątkowe. Pozwala to ominąć ograniczenia, jakie stwarza wsadowy tryb przetwarzania, i umożliwia współpracę ze zbiorami o nieznanej strukturze i zawartości.

Zakres realizowanych przez TEED funkcji został podzielony między kilka instrukcji, z których każda steruje wykonaniem jednej prostej czynności.

Instrukcje TEED pozwalają dokonywać operacji na zbiorze następujących obiektów:

- zbiór wejściowy: jest to taśma magnetyczna z zapisaną na niej informacją znakową; zbiór wejściowy może być traktowany jedynie jako źródło niezniszczalnej w trakcie edycji informacji

- plik kart

- zbiór wyjściowy: jest to taśma magnetyczna, na której, zgodnie z treścią programu sterującego, powstanie zbiór znakowy o żądanej strukturze i zawartości

- wskaźniki; są to obiekty przypisane do zbioru wejściowego, identyfikują one miejsce, do którego odnosić się będą kolejne instrukcje TEED; zgodnie z poziomą strukturą zbioru wejściowego rozróżniane są trzy wskaźniki: 1) główny, wskazujący na podzbiór w zbiorze wejściowym, 2) wierszowy, wskazujący na rekord, 3) znakowy, wskazujący na znak w rekordzie

- zmienna znakowa; jest to bufor umożliwiający konkatenację zapisywanych do niego rekordów i ich części.

Z przedstawionych tu obiektów jedynie trzy pierwsze są fizycznie dostępne dla użytkownika. Pozostałe realizuje wewnętrznie TEED i operowanie nimi jest możliwe jedynie poprzez jego instrukcje.

Omówione wyżej obiekty są podstawowymi elementami, do których odnoszą się instrukcje edytora TEED. Instrukcje te pozwalają na realizację następujących czynności:

- przesuw wskaźników i ustawienie ich na następujących pozycjach:

- głównego — według nazwy podzbioru lub jego miejsca w zbiorze
- wierszowego — według zawartości rekordu lub jego numeru
- znakowego — według kolejności znaków w rekordzie lub ich kontekstu

- kopiowanie według wskaźników:

- całych podzbiorów do zbioru wyjściowego
- rekordów lub znaków do zmiennej znakowej
- rekordów lub znaków do zbioru wyjściowego

- kopiowanie zawartości zmiennej znakowej do zbioru wyjściowego

- kopiowanie pliku kart do zbioru wyjściowego.

Przedstawiony wyżej spis obiektów i wykaz oferowanych przez TEED czynności scharakteryzowany został w sposób uproszczony. Obrazuje on jednak podstawową ideę, na której opiera się zasada działania tego edytora. Operując na niewielkiej liczbie obiektów kilkoma zaledwie prostymi czynnościami, pozwala on wykonywać praktycznie wszystkie, nawet najbardziej złożone operacje. Ich zakres rozszerza się jeszcze bardziej jeżeli uwzględnimy fakt, że TEED został wyposażony także w instrukcje strukturalne. Pozwalają one na warunkowe, zależne od aktualnego położenia dowolnego wskaźnika, lub wielokrotne wykonywanie wydzielonych fragmentów programu.



Mgr inż. Krzysztof BIESAGA ukończył Wydział Elektroniki Politechniki Warszawskiej (kierunek „informatyka”) w 1976 roku. Pracuje w Instytucie Informatyki, Technicznej i Ekonomicznej, w Zakładzie Systemów Informatycznych, na stanowisku starszego asystenta.

Omówiony w niniejszym artykule opis języka TEED nie jest pełny — bliższe dane zainteresowani mogą znaleźć w [1, 2]. Edytor ten bowiem zawiera jeszcze dodatkowe, nie uwzględnione powyżej możliwości. Należą do nich między innymi:

- korzystanie w trakcie jednego przebiegu z nieograniczonej liczby taśm wejściowych i plików kart
- równoległe korzystanie z zawartości dwóch zbiorów wejściowych
- tworzenie w trakcie jednego przebiegu dowolnej liczby zbiorów wyjściowych
- korzystanie z zapamiętanej wcześniej pozycji wskaźników.

Niestety za bogactwem oferowanych przez TEED możliwości kryją się również pewne niedostatki. Wynikają one głównie ze znacznej długości programów sterujących edycją. Wada ta jest jednak łagodzona przez bogatą diagnostykę błędów syntaktycznych i nie powinna przeszkodzić w szerokim stosowaniu edytora TEED.

Przykład programu napisanego w języku TEED

W celu lepszego zilustrowania możliwości edytora przedstawiono poniżej krótki, przykładowy program napisany w języku TEED. Jego zadaniem jest wprowadzenie poprawek do podzbioru PODZBIOR, zawartego w zbiorze ZBIOR DANYCH. Program ten uwzględnia ponadto następujące założenia:

- podzbiór prosty PODZBIOR może zawierać nie więcej niż 200 rekordów znakowych
- rekordy rozpoczynając się od znaku # należy wymienić na kolejne rekordy, zawarte w zbiorze prostym o nazwie POPRAWKI
- poprawiony zbiór i podzbiór powinny mieć numer generacji 2.

- ```
(1) $IN POPRAWKI
(2) $IN ZBIOR DANYCH, 1
(3) $OUT /ZBIOR DANYCH/ 2
(4) $COPY PODZBIOR, 1
(5) $OSF PODZBIOR, 2
(6) FOR & 200
(7) IF V
(8) ELSE
(9) IF P = . + ' # '
(10) $FACT
(11) WP.
(12) P + 1.
(13) $ACT
(14) ELSE
(15) W P.
(16) FI
(17) P + 1.
(18) FI
(19) LOOP
(20) $CSF
(21) $SKIP
(22) $COPY DOKONCA
(23) xxxx
```

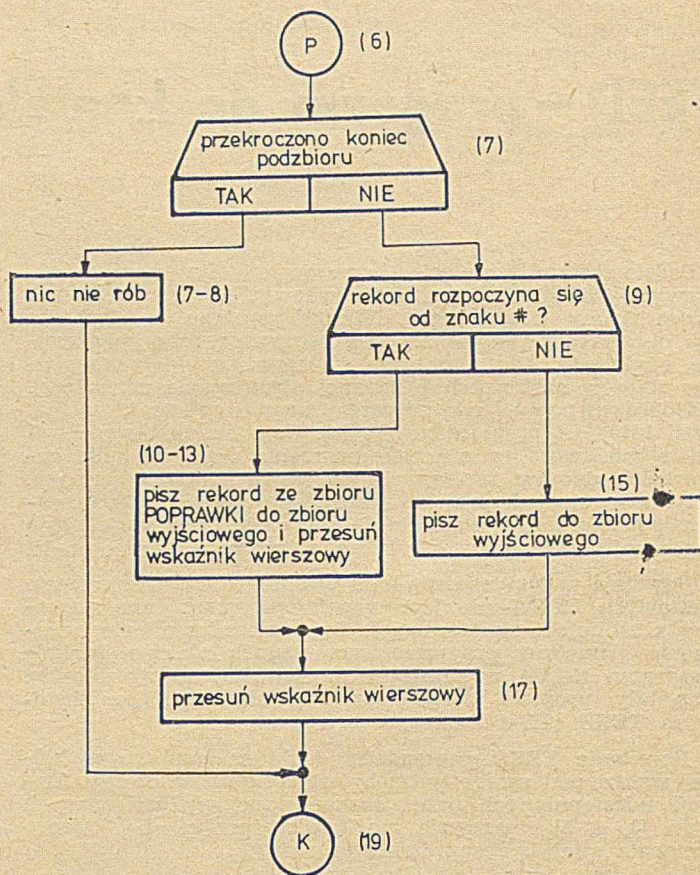
Instrukcje zawarte w wierszach 1—3 definiują zbiory POPRAWKI i ZBIOR DANYCH (numer generacji 1) jako wejściowe oraz powodują otwarcie na taśmie roboczej zbioru wyjściowego ZBIOR DANYCH (numer generacji 2).

Instrukcja \$COPY (wiersz 4) kopiuje na taśmę wyjściową zawartość podzbiorów ze ZBIOR DANYCH od jego początku aż do napotkania podzbioru PODZBIOR, a \$OSF (wiersz 5) zapisuje na taśmie wyjściowej oznacznik poprawianego podzbioru.

W kolejnych wierszach, zamknięte w nawiasy strukturalne FOR — LOOP (wiersze 6—19), znajdują się instrukcje wykonywane w pętli i realizujące zasadniczą część zadania.

Na koniec wreszcie, poprawiony podzbiór zostaje zamknięty (wiersz 20), a taśma wejściowa skopiowana aż do końca (wiersze 21, 22).

Sens czynności wykonywanych w zasadniczej części (wiersze 7—18) programu obrazuje jego schemat (rysunek). Liczby umieszczone obok symboli graficznych odpowiadają numerom wierszy, w których znajdują się instrukcje realizujące opisane czynności.



Program operuje równoległe na dwóch zbiorach wejściowych. Ponieważ instrukcje dotyczące wskaźników mają zawsze tę samą postać, aby rozróżnić zbiory, do których się one odnoszą, wprowadzono instrukcje \$ACT i \$FACT (bardziej precyzyjne wyjaśnienie w bibliografii — poz. 1, 2). Komentarze zawarte wewnątrz symboli schematu, jeżeli nie zaznaczono inaczej, odnoszą się do zbioru ZBIOR DANYCH.

#### LITERATURA:

- [1] Biesaga K.: Implementacja edytora kontekstowego dla maszyn cyfrowych ODRA serii 1300. Praca dyplomowa, Instytut Informatyki Politechniki Warszawskiej, Warszawa 1976
- [2] Biesaga K.: Edytor kontekstowy TEED. Instrukcja użytkownika. Instytut Informatyki Politechniki Warszawskiej, Warszawa 1978
- [3] Opis programu edytora GEORGE 3—4. ICL-ODRA seria 1300
- [4] Opisy edytorów systemu CYBER 70
- [5] PDP-10 Timesharing Handbook. PDP-10 Handbook Series. D.E.C. 1970
- [6] Programy organizacyjne dla pamięci taśmowej ODRA 1304. Dokumentacja techniczno-ruchowa, ELWRO 1971

# Systemy komputerowe ze strukturami nadmiarowymi

Pelniejsze i bardziej efektywne wykorzystanie systemów komputerowych wymaga zainteresowania się metodami poprawy niezawodności zarówno sprzętu komputerowego, jak i kompletnych systemów.

Poprawę niezawodności systemów komputerowych można uzyskać przez:

- stosowanie bardziej niezawodnych elementów
- odpowiednie zaprojektowanie struktury funkcjonalnej
- polepszenie technicznych warunków eksploatacji.

Strukturalne metody poprawy niezawodności to:

- uproszczenie konstrukcji obiektu
- redukcja katastroficznych następstw uszkodzeń przez stosowanie odpowiednich konfiguracji strukturalnych i funkcjonalnych
- rozbudowa układów kontroli
- rozbudowa układów diagnostyki z równoczesnym rezerwowaniem.

Z wyjątkiem niektórych typów rezerwowania, wszystkie pozostałe metody poprawy niezawodności wymagają zmian konstrukcyjnych obiektu i mogą być uwzględnione jedynie przy opracowywaniu nowych rozwiązań konstrukcyjnych.

## REZERWOWANIE W SYSTEMACH KOMPUTEROWYCH

Techniki redundancyjne stosowane w systemach komputerowych dzielą się na 4 zasadnicze typy:

- tworzenie rezerwy strukturalnej tj. wprowadzenie do systemu dodatkowych środków technicznych, zbędnych z funkcjonalnego punktu widzenia
- tworzenie rezerwy funkcjonalnej, tj. wykorzystanie zdolności pewnych obiektów do wykonywania dodatkowych zadań
- tworzenie rezerwy czasowej, tj. wielokrotne powtarzanie pewnych operacji, programów lub ich fragmentów w celu zwiększenia pewności działania (jest to możliwe przy istnieniu nadmiaru czasu obliczeniowego)
- tworzenie rezerwy informacyjnej, tj. stosowanie nadmiaru informacji podczas przetwarzania (np. kody nadmiarowe).

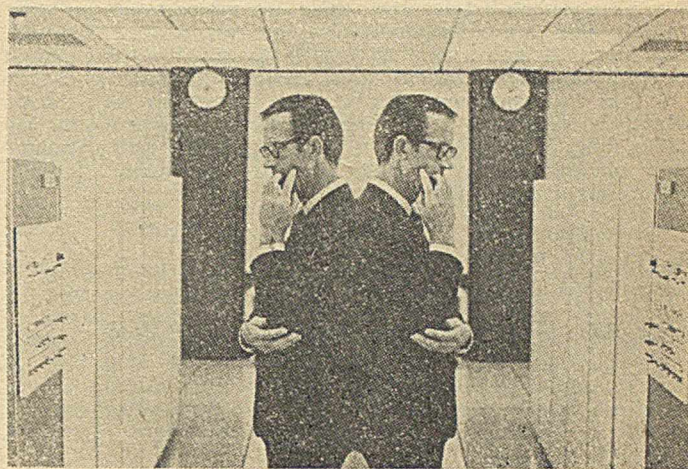
Rezerwa informacyjna jest stosowana w układach automatycznej korekty przekłamań i błędów trwałych oraz w układach kontroli i diagnostyki. Jest ona ściśle związana ze sposobem kodowania informacji i dlatego musi być uwzględniona już na etapie projektowania architektury logicznej systemu.

Rezerwa czasowa pozwala na bieżąco poprawiać błędy powstałe na skutek przekłamań. Z uwagi jednak na konieczność wykorzystania dodatkowego czasu obliczeniowego może być stosowana tylko w niektórych typach systemów komputerowych. Jest ona związana z systemami kontroli i diagnostyki oraz wymaga dodatkowych umiejętności od operatora systemu.

Rezerwa funkcjonalna jest praktycznie bardzo rzadko stosowana w systemach cyfrowych. Wynika to z faktu, że prawie cały współczesny sprzęt komputerowy to urządzenia bardzo wyspecjalizowane i zastępowanie ich innymi podobnego rodzaju, powoduje znaczne zmniejszenie efektywności.

## ANALIZA MOŻLIWOŚCI REZERWY STRUKTURALNEJ W SYSTEMACH KOMPUTEROWYCH

Rezerwa strukturalna wymaga stosowania dodatkowego sprzętu, który w sytuacjach awaryjnych zostaje włączony do pracy w miejsce urządzeń uszkodzonych oraz odpowiednich przełączników przeprowadzających bieżącą diagnostykę i sterujących działaniem całej grupy rezerwowej.



### Jednostka centralna

W większości systemów komputerowych JC stanowi tzw. jądro niezawodnościowe systemu, tzn. jej parametry niezawodnościowe decydują o niezawodności całego systemu. Z tego względu byłoby korzystne wprowadzenie rezerwy JC np. w postaci systemu dwumaszynowego. Istotną przeszkodą w tworzeniu takiej konfiguracji jest skomplikowany układ przełącznika, który powinien zapewniać odnowę informacyjną systemu w stanach awaryjnych, a przy tym posiadać bardzo wysoką niezawodność. Istniejące konfiguracje dwumaszynowe (dwie JC + adapter kanał-kanał + wspólna PAO) nie spełniają tych wymagań, zaś wykorzystywanie wspólnej pamięci operacyjnej przenosi ciężar utrzymania sprawności systemu (jądro niezawodnościowe) właśnie na PAO. Z tego względu istniejące konfiguracje dwumaszynowe należy raczej rozważać jako przykład rezerwy funkcjonalnej.

### Pamięci zewnętrzne

Możliwe są dwa poziomy rezerwowania pamięci zewnętrznych systemu cyfrowego: ogólny, odnoszący się do poszczególnych bloków pamięci oraz szczegółowy, dotyczący modułów pamięci (JSP), oraz szczegółowy, dotyczący modułów pamięci. Stosowanie rezerwy ogólnej wymaga odpowiedniego przełącznika przystosowanego do wymagań interfejsu systemu. Przełącznik taki powinien od strony JC symulować pojedynczą JSP, zaś od strony poszczególnych JSP powinien zapewnić ich diagnostykę i automatyczne przełączanie w sytuacjach awaryjnych. Rezerwowanie szczegółowe polega na zastosowaniu pewnej liczby dodatkowych modułów w każdym bloku pamięci, co umożliwia tworzenie struktur nadmiarowych typu  $k$  z  $n$ .

### Urządzenia WE-WY

Wyróżnia się tu trzy zasadnicze sposoby tworzenia rezerwy strukturalnej urządzeń zewnętrznych:

- zmiana kolejności pracy urządzeń w przypadku stwierdzenia awarii niekatastroficznej, tj. nie wymagającej wyłączenia systemu (wymaga interwencji operatora systemu)
- zastosowanie przełącznika prowadzącego ciągłą kontrolę stanu urządzeń i automatycznie wyłączającego urządzenia w stanach awaryjnych
- zamiana urządzeń uszkodzonych, wymagająca z reguły przerwania pracy systemu.

Spośród wymienionych metod najskuteczniejsza, lecz także najtrudniejsza do realizacji jest metoda druga, powszechnie natomiast stosuje się zamianę urządzeń uszkodzonych sprawnymi.

## OCENA NIEZAWODNOŚCI SYSTEMU CYFROWEGO Z UWZGLĘDNIENIEM REZERWY

Podjęcie decyzji o tworzeniu struktur rezerwowych dla poszczególnych urządzeń systemu komputerowego powinno być poprzedzone wnikliwą analizą niezawodności oraz oceną efektywności rezerwowania. Wówczas bardzo istotny jest właściwy wybór miary niezawodności systemu, odpowiedniej do jego przeznaczenia i wymagań eksploatacyjnych.

Przy ocenie niezawodności systemu zakłada się, że poszczególne grupy urządzeń pracują w konfiguracji szeregowej. Konsekwencją tego założenia jest możliwość potraktowania probabilistycznych miar niezawodności systemu (takich jak np. niezawodność przy wykonaniu zadania, współczynnik gotowości, itp.) jako iloczynu odpowiednich miar niezawodności poszczególnych grup urządzeń.

I tak dla systemów działających okresowo najbardziej właściwą ocenę daje tzw. niezawodność przy wykonaniu zadania (ang. *mission reliability*)  $R(T)$ , gdzie  $T$  jest wymaganym okresem bezawaryjnego działania systemu. Zgodnie z przyjętym założeniem:

$$R_S(T) = \prod_{i=1}^{i=n} R_i(T)$$

gdzie  $R_i(T)$  — niezawodność  $i$ -tej grupy urządzeń.

Natomiast w systemach czasu rzeczywistego oraz w niektórych systemach przetwarzania wsadowego najważniejszymi wskaźnikami niezawodności są tzw. współczynnik gotowości technicznej  $K_{er}$  oraz średni czas naprawy MRT.

W przypadku stałych intensywności uszkodzeń  $\lambda_i = \text{const}$  oraz intensywności naprawy  $\mu_i = \text{const}$ , zgodne z przyjętym poprzednio założeniem, otrzymuje się:

$$K_{erS} = \prod_{i=1}^{i=n} K_{er_i}$$

oraz:

$$MRT_S = \left( \sum_{i=1}^{i=n} \mu_i \right)^{-1}$$

Przytoczone zależności mogą stanowić podstawę oceny efektywności tworzenia rezerwy strukturalnej poszczególnych urządzeń systemu cyfrowego, jak również mogą być wykorzystane przy rozwiązywaniu zagadnień optymalizacji. Ponadto umożliwiają one wykrycie najsłabszych ogniw w strukturze niezawodnościowej, co jednak wymaga z reguły analizy obciążeń wykonawczych poszczególnych urządzeń.

JAN WALUSZEWSKI

Politechnika Świętokrzyska

## Czy oprogramowanie będzie prawnie chronione?

W dniach 27 i 28 marca br. odbyło się w Budapeszcie spotkanie międzynarodowego zespołu ekspertów<sup>1)</sup> na temat ochrony prawnej oprogramowania komputerów.

Celem spotkania było sformułowanie wspólnego stanowiska w sprawie ochrony oprogramowania w ustawodawstwach krajowych oraz zawarcie porozumienia międzynarodowego w tym zakresie. Rozważania koncentrowały się wokół ustawy wzorcowej o ochronie oprogramowania komputerowego z dnia 24 października 1977 r. (AGCP/NGO/IV. 8.Rev.), opracowanej przez Światową Organizację Własności Intelektualnej (WIPO)<sup>2)</sup>.

Ustawa ta została pozytywnie oceniona przez XXX Kongres AIPPI, obradujący w Monachium w 1978 roku. Kongres zalecił, żeby grupy narodowe AIPPI oddziaływały na odpowiednie władze w swych krajach w kierunku wprowadzenia ochrony oprogramowania, a także aby uwzględniono rozwiązania przyjęte w ustawie wzorcowej. Ponadto Kongres uznał za wskazane zawarcie międzynarodowego porozumienia w dziedzinie ochrony prawnej oprogramowania.

Biorąc pod uwagę powyższe zalecenia XXX Kongresu AIPPI, uczestnicy spotkania w Budapeszcie przeprowadzili analizę rozwiązań przyjętych w ustawie wzorcowej WIPO, oceniając ją przede wszystkim z punktu widzenia swoich krajów. Pozwoliło to na uzgodnienie wspólnego stanowiska, które przedstawię w ostatniej części artykułu.

Uczestnicząc w spotkaniu w Budapeszcie z ramienia polskiej grupy AIPPI, przedstawiłem nasze stanowisko w kwestii ochrony oprogramowania komputerów<sup>3)</sup>.

Generalnie można stwierdzić, że w interesie Polski nie leży obecnie wprowadzenie ochrony oprogramowania na zasadach określonych w ustawie wzorcowej WIPO. Przemawiają za tym argumenty dwojakiego rodzaju:

- 1) związane z sytuacją w dziedzinie produkcji i obrotu oprogramowaniem w Polsce
- 2) oparte na krytycznej ocenie poszczególnych rozwiązań przyjętych w ustawie wzorcowej.

Ad. 1. W literaturze tradycyjnie rozróżnia się cztery grupy podmiotów zainteresowanych problemem ochrony oprogramowania komputerów. Są to producenci maszyn cyfrowych oraz ich użytkownicy, którzy — aczkolwiek z różnych przyczyn — występują przeciwko ochronie oprogramowania, oraz producenci oprogramowania i autorzy programów, którzy opowiadają się za ich ochroną.

W Polsce nie występuje tak wyraźna polaryzacja stanowisk pomiędzy podmiotami zaangażowanymi w produkcję i obrót oprogramowaniem. Obrót sprzętem komputerowym jest ściśle reglamentowany, a zatem jego producenci nie odczuwają skutków wprowadzenia ochrony oprogramowania.

Również twórcy programów nie są w Polsce szczególnie zainteresowani ochroną oprogramowania w sposób proponowany w ustawie wzorcowej. Ponieważ oprogramowanie niemal w całości wykonywane jest w ramach stosunków

<sup>1)</sup> W skład zespołu wchodzi specjalista delegowany przez narodowe grupy Międzynarodowego Stowarzyszenia Ochrony Własności Przemysłowej (AIPPI) z krajów członkowskich RWPG

<sup>2)</sup> WIPO przyjęło trzecią wersję ustawy wzorcowej, poprzednią — drugą — wersję omówiłem bliżej w artykule „Zagadnienie ochrony prawnej” programów dla maszyn cyfrowych na forum międzynarodowym. INFORMATYKA nr 11/1977

<sup>3)</sup> Opierałem się przy tym na opinii przygotowanej uprzednio przeze mnie na zlecenie polskiej grupy AIPPI



pracy, tylko wyjątkowo prawa na oprogramowaniu powstałyby na rzecz twórców. Zgodnie bowiem z art. 2 ustawy wzorcowej, prawa te w powyższej sytuacji powstają pierwotnie na rzecz pracodawcy. W przypadku wprowadzenia ochrony opartej na ustawie wzorcowej, sytuacja twórców nie uległaby zmianie. Także ochrona ich interesów osobistych nie zostałaby wzmocniona, gdyż i tak można by ją opierać w Polsce jedynie na przepisach kodeksu cywilnego<sup>4)</sup>.

Jak wynika z powyższych wywodów wprowadzenie ochrony oprogramowania w formie zalecanej przez WIPO odczuli by jedynie producenci oprogramowania oraz jego użytkownicy. Czy jednak skutki wprowadzenia takiej ochrony byłyby rzeczywiście odczuwalne, a jeśli tak, czy prowadziłyby do korzystnych zmian w dziedzinie produkcji i obrotu oprogramowaniem w Polsce?

Sądzę, iż można udzielić negatywnej odpowiedzi na postawione wyżej pytania. Na tle obowiązujących obecnie zasad wytwarzania i rozpowszechniania oprogramowania ochrona prawna oprogramowania nie wywołałaby pożądanych zmian.

W Polsce przeważająca część oprogramowania wytwarzana jest w ramach Zjednoczenia Informatyki, gdzie obowiązuje obecnie zarządzenie nr 12 Naczelnego Dyrektora ZI z dnia 19 października 1978 r. w sprawie zasad, organizacji i trybu podejmowania prac projektowo-programowych, realizacji dostaw oraz koordynacji w tym zakresie.

W przepisach tego zarządzenia wyróżniono dwa rodzaje prac w dziedzinie programowania: prace badawczo-rozwojowe i prace nad zastosowaniami informatyki w gospodarce narodowej. W zależności od charakteru pracy różny jest tryb ich prowadzenia, rozpowszechniania wyników oraz rozliczenia finansowego. Jednakże w obu przypadkach generalnym celem wynikającym z wymienionego zarządzenia jest zapobieganie powtarzaniu się prac nad tymi samymi zagadnieniami, efektywno wykorzystanie i rozpowszechnianie już uzyskanych rozwiązań oraz koncentracja wysiłków na najistotniejszych problemach. Wiąże się z tym obowiązek zgłaszania podejmowanych prac nad oprogramowaniem oraz jego sformalizowanie ewidencjonalnie. Także rozpowszechnianie gotowego oprogramowania, odbywające się w ramach ZI lub dla użytkowników z zewnątrz, jest szczegółowo uregulowane.

Tak więc sytuacja w omawianej dziedzinie w Polsce determinowana jest warunkami ekonomicznymi, tj. niewielką ilością sprzętu, ograniczoną liczbą producentów oprogramowania oraz koniecznością optymalnego wykorzystania tych zasobów. W związku z tym produkcja i obrót oprogramowaniem są w Polsce ściśle reglamentowane i obecnie trudno sobie wyobrazić naruszenie przepisów o ochronie oprogramowania, gdyby miały one postać zaproponowaną przez WIPO.

Pozycja Polski na międzynarodowym rynku oprogramowania także nie uzasadnia wprowadzenia ochrony programów komputerowych. Jesteśmy przede wszystkim importem oprogramowania, musimy zatem liczyć się z tym, że chroniąc oprogramowanie własne, będziemy zobowiązani — na zasadzie wzajemności lub porozumień międzynarodowych — chronić również programy sprowadzane z zagranicy.

Ad. 2. Spośród stosunkowo licznych zastrzeżeń, jakie można zgłaszać w odniesieniu do poszczególnych rozwiązań przyjętych w ustawie wzorcowej, na plan pierwszy wysuwają się uwagi dotyczące określenia przedmiotów podlegających ochronie (art. 1, 3 i 6 ust. 2 ustawy wzorcowej).

Art. 1 ustawy wzorcowej określa przedmioty intelektualne objęte wspólną nazwą „oprogramowanie komputerowe” (*computer software*). Są to „programy komputerowe” (*computer programs*), „opisy programów” (*program descriptions*) i „materiały pomocnicze” (*supporting materials*).

Zgodnie z art. 3 ustawy, aby oprogramowanie podlegało ochronie wynikającej z ustawy, powinno być oryginalne, tj. musi stanowić wynik własnego wysiłku intelektualnego twórcy. Z przepisu art. 6 ust. 2 ustawy wynika, zaś, że czyny dotyczące identycznego lub istotnie podobnego oprogramowania, które zostało niezależnie stworzone przez inną osobę, nie stanowią naruszenia praw na oprogramowaniu przysługujących uprawnionemu.

Ograniczone ramy artykułu uniemożliwiają przeprowadzenie szerszej, teoretycznej krytyki wyżej opisanej konstrukcji prawnej. Można jednak już *prima facie* dostrzec, jak olbrzymie praktyczne trudności pociąga ona za sobą. Bardzo skomplikowane jest np. zagadnienie identyfikacji programu oraz stwierdzenia istnienia lub braku jego tożsamości z programem opracowanym przez drugą osobę. Jeszcze większe trudności dowodowe wynikną przy ustalaniu czy program był wykonany niezależnie od identycznego lub istotnie podobnego programu, w stosunku do którego uprawniona jest inna osoba.

Można więc przyjąć, że rozwiązania przyjęte w ustawie wzorcowej preferują potentatów w zakresie produkcji oprogramowania. Tylko oni bowiem mają odpowiednie środki na prowadzenie długotrwałych i kosztownych procesów o naruszenie praw na oprogramowaniu.

Następstwem wprowadzenia ochrony oprogramowania w poszczególnych krajach, a także w stosunkach międzynarodowych, byłaby dalsza monopolizacja produkcji oprogramowania. Słabi producenci obawiliby się dostarczać swoje wyroby na rynek, gdyż wiązałoby to się z ryzykiem procesu, którego wynik mógłby być dla nich niekorzystny.

Innym przepisem ustawy wzorcowej, który także wydaje się kontrowersyjny, jest art. 7 ustawy, przewidujący 20-letni okres trwania praw na oprogramowaniu. Okres ten liczony jest od chwili pierwszego zastosowania programu na maszynie cyfrowej, bądź od chwili, gdy oprogramowanie po raz pierwszy zostało wprowadzone do obrotu. Prawa na oprogramowanie nie mogą jednak trwać dłużej niż 25 lat od chwili jego stworzenia.

Okresy przyjęte w ustawie wzorcowej należy uznać za zbyt długie. Szybki rozwój metod i technik programowania sprawia, że kilkunastoletnie, a niekiedy kilkuletnie oprogramowanie uważane jest za przestarzałe. Szybki jest również rozwój nowych konstrukcji technicznych w dziedzinie budowy komputerów, z czym wiąże się zjawisko moralnego starzenia się sprzętu, a co za tym idzie także oprogramowania komputerowego.

Omawiając tę ostatnią kwestię na spotkaniu w Budapeszcie, zwróciłem uwagę na zmiany w zakresie programowania, jakie powoduje szerokie stosowanie układów scalonych o dużej skali integracji. Rozwijające się oprogramowanie sprzętowe samo może sprawi, że problem ochrony prawnej oprogramowania tradycyjnego umrze śmiercią naturalną. W stosunku do zintegrowanych układów scalonych będzie bowiem można stosować formy ochrony zbliżone do ochrony patentowej.

Stanowiska reprezentowane przez pozostałych członków zespołu ekspertów, oparte na zbliżonej argumentacji, były podobne do przedstawionego w imieniu polskiej grupy AIPPI. Znalazło to wyraz w końcowym dokumencie opracowanym na budapesztańskim spotkaniu.

Biorąc pod uwagę dotychczasowe rezultaty działalności międzynarodowego zespołu ekspertów oraz sytuację w poszczególnych krajach członkowskich RWPG, stwierdzono, że możliwość wprowadzenia w poszczególnych krajach ochrony oprogramowania związana jest ściśle z warunkami, jakie istnieją w tych krajach w dziedzinie produkcji i obrotu oprogramowaniem.

Opracowana przez WIPO ustawa wzorcowa stanowi kompromis godzący przeciwstawne interesy w zakresie ochrony oprogramowania. W ustawie wzorcowej uwzględniono co prawda niektóre krytyczne uwagi zgłaszane uprzednio przez kraje RWPG w odniesieniu do wcześniejszych projektów, ale generalnie rozwiązania przyjęte w ustawie wzorcowej nie mogą stanowić podstawy regulacji ochrony oprogramowania w krajach RWPG, mimo iż niektóre z tych rozwiązań należy uznać za nader interesujące.

W przedmoście międzynarodowego porozumienia o ochronie oprogramowania komputerów zespół ekspertów stwierdził w dokumencie końcowym, że propozycja zawarcia takiego porozumienia jest jeszcze przedwczesna. Nie uregulowano jeszcze tych zagadnień w ustawodawstwach krajowych, nie wytworzyła się jeszcze praktyka prawna w tym zakresie i nie ma doświadczeń, które mogłyby stanowić podstawę prac na forum międzynarodowym.

<sup>4)</sup> Zob.: Waluszewki J.: Prawa twórców programów, *INFORMATYKA* nr 7—8/1978

Niewiele wiemy o hiszpańskiej informatyce i niewiele jest w Polsce materiałów na ten temat. Skoro więc trafiła do redakcji obszerna (332 strony) publikacja „La informatica en Espana 1976”, opracowana przez potężny zespół specjalistów na użytek władz państwowych, nie bacząc na niezbyt aktualne dane (odnoszące się głównie do lat 1975 i 1976), co ciekawsze fragmenty wydawnictwa przekazujemy poniżej naszym Czytelnikom.

# Informatyka w Hiszpanii

## SYTUACJA ORGANIZACYJNO PRAWNA

Organizacja informatyki w sektorze państwowym przedstawia się następująco (dane z 1975 r.). Przy Urzędzie Prezydium Rady Ministrów w ramach Generalnego Sekretariatu ds. Techniki działa Centralna Dyrekcja Informatyki, przy której z kolei pracuje Międzyresortowa Komisja Informatyki. W poszczególnych ministerstwach (z wyjątkiem Ministerstwa Sprawiedliwości) działają komisje ds. informatyki, nadzorujące wszystkie jednostki resortu. W niektórych resortach istnieją centralne ośrodki informatyczne, w innych działalność informatyczna jest zdecentralizowana. Zakupy sprzętu informatycznego na potrzeby sektora państwowego podlegają Komisji Międzyministerialnej, której kompetencje rozciągają się na takie dziedziny, jak: kontrola kosztów, kontrola stopnia wykorzystywania sprzętu oraz systemów zastosowań, sprawy normalizacji itp. Powstanie każdego ośrodka informatycznego lub wydzielonej służby informatycznej wymaga wydania aktu rządowego odpowiedniego szczebla. Centralizacja ta nie zmniejsza jednak obowiązku poszczególnych służb informatycznych do przestrzegania trzech podstawowych zasad działania: przejrzystości, niezawisłości i rentowności.

W sektorze prywatnym działają mechanizmy ekonomiczne przewidziane w narodowych planach rozwoju.

Zagadnienia prawne dotyczące ochrony praw pracowników ośrodków informatycznych, praw autorskich, własności intelektualnej, ochrony danych itp. są już w Hiszpanii częściowo uregulowane.

## PRZEMYSŁ KOMPUTEROWY I PARK MASZYNOWY

Rynek komputerowy w Hiszpanii jest prawie całkowicie zdominowany przez firmy amerykańskie (73,1% ogólnej wartości zainstalowanego sprzętu).

Tabela 1. Park komputerów w Hiszpanii wg producentów (1975 r.)

| Producent      | Liczba zainstalowanych maszyn | Procent      |
|----------------|-------------------------------|--------------|
| IBM            | 1634                          | 67,0         |
| HONEYWELL-BULL | 239                           | 11,0         |
| UNIVAC         | 233                           | 9,5          |
| NCR            | 205                           | 8,4          |
| inni           | 100                           | 4,1          |
| <b>Łącznie</b> | <b>2450</b>                   | <b>100,0</b> |

Podział komputerów na poszczególne grupy wielkości charakteryzuje tabela 2.

Jeśli chodzi o liczbę zainstalowanych minikomputerów, to wynosiła ona w 1975 r. — wg jednej oceny — 6100, co stanowiło wartość 10,8 mld peset, natomiast wg innej oceny (łącznie z maszynami do księgowania) — 9299, wartości 13 mld peset. Jest to liczba znaczna, zwłaszcza w odniesieniu do komputerów uniwersalnych 2450 szt., wartości 612 mld peset).

Tabela 2. Park komputerowy w Hiszpanii wg podstawowych grup wielkości maszyn (1975 r.)

| Grupa komputerów | Liczba zainstalowanych maszyn | Procent      |
|------------------|-------------------------------|--------------|
| — b. duże        | 161                           | 6,6          |
| — duże           | 240                           | 9,8          |
| — średnie        | 1106                          | 45,1         |
| — małe           | 943                           | 38,3         |
| <b>Łącznie</b>   | <b>2450</b>                   | <b>100,0</b> |

Na czele producentów eksploatowanych minikomputerów figuruje firma Philips (30,0% w ujęciu wartościowym), a następnie NCR (24%). Dalsze miejsca zajmują: NIXDORF, IBM, TELEINCRO, LOGABAX, KIENZLE, OLIVETTI itp. Większość minikomputerów zainstalowanych jest w Barcelonie i Katalonii (48%) oraz w Madrycie (30%).

W grupie minikomputerów pracujących w czasie rzeczywistym dominuje firma IBM (33%) oraz HEWLETT-PACKARD (23%). Ogółem takich minikomputerów było w 1975 r. — 340, wartości ok. 1,4 mld peset.

Narodowy przemysł komputerowy w Hiszpanii nie istnieje. Jedynie kilka firm prywatnych zajmuje się produkcją minikomputerów, maszyn do księgowania i niektórych rodzajów urządzeń peryferyjnych. Firmy te są albo oddziałami firm zagranicznych albo spółkami z udziałem kapitału zagranicznego, produkującymi na licencjach zagranicznych.

Dane dotyczące produkcji, importu i eksportu sprzętu informatycznego w roku 1975 oraz prognozy na rok 1979 zawiera tabela 3.

Tabela 3. Produkcja import i eksport sprzętu komputerowego w Hiszpanii

| Rodzaj i działalność | Wartość w mln peset (1\$ — 73 pesety) |         |
|----------------------|---------------------------------------|---------|
|                      | 1975 r.                               | 1979 r. |
| Produkcja            | 1 398                                 | 10 626  |
| Import               | 10 277                                | 15 698  |
| Eksport              | 585                                   | 2 700   |

O ile pokrycie importu własnym eksportem wynosiło w roku 1975 tylko 5,7%, to w planie na rok bieżący zakłada się pokrycie do 18,1%. W okresie od 1976 do 1979 produkcja ma wzrosnąć o 55,6%, a import — tylko o 11,1%, eksport zaś — o 44,0%. Dane te wskazują wyraźnie na tendencję do tworzenia w Hiszpanii zrębów własnego niezależnego przemysłu informatycznego.

W 1976 roku polski sprzęt informatyczny — wartości zaledwie 15 tys. peset występował w statystyce celnej w imporcie hiszpańskim. Natomiast dostawy NRD były znaczne, wartości 22,5 mln peset. Omawiana statystyka nie podaje obrotów z innymi krajami RWP.

Tabela 4. Zatrudnienie personelu informatycznego w Hiszpanii wg badań ankietowych

|                                 | Grupy zawodowe |           |             |            |      | Ogółem | Przygotowanie danych | Razem |
|---------------------------------|----------------|-----------|-------------|------------|------|--------|----------------------|-------|
|                                 | Kierownicy     | Analitycy | Programiści | Operatorzy | Inni |        |                      |       |
| Ankieta Planu Rozwoju (1974 r.) | 1464           | 2745      | 5856        | 31542      | 1281 | 42888  | 54900                | 97788 |
| Ankieta ENTEL (1975 r.)         | 3535           | 5301      | 9619        | 18408      | 4978 | 41841  | 17694                | 59535 |

## ZATRUDNIENIE, SZKOLENIE ORAZ BADANIA NAUKOWE

Istnieją dosyć szczegółowe dane dotyczące stanu zatrudnienia informatyków, kilkakrotnie zbierane systemem ankietowym na przestrzeni lat 1972—1976. Dla porównania zostaną scharakteryzowane dwie ankiety, opracowane przez grupę roboczą IV Planu Rozwoju i przez Instytut ENTEL. Pierwszą opracowano w 1974 r. drugą — rok później. Stąd znaczne rozbieżności — zarówno w ocenie ogólnej liczby pracowników, jak i w podziale ich na poszczególne kategorie.

Zatrudnienie w hiszpańskiej informatyce wg dwóch wspomnianych ankiet przedstawia tabela 4.

Podobnie jak w innych krajach występuje w Hiszpanii zjawisko dużej koncentracji personelu informatycznego w stolicy oraz w głównych ośrodkach przemysłowych (Barcelona i okręg Vizcaya). Koncentrację tę ilustruje tabela 5.

Tabela 5. Koncentracja personelu i sprzętu informatycznego w Hiszpanii

| Miasto lub okręg | Udział w krajowym zatrudnieniu (procent) | Udział w krajowym parku komputerów (procent) |
|------------------|------------------------------------------|----------------------------------------------|
| Madryt           | 43,2                                     | 33,9                                         |
| Barcelona        | 24,2                                     | 24,3                                         |
| Vizcaya          | 7,1                                      | 6,2                                          |

Tak np. w Madrycie pracuje aż 47% wszystkich kierowników, 49% analityków systemów i 48% programistów. Nic dziwnego, znajduje się tu Instytut Informatyczny i wiele dużych ośrodków obliczeniowych, większość producentów oprogramowania również zgromadziła się w stolicy. Zatrudnienie jest więc większe niż wynikałoby to z liczby komputerów zainstalowanych w Madrycie.

Według omawianego opracowania płace w informatyce nie należą bynajmniej do wysokich w skali całej gospodarki narodowej. Dla przykładu: dobry programista zarabia mniej więcej tyle ile dobry kreślarz, operatorka urządzeń przygotowania danych — tyle co dobra maszynistka, a kierownik ośrodka obliczeniowego często mniej niż radca prawny dużej organizacji przemysłowej. Wystarczy powiedzieć, że różnice w płacach między programistą a analitykiem nie przekraczają zwykle stosunku 1:3, natomiast pomiędzy średnim programistą a kierownikiem ośrodka — stosunku 1:2. Prognozy rozwoju parku komputerów oraz zatrudnienia w roku bieżącym ilustruje tabela 6.

Tabela 6. Prognozy rozwoju parku komputerów i zatrudnienia w Hiszpanii na 1979 r.

| Rodzaj hipotezy | Liczba komputerów | Zatrudnienie (osób) <sup>1</sup> |
|-----------------|-------------------|----------------------------------|
| pesymistyczna   | 3 517             | 85 390                           |
| średnia         | 4 248             | 103 191                          |
| optymistyczna   | 4 980             | 120 913                          |

<sup>1</sup> Łącznie z personelem w działach przygotowania danych, ocenianym na około 40% wszystkich zatrudnionych

Jeśli nawet przyjąć za podstawę dalszych rozważań tylko hipotezę średnią, to w stosunku do zatrudnienia w roku 1975 (tab. 4) zapotrzebowanie na personel informatyczny przez 4 lata w poszczególnych kategoriach pracowniczych wyniosłoby:

|                                        |             |
|----------------------------------------|-------------|
| kierownicy                             | 2633 osoby  |
| analitycy                              | 3874 osoby  |
| programiści                            | 7033 osoby  |
| operatorzy                             | 13494 osoby |
| pracownicy działu przygotowania danych | 21600 osób. |

Tak więc przewidywany wzrost zatrudnienia miał wynieść ok. 48 tys. osób, co oznacza podwojenie liczby pracowników. Realizacja tego zapotrzebowania wymaga wielkiego wysiłku ze strony organizatorów i koordynatorów informatyki hiszpańskiej, głównie w dziedzinie szkolenia. Toteż odpowiednio wcześniej podjęto już decyzje dotyczące szkolenia informatycznego na różnych poziomach i w różnych formach. Utworzono Instytut Informatyczny w Madrycie oraz wydziały informatyczne na uniwersytetach w Barcelonie i San Sebastian. Zorganizowano setki kursów zawodowych, które w okresie 1965—1975 wyszkoliły:

15971 osób w zakresie perforowania kart  
63723 osoby w zakresie programowania  
7832 osoby w zakresie analizy i projektowania systemów informatycznych.

Na kursach organizowanych w ośrodkach akademickich w okresie od 1971 do 1975 roku wyszkolono 9234 programistów, 3051 analityków i 568 techników obsługi systemów informatycznych. Dane te nie obejmują studentów kształcących się na kierunkach informatycznych wydziałów automatyki wyższych uczelni technicznych.

Obecnie, w oparciu o doświadczenia zagraniczne oraz własne prace badawcze i studialne, przygotowuje się plan szkolenia z horyzontem czasowym do roku 1985. Główny nacisk położony jest na zachowanie odpowiednich proporcji w szkoleniu różnych kategorii specjalistów, biorąc pod uwagę istniejącą tendencję do zmniejszania się średniego zatrudnienia na 1 komputer.

Istnieje też znaczny dorobek legislacyjny. Wszystkie większe firmy mają zatwierdzone przez Naczelną Dyрекcję Pracy umowy zbiorowe z pracownikami, bardzo dokładnie precyzujące obowiązki pracowników na poszczególnych stanowiskach (z wyjątkiem osób piastujących stanowiska kierownicze), a także wymagane kwalifikacje zawodowe oraz tryb awansowania.

W sektorze publicznym zatrudnienie personelu informatycznego jest stosunkowo małe. Ogółem w sektorze tym pracowało w 1973 roku 1790 osób, natomiast do roku 1980 przewiduje się wzrost zatrudnienia do ok. 2095 osób. Większość tych osób zatrudnionych jest w trzech resortach: szkolnictwa i nauki, spraw wewnętrznych oraz planowania i rozwoju.

Informacje na temat badań zostały zebrane drogą ankietyzacji wszystkich ośrodków informatycznych w kraju. Ankieta zawierała pytania dotyczące badań zarówno w zakresie sprzętu i oprogramowania, jak i zastosowań.

Z odpowiedzi wynika, że w temacie „sprzęt informatyczny” większość pracochłonności (60%) oraz kosztów (81%) poświęcana jest badaniom dotyczącym urządzeń peryferyjnych, technologii przetwarzania danych oraz sprzętu specjalnego. Wynika to najpewniej z tego, że — jak już wspomniano — w Hiszpanii właściwie nie istnieje przemysł komputerowy. W dziale badań nad oprogramowaniem większość pracochłonności i kosztów (65% i 85%) poświęcona jest oprogramowaniu teleinformatyki.

Podjęto studia w zakresie wypracowania własnego modelu jednostki badawczej oraz tematyki badań, a także ustalenia wskaźników kosztów i wydajności 1 pracownika.

## PRODUKCJA OPROGRAMOWANIA ORAZ USŁUGI INFORMATYCZNE

W 1975 r. istniało w Hiszpanii 105 (wraz z oddziałami — 135) firm zajmujących się produkcją oprogramowania oraz usługami informatycznymi. W większości są to firmy o charakterze lokalnym, a jedynie 14,1% tych firm działa na terenie całego kraju. Podobnie jak w przypadku personelu informatycznego znajdują się one głównie w wielkich miastach (w Barcelonie — 42, w Madrycie — 34 firmy). Zatrudniają łącznie 3978 osób (w tym 3157 pracowników etatowych). Znaczna część personelu (ca 30%) to analitycy systemów i programiści. Jedynie 40% firm ma charakter niezależnych przedsiębiorstw, pozostałe należą do użytkowników systemów informatycznych (np. banków). Większość z nich to niewielkie przedsiębiorstwa o stosunkowo małym kapitale akcyjnym (80% firm ma kapitał do 10 mln peset). Niewielkie są również obroty tych organizacji (ok. 75% firm wykazuje obroty roczne do 30 mln peset).

W latach 1972—1975 nastąpił znaczny rozwój tych firm. Dla przykładu łączne ich obroty w roku 1975 wyniosły 1165 mln peset, co stanowi wzrost w porównaniu do roku poprzedniego o 63%. Dane te dotyczą jedynie usług tzw. intelektualnych. Wartość usług wykonywanych za pomocą komputerów wyniosła w tym samym roku 2,2 mld peset i była większa niż w roku poprzednim o 20%.

W roku bieżącym ogólny koszt własny informatyki ma wynieść wg planu ok. 0,91% dochodu narodowego brutto, natomiast wartość usług informatycznych ma przekroczyć 1,2% tego dochodu.

Do rozwoju usług informatycznych przywiązuje się w Hiszpanii dużą wagę. Istnieje nawet zamiar stworzenia organizacji państwowej, która miałaby za zadanie promocję tej działalności i przygotowanie odpowiednich bodźców ekonomicznych.

## TELEINFORMATYKA

Wyjątkowo szerokie omówienie tego zagadnienia we wspomnianej publikacji świadczy o dużej wadze, jaką w Hiszpanii przywiązuje się do rozwoju teleinformatyki. Wynika to najpewniej z przyjętego modelu rozwoju informatyki w kraju, który wprawdzie nie rozwinął własnego przemysłu narodowego komputerowego, lecz produkuje urządzenia peryferyjne, a przede wszystkim terminale. Podobnie ma się sprawa z produkcją oprogramowania — główny nacisk położono na obsługę systemów teleinformatycznych.

Hiszpania ma już stosunkowo rozległą sieć teleinformatyczną, opartą na 6 ośrodkach regionalnych (Madryt, Barcelona, Valencja, Leon, Bilbao, Sevilla). Opracowano szczegółowe zamierzenia rozwoju teleinformatyki aż do roku 1985, w którym — w porównaniu do roku 1972 — nastąpi prawie trzydziestokrotny wzrost liczby informacji przekazywanych w ciągu 1 dnia, zainstaluje się ok. 41,3 tys. terminali zdolnych przekazać ok. 94 miliardów bitów informacji dziennie. Na rok bieżący plan zakładał osiągnięcie stanu 18237 terminali o łącznej zdolności przekazywania ok. 33 mld bitów informacji (w tym: ośrodki usług informatycznych — ok. 14,6 mld bitów, banki — 5,4 mld bitów, transport lotniczy — 3 mld bitów, administracja publiczna — 1,3 mld bitów).

Aktualnie w Hiszpanii eksploatowane są trzy odrębne sieci teleinformatyczne: Red Especial, Red Automatica Telefonica i Red Punto a Punto. Najszybsze urządzenia ma ostatnia sieć — w roku bieżącym zainstaluje się tu ok. 2340 terminali (większość z nich pracuje z szybkością od 1200 do 2400 bodów, a tylko 274 z szybkością ponad 2400 bodów). Pozostałe sieci mają większość urządzeń o szybkości w granicach 50—200 bodów, a tylko niektóre w przedziale 600—1200 bodów.

W roku bieżącym w samych tylko bankach i instytucjach finansowych mają być zainstalowane 6673 terminale, natomiast w usługowych ośrodkach przetwarzania danych — 5063 terminale.

Duży nacisk położono na rozwój narodowego przemysłu telekomunikacyjnego, unifikację sprzętu, opracowanie przepisów oraz badania dotyczące wykorzystania i funkcjonowania sieci teleinformatycznych, szkolenie personelu pozostawo, nadanie priorytetów finansowych w dziedzinie teleinformatyki, uruchomienie bodźców ekonomicznych umożliwiających szybki jej rozwój. Podejmowane są również działania zmierzające do podjęcia eksportu sprzętu teleinformatycznego i związanych z tym usług do Ameryki Południowej i krajów arabskich.

## UŻYTKOWNICY INFORMATYKI

Jeśli przyjmiemy, że zrealizowane zostały (wg hipotezy średniej) założenia Narodowego Planu Rozwoju na lata 1976—1979, w roku bieżącym powinno być zainstalowanych w Hiszpanii (z wyłączeniem minikomputerów) około 4240 komputerów, o wartości ok. 97,7 mld peset. Średni roczny przyrost parku w okresie tych czterech lat winien więc wynieść w ujęciu ilościowym 16%, a wartościowym — 12%. Wynika z tego, że wskaźnik liczby osób czynnych zawodowo przypadających na 1 komputer powinien wynieść w br. około 3,4 tys. osób (w 1964 roku wynosił on 89 tys. osób, a w roku 1970 — 121 tys. osób). Takie założenia świadczą wymownie o bardzo dużych ambicjach, a początki hiszpańskiej informatyki były więcej niż skromne (w 1964 roku — 130 maszyn).

W usługach informatycznych struktura kosztów przedstawia się następująco (dane z 1975 r.):

|                      |       |
|----------------------|-------|
| — koszty osobowe     | 44,4% |
| — koszty sprzętu     | 42,9% |
| — zakup usług obcych | 2,8%  |
| — inne koszty        | 9,9%  |

Używane języki programowania:

|           |       |
|-----------|-------|
| COBOL     | 28,6% |
| FORTRAN   | 3,5%  |
| ASSEMBLER | 67,9% |

Nasylenie sprzętem komputerowym podstawowych grup użytkowników ilustruje tabela 7:

Tabela 7. Rozmieszczenie komputerów wg podstawowych dziedzin gospodarki narodowej Hiszpanii

| Dział gospodarki narodowej                                                                        | Udział w krajowym parku komputerowym |                     |
|---------------------------------------------------------------------------------------------------|--------------------------------------|---------------------|
|                                                                                                   | Ilościowy (proc.)                    | wartościowy (proc.) |
| Rolnictwo, leśnictwo, rybołówstwo                                                                 | 1,1                                  | 0,4                 |
| Przemysł, budownictwo, energetyka                                                                 | 46,0                                 | 64,0                |
| Finanse, handel, administracja publiczna, transport, komunikacja, usługi i przemysł informatyczny | 52,9                                 | 35,6                |

Podjęte zostały również prace zmierzające do organizacyjnego i prawnego uporządkowania problemu banków danych, z których korzysta zarówno sektor państwowy, jak i sektor prywatny. Do roku 1985 zostaną wytyczone w tym zakresie główne kierunki działania.

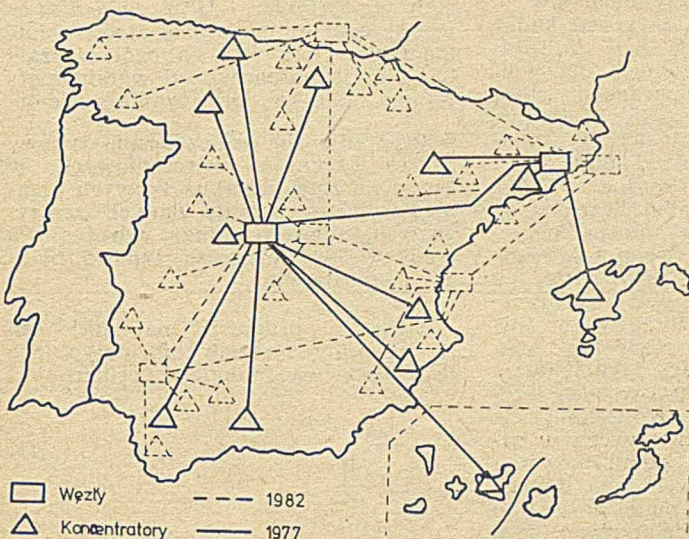
Opracował Witold TRYUK

# RETD – ogólnodostępna sieć transmisji danych

Rozwój informatyki w Hiszpanii może stanowić materiał do interesujących porównań: jest to kraj zbliżony pod względem wielkości zaludnienia i rocznego dochodu narodowego do Polski, niewiele większy od niej terytorialnie, a także — kraj, który ze znacznym opóźnieniem w stosunku do innych krajów zachodnioeuropejskich wystartował do naukowo-technicznej rekonstrukcji swej gospodarki. Dodajmy, iż Hiszpania nie rozwinęła na większą skalę swego narodowego przemysłu komputerowego, niemniej informatyka odgrywa tam coraz większą rolę. Jednym z przykładów tej aktywności jest Red Especial de Transmisión de Datos — ogólnodostępna sieć transmisji danych, wydzielona fizycznie z krajowej sieci telekomunikacji, zarządzanej przez Compañia Telefónica Nacional de España.

Prace nad budową RETD zostały podjęte w 1971 r. Jest to pierwsza na naszym kontynencie publiczna sieć transmisji danych, działająca na zasadach handlowych — co z dumą podkreślają jej twórcy. Z siecią współpracują komputery firm IBM, NCR, CII, UNIVAC, SIEMENS, HONEYWELL i in. oraz ponad 5 tys. punktów abonenckich, przy czym ta ostatnia liczba wzrastać będzie w przybliżeniu o 50% rocznie. Konstruktorzy zapewniają, że przyłączenie nowego punktu abonenckiego zajmie nie więcej niż 75 godzin, łącznie z przeprowadzeniem niezbędnych testów technicznych.

Obecną oraz przewidywaną w najbliższych latach strukturę geograficzną sieci przedstawia poniższy rysunek.



Z organizacyjno-technicznego punktu widzenia w RETD wyodrębnia się następujące szczeble połączeń:

- I — połączenia między węzłami komunikacyjnymi oraz między ośrodkami przetwarzania danych użytkowników
- II — połączenia między węzłami komunikacyjnymi i koncentratorami
- III — połączenia między koncentratorami i punktami abonenckimi.

Stosunkowo wczesne podjęcie prac nad budową sieci RETD, które zmusiło autorów do poszukiwania rozwiązań oryginalnych, jest również źródłem pewnych kłopotów: RETD stoi wobec konieczności pełniejszego przystosowania swych protokołów komunikacyjnych do międzynarodowych zaleceń CCITT, a zwłaszcza — wobec konieczności adaptowania protokołu X.25 i protokołu liniowego HDLC, co powinno nastąpić już w najbliższym czasie. Pozwoli to na bezpośrednią współpracę RETD z innymi sieciami teleinformatycznymi w Europie. Natomiast pod względem sprzętowym RETD jest rozwiązaniem w znacznym stopniu samodzielnym: duże doświadczenie zawodowe CTNE, jako instytucji kierującej rozwojem telekomunikacji w Hiszpanii, pozwoliło na podjęcie trudu opracowania własnych

modeli urządzeń dla sieci transmisji danych, łącznie z procesorami komunikacyjnymi .M.in. w połączeniach I szczebla używane są modemy własnej konstrukcji o szybkości 9600 bodów, natomiast na II i III szczeblu — również oryginalne modemy o szybkości 1200/2400 bodów.

Dużo uwagi poświęcono zagadnieniom podniesienia warunków eksploatacyjnych sieci. Gęstość połączeń zaplanowano z 20-procentowym zapasem w stosunku do przewidywanych szczytowych obciążeń łączy, w 1977 r. wielkość ruchu w godzinach szczytu sięgała globalnie 100 tys. bodów. W celu zwiększenia niezawodności sieci zdublowano w niej wszystkie urządzenia w węzłach komunikacyjnych i punktach koncentracji. W rezultacie uzyskano tam średni czas pracy bezawaryjnej równy 32 dniom dla połączeń III szczebla oraz 28 dniom dla połączeń szczebli I i II (tj. dla węzłów komunikacyjnych i koncentratorów). Za „uszkodzenie” na poziomie III przyjmuje się przy tym taki stan, w którym przez co najmniej 2 min. niezawodny jest, z przyczyn technicznych, nie mniej niż 50% punktów abonenckich przyłączonych do jednego węzła lub koncentratora. Uszkodzenia w łączach na szczeblu I występują średnio w ilości 0,4 na dobę, przy czym średni czas naprawy wynosi 6,24 godz. W większości przypadków, ze względu na nadmiarowość strukturalną sieci, nie powodują one całkowitego zerwania łączności między węzłami. Średni czas przejścia pakietu przez sieć nie przekracza 1 s. dla ok. 86% przypadków.

Nadzór operacyjny nad siecią, włączając w to zarówno kontrolę jej stanu technicznego, natężenia ruchu na poszczególnych trasach, jak i identyfikację zgłaszających się użytkowników, ochronę dostępu do ich baz danych itp., zapewnia współpracujący z siecią podsystem SAD (*Servizio Auxiliar de Datos*) oparty na komputerze UNIVAC 1100. W ramach SAD użytkownicy mogą też korzystać z odpłatnych usług w zakresie przetwarzania danych. Tak więc tandemem RETD-SAD jest systemem ogólnodostępnych usług kompleksowych, obejmujących gromadzenie, przesyłanie i przetwarzanie danych.

CTNE — jako instytucja świadcząca te usługi — stara się stwarzać bodźce zachęty do korzystania z usług przez użytkowników różnych kategorii. Jest to widoczne m.in. na przykładzie taryfy opłat za transmisję danych: taryfa ta obejmuje stałą opłatę jednorazową za przyłączenie do sieci, opłatę miesięczną, zależną od rodzaju użytkowanego łącza i odległości punktu abonenckiego od najbliższego punktu wejściowego sieci, a wreszcie — opłatę zmienną, zależną od ilości przesłanej informacji, strefy i okresu, w których było użytkowane dane połączenie wirtualne. Tak pomyślany system taryfowy miał m.in. na celu wyrównanie szans abonentów znajdujących się zarówno w dużych aglomeracjach miejskich, jak i poza nimi.

W planach rozwojowych RETD na najbliższe lata przewiduje się m.in.:

- stopniową wymianę wyposażenia technicznego sieci na sprzęt o lepszych parametrach technicznych
- pełne dostosowanie protokołów komunikacyjnych sieci do standardów międzynarodowych CCITT (kolejne wersje protokołów serii X)
- powiązanie RETD z innymi sieciami teleinformatycznymi kontynentu europejskiego
- stworzenie możliwości dostępu do sieci poprzez łącza ogólnodostępnej komutowanej sieci telefonicznej i teleksowej
- wprowadzanie multipleksingu na III szczeblu połączeń sieci
- wprowadzenie automatycznego nadzoru techniczno-operacyjnego nad połączeniami wirtualnymi w sieci i in.

Autorzy projektu są przy tym pełni optymizmu, jeśli idzie o opłacalność handlową sieci, jakkolwiek podkreślają, że wymaga to z ich strony dużej przedsiębiorczości. Wypada również podkreślić parokrotnie deklarowaną przez nich gotowość nawiązania bliższych kontaktów z instytucjami odpowiedzialnymi za rozwój teleinformatyki w Polsce, potwierdzoną udostępnieniem znacznej ilości wartościowych materiałów informacyjnych.

Juliusz Lech KULIKOWSKI

## O cenach za wdrażanie systemów użytkowych

ETOB, podobnie jak większość organizacji zawodowo świadczących usługi informatyczne, korzysta z zetowskiego Cennika 1-U/75. Niepodważalną zaletą tego cennika jest jego... istnienie. Trudniej jednak chwalić to opracowanie, jeśli odbiorcy usług (z niezbyt jasnych powodów nazywani „użytkownikami”) chcą wiedzieć ile i za co mają płacić. A w budownictwie, zwłaszcza po wprowadzeniu tzw. nowego systemu ekonomiczno-finansowego, zainteresowanie cenami za usługi informatyczne wyraźnie wzrosło.

Trzymając się ściśle Cennika 1-U/75, jednostka informatyczna występuje w roli krawca, który zapytany o cenę uszycia garnituru odpowiada: — tego nie wiem, ale służę cennikiem za centymetry kroju i szycia, za przyszywanie guzików, prasowanie itp. Rozliczymy się po koszcie wynikowym. Co gorsze — wyceny za eksploatację systemów użytkowych, oparte na opłatach za czas korzystania ze sprzętu komputerowego, nie są dopingiem do tworzenia sprawnych technologicznie (a więc tańszych dla użytkownika) opracowań programowych i do usprawnienia organizacji przetwarzania danych. Wręcz przeciwnie.

Nie jest moją intencją generalne krytykowanie Cennika. Graniczyłoby to bowiem z tzw. krytykanctwem. Ważniejsze jest jednak, że cennik opracowany na miarę ówczesnych możliwości kadrowych i eksperymentalnych — wypełnił istniejącą lukę i ograniczył dowolność wycen za usługi informatyczne. Można go nawet bronić tezą, że jako wykładnia podstaw dla wycen maksymalnych nie ogranicza konstruktywnej inwencji w takim ustalaniu wycen rzeczywistych, aby ośrodki informatyczne nie narażały się na konflikty z użytkownikami i z etyką zawodową. Mimo to marginesy dowolności interpretacji wycen — zwłaszcza na korzyść ośrodków obliczeniowych — jest stanowczo za duży.

Organizacja ETOB, wykorzystując swe uprawnienia zjednoczenia, od od trzech lat systematycznie ryczałtuje ceny za usługi informatyczne w sposób zrozumiały dla użytkowników i nie hamujący motywacji ośrodków informatycznych do usprawniania technologii i organizacji przetwarzania danych. Za sprawniej wykonaną usługę ośrodek otrzymuje tę samą opłatę zryczałtowaną. Stosuje się przy tym zasadę, że cena zryczałtowana musi być niższa od tej, jaką dla większości przypadków użytkowych można by z Cennika wyliczyć.

Najwięcej bodaj zastrzeżeń budzą ustalenia Cennika dotyczące usług zwanych „wdrażaniem systemów lub pakietów standardowych”. Trudno oprzeć się wrażeniu, że pokutują tu nie tak dawne przecież czasy, gdy na modłę scholastyczną poważnie deliberowano nad irracjonalnym wręcz problemem definiowania systemów „powielanych”, „powtarzalnych”, „typowych” czy „standardowych”, zastanawiając się następnie (tu już o wiele słuszniej), jaki termin — „system” czy „pakiet” — jest merytorycznie poprawny. A chodziło w istocie o rzecz banalną i ogólnie zrozumiałą: o systemy informatyczne na tyle uniwersalne, że przydatne dla wielu użytkowników.

Dla ustalania opłat za pracę kadry ośrodka obliczeniowego przy wdrażaniu systemu przeznaczanego dla wielu użytkowników (nazywanego „standardowym”), choć jest to ewidentne niechlujstwo terminologiczne), podstawę cennikową stanowi albo koszt opracowania systemu (można wdrażać za 10% tego kosztu) albo liczbą stron dokumentacji systemu (można pobierać 490 zł za każdą stronę tej dokumentacji). Nietrudno stwierdzić, że najbardziej opłacalne jest wdrażanie systemów opracowanych za wygórowane ceny lub systemów rozwlekłe i nieudolnie opisanych w dokumentacji. Komplet zarzutów podstawowych trzeba uzupełnić faktem, że Cennik nie przewiduje różnicowania wycen w zależności od stopnia skomplikowania czynności wdrożeniowych tego samego systemu (np. duże czy małe, proste lub skomplikowane przedsiębiorstwo), jeśli nie liczyć enigmatycznego prawa do stosowania „opustów”.

Centrum ETOB wprowadziło dla swych ośrodków jednolite zasady wycen za usługi wdrożeniowe — w niniejszej prezentacji tylko do pracy kadry odnoszone — oparte na następujących założeniach.

• Umowną miarą stopnia pracochłonności wdrażania jest liczba elementów (pozycji) użytkowych przez system obsługiwanych. Zwykle pozycje te są także przedmiotem ryczałtowania cen za usługi eksploatacyjne. Ich przykłady: liczba środków trwałych w systemie ewidencji i rozliczenia środków trwałych, liczba maszyn budowlanych przy ewidencji i rozliczeniu pracy tych maszyn, przewidywana miesięczna liczba transakcji w systemie finansowo-księgowym. Jeżeli nawet tak rozumiana liczba elementów budzi wątpliwości jako precyzyjna miara praco-

chłonności wdrażania, to przynajmniej opłaty za tę usługę pobiera się od użytkowników w sposób sprawiedliwy: proporcjonalnie do późniejszych kosztów eksploatacji systemu.

• Stosuje się zasadę malejącej progresji cen, niżej opisaną.

• Ustala się dolną wartość ceny wdrożeń oraz odpowiadającą jej liczbę elementów (pozycji) obsługiwanych systemem.

• Ceny wdrożeń wszystkich lub zdecydowanej większości przewidywanych przypadków wdrożeń systemu powinny być niższe niż wynika to z kalkulacji wg Cennika 1-U/75, zwłaszcza w wersji opartej na liczbie stron dokumentacji, która jest na ogół tańsza i bardziej jednoznaczna od wersji opartej na cenie za opracowanie systemu.

• Precyzuje się jednoznacznie typowy zakres czynności wdrożeniowych, w tym zwłaszcza liczbę użytkowych przebiegów przetwarzania danych rzeczywistych, objętych usługą wdrożeniową poprzedzającą pierwszą typową usługą eksploatacyjną.

Algorytm progresji malejącej cen za wdrażanie systemu informatycznego jest oparty na funkcji o postaci:

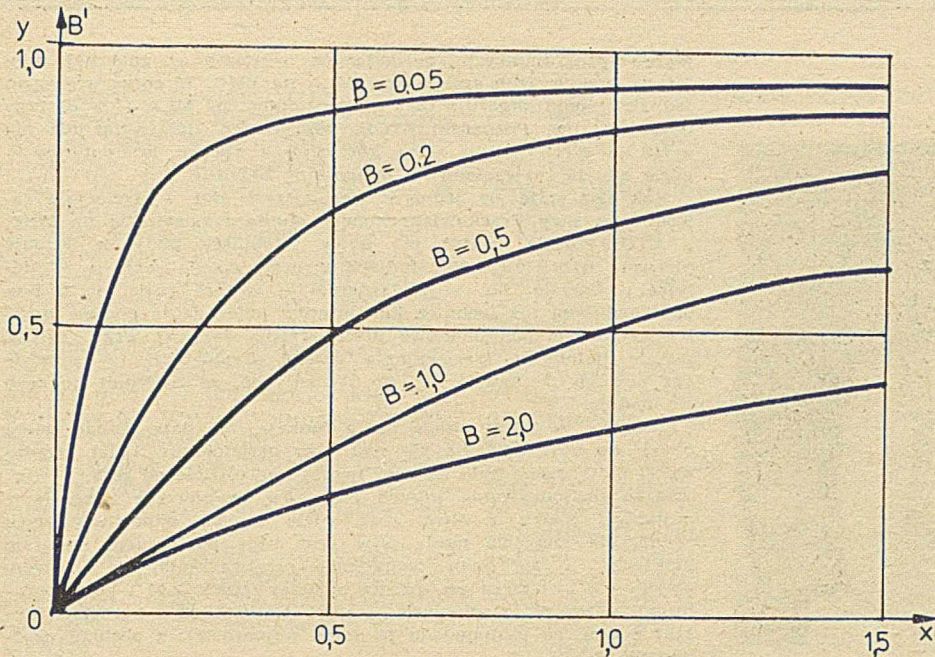
$$y = \frac{A}{1 + \frac{B}{x}} \quad (1)$$

gdzie  $A$  jest wartością asymptoty, do której zmierza  $y(x)$  przy  $x \rightarrow \infty$ ,  $B$  jest współczynnikiem doboru skali zmiennej  $x$ , łatwym do interpretacji praktycznej w dalszym wywodzie.

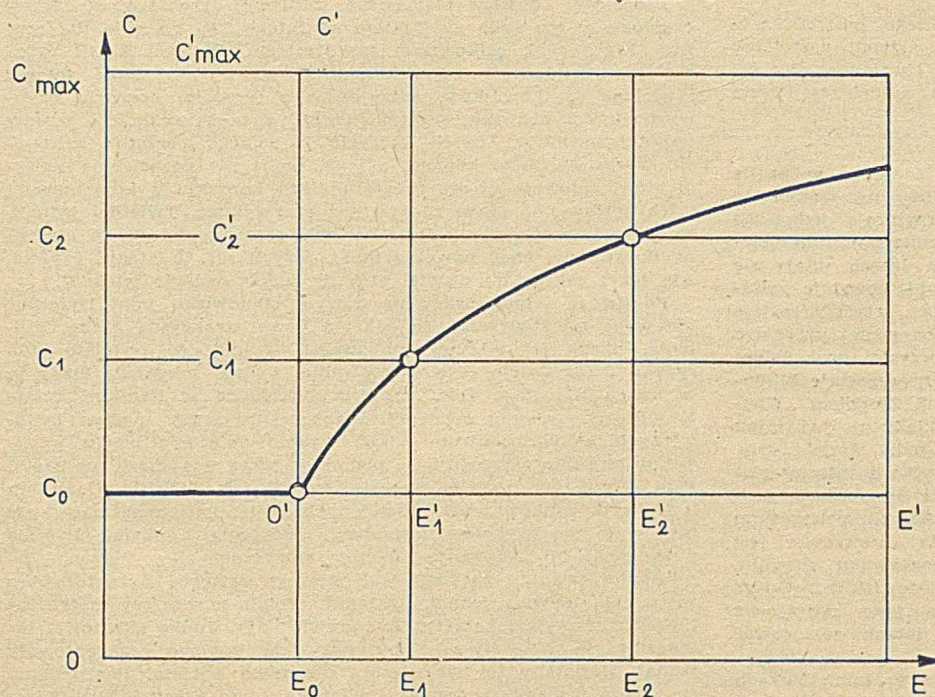
Ilustrację graficzną funkcji (1) przy  $A=1$  dla różnych wartości  $B$  przedstawiono na rys. 1.

Progresywny wzrost cen za wdrażanie systemów należy stosować od niezerowej ceny minimalnej  $C_0$ , przyporządkowanej niezerowej liczbie elementów obsługiwanych  $E_0$ . Sytuację tę zilustrowano na rys. 2, stosując dla uproszczenia dalszych wyrażań matematycznych, obok „bezwzględny” układ współrzędnych dla  $C=f(E)$ , gdzie:  $C$  — ceny wdrożeń (kadra),  $E$  — liczba elementów (pozycji) obsługiwanych — przesunięty układ współrzędnych dla  $C'=f(E')$  z początkiem w punkcie  $(E_0, C_0)$ .

Rysunek 1



Rysunek 2



Dla wyznaczenia przebiegu cen w zależności od liczby elementów, oprócz wspomnianego punktu  $(E_0, C_0)$ , należy — w wyniku odpowiednich analiz — wyznaczyć punkty  $(E_1, C_1) \equiv (E'_1, C'_1)$  oraz  $(E_2, C_2) \equiv (E'_2, C'_2)$ , możliwie od siebie odległe. Szczególną uwagę warto poświęcić wyborowi punktu  $(E_1, C_1)$ , bowiem powinien on być traktowany jako typowy lub przeciętny dla zastosowań danego systemu, stanowiąc formalne odniesienie porównawcze do Cennika 1-U/75. W idealnym przypadku punkt ten powinien spełniać równanie:

$$C(E_1) = \frac{1}{N} \sum_{i=1}^N C_i \quad (2)$$

gdzie:  $N$  — licznosc zbioru spodziewanych zastosowań systemu przy wszelkich wartościach  $E$ .

W praktyce wystarczy interpretować  $E_1$  jako średnią liczbę elementów obsługiwanych systemem w przewidywanym zakresie jego zastosowań. Faworyzując cenowo użytkowników charakter nieliniowości krzywej  $C(E)$  dla  $E < E_1$  jest częściowo skompensowany na korzyść ośrodków obliczeniowych ceną stałą  $E' < E_0$ .

Dla prostoty wyrażen matematycznych będziemy w dalszym ciągu posługiwać się „primowanym” układem współrzędnych  $(E' C')$ . Tak więc:

$$C' = \frac{C_{\max}}{1 + \frac{B}{E'}} \quad (3)$$

Stąd:

$$C'_{\max} = C' \left( 1 + \frac{B}{E'} \right) \quad (4)$$

Niewiadome  $C'_{\max}$  i  $B$  można wyznaczyć z układu równań dla punktów  $(E'_1, C'_1)$  i  $(E'_2, C'_2)$ :

$$C'_{\max} = C'_1 \left( 1 + \frac{B}{E'_1} \right) \quad (5)$$

$$C'_{\max} = C'_2 \left( 1 + \frac{B}{E'_2} \right) \quad (6)$$

Przyrównując te równania stronami, otrzymujemy po przekształceniach

$$B = \frac{C'_2 - C'_1}{\frac{C'_1}{E'_1} - \frac{C'_2}{E'_2}} \quad (7)$$

Dysponując — na podstawie analiz — danymi liczbowymi dla punktów  $(E'_1, C'_1)$ ,  $(E'_2, C'_2)$ , oblicza się najpierw wartość  $B$ , a po jej podstawieniu do wzorów (5) lub (6) — otrzymuje się wartość  $C'_{\max}$ . Wtedy wzór (3) staje się jednoznaczna relacją  $C' = f(E')$ , a po uwzględnieniu danych dla punktu  $(E_0, C_0)$  uzyskuje się — jak na rys. 2 — szukany przebieg funkcji  $C = f(E)$ . Wystarczy go przeanalizować i ewentualnie skorygować, posługując się niematematyczną już zasadą zdrowego rozsądku, aby dojść do ostatecznej formuły na wyznaczanie sprawiedliwych dla użytkowników cen maksymalnych za pracę kadry przy wdrażaniu danego systemu informatycznego, charakteryzującego się wielokrotnym zastosowaniem.

Tytułem przykładu z życia podaję, iż w odniesieniu do punktu  $(E_1, C_1)$  uzyskano w sieci ETOB następujące reprezentatywne wyniki, wprowadzone oczywiście do praktyki wdrożeniowej:

- dla systemu ewidencji i rozliczania środków trwałych: 50% wartości cennikowej, a przypadki maksymalne nie przekraczają 70% tej wartości
- dla systemu ewidencji i rozliczania pracy maszyn budowlanych: 60% wartości cennikowej, a wartość ta jest przekraczana dla ok. 10 przypadków na ponad 300 (gdy przedsiębiorstwo ma ponad 180 maszyn budowlanych)
- dla systemu finansowo-księgowego: ok. 35% wartości cennikowej, a przypadki maksymalne nie przekraczają ok. 65% tej wartości.

#### Z ostatniej chwili

Już po oddaniu tego tekstu do druku, Zjednoczenie Informatyki wprowadziło nowe zasady wycen za wdrażanie systemów. Dobrze, że prowadzą one na ogół do cen niższych od dotychczasowych. Szkoda jednak, że ich nie różnicują od wielkości i stopnia skomplikowania użytkownika, o czym mowa w artykule.

### Ewa Frąckowiak



Trudno oprzeć się wrażeniu, iż bohaterka tego portretu wyprzedziła o pokolenie to, co teraz „owczym pędem” nastolatek zwykło się określać. Ale też jaki trzeba mieć charakter, aby ów pęd nie przeszkodził rozsądnemu i dającym satysfakcję życia ułożeniu.

Zacznijmy jednak od początku.

Poznańska szesnastolatka, Ewa Szafranska, znajduje w prasie ogłoszenie i deklaruje swój udział w konkursie na wokalistkę Zespołu „Słask”. Konkurs wygrywa, choć przyjmują jedną na pięćset. Kulturowane od pokoleń rodzinne zamilowania do śpiewu tłumaczą, że rodzicom dopiero po dwóch latach udaje się ściągnąć dziewczynę do domu, bo w Poznaniu powstaje właśnie Zespół „Wielkopolska”, w którym młodzież artystka zostaje solistką. A że zespół to jednak amatorski, trzeba gdzieś pracować i zrobić wreszcie maturę i w ogóle żyć jak przystoi poznańskiej panience. W taki oto sposób na całe sześć lat Ewa zostaje bibliotekarką w Wojewódzkiej Bibliotece Publicznej. Z Zespołem „Wielkopolska” rozstanie się po dziesięciu latach, kiedy to już nijak występów artystycznych z pracą zawodową pogodzi się nie da.

Flirt z informatyką — a okaże się, że nie tylko z informatyką — rodzi się przypadkowo: z namowy koleżanki do podjęcia pracy w tworzonym wówczas (od 1 stycznia 1961 r.) poznańskim oddziale Biura Rozliczeń Budownictwa. Młoda dziewczyna jest skuszona ciekawszymi niż w bibliotece perspektywami rozwoju, a zapewne także nową techniką i nowym środowiskiem. Omija „damski fach”, tj. maszyny lekkie. Zaczyna jako operatorka sorterów, potem tabulatorów, by po dwóch latach awansować na kierowniczkę zmiany MLA.

Już po pierwszych kursach i zasmakowaniu w łamigłówkach programowych polyka bacyła nowego zawodu. Szybko wchodzi w skład grupy zapaleńców przetwarzania danych — jak sama to określa. Jest wśród nich także Feliks Frąckowiak, co dla pani Ewy spowoduje dożywotnie konsekwencje.

Po pięciu latach pracy owi zapaleńcy są już przekonani o potrzebie reorientacji zawodowej na rzecz przetwarzania elektronicznego. Tym bardziej, że nowy szef BRB — mgr Leon Judka — jest niecierpliw: jak najszybciej chciałby widzieć u siebie MIŃSK-32. Nie zobaczy go zresztą nigdy — najpierw z powodu braku pomieszczenia, a potem wobec nastania okresu maszyn ODR 1300. Doprowadził jednak do przeszkolenia grupy osób na MIŃSKU, co miało ważne ogólnokształcące znaczenie.

W tej wcześniej utworzonej komórce rozwojowej ETO zabrakło jednak miejsca dla naszej bohaterki. — Jesteś bardziej potrzebna przy MLA, bo tu na razie prawdziwa robota — tak jej oświadczoneo. Ambitna kobieta potraktowała tę decyzję ze zrozumieniem, ale też wyciągnęła praktyczne wnioski (nietaktem byłoby pytanie jaką tu rolę odegrał pan Frąckowiak). W 1966 r. podejmuje studia na WSE, kończy kursy COBOL-u, PLANU oraz projektowania systemów. Swoją cel osiąga dwa lata później: zostaje przeniesiona do komórki ETO jako analityk w dziale studiów i projektowania systemów. Bierze czynny udział w pracach pro-

jektowo-programowych, polegających wówczas w znacznej mierze na przeniesieniu systemów z MLA na EMC. W coraz większym jednak stopniu angażuje się — jak przedtem na MLA — w obsługę użytkowników. Poznański ETOB, poza ODR 1103, wciąż nie ma urządzeń ETO. Trzeba więc korzystać z obcych komputerów — gdzie się da, najczęściej w bydgoskim ETOB-ie.

Rok 1973 staje się jedną z ważniejszych dat w życiorysie zawodowym Ewy Frąckowiak: kończy studia magisterskie na WSE, a ETOB-Poznań, mając już nowy wspaniały budynek, kupuje pierwszą ODR 1305. Na studiach wybrała specjalizację w ekonomice przemysłu, bo — jak twierdzi — zawsze czuła się bliższa użytkownikom niż technice komputerów (choć zaraz podjęła dwuletnie studium podyplomowe przetwarzania danych). Przy ODRZE zostaje natomiast kierowniczką działu eksploatacji (w zasadzie bez obsługi użytkowników), bo stwierdzono, że najlepiej do tego się nadaje.

Sytuacja, w jakiej objęła nową funkcję, nie dała jednak wielu okazji do zastanawiania się nad tym paradoksem. Dział eksploatacji EMC trzeba było bowiem tworzyć całkowicie od zera. A więc uważne podpatrywanie innych ośrodków, wybieranie i szkolenie najlepszej kadry z MLA oraz wiele innych odpowiedzialnych zadań. Na szczęście miała przy tym silne partnerskie wsparcie koleżanki w sprawach technologii przetwarzania. Z upływem czasu — a zwłaszcza po zakupie drugiej ODRY 1305 i po reorganizacji służb wdrożeniowych — traci jednak coraz bardziej kontakt z tym, co pasjonowało ją dotąd najbardziej: z obsługą użytkowników.

Propozycja objęcia stanowiska zastępcy dyrektora do spraw eksploatacji (co stało się faktem od lutego 1977 r.) była dla mgr Ewy Frąckowiak całkowitym zaskoczeniem. W sieci ETOB oznacza to bezpośrednią odpowiedzialność za wykorzystanie sprzętu i obsługę użytkowników. Jako osoba o wysokim poczuciu odpowiedzialności obawiała się tej funkcji, a także reakcji i postaw współpracowników, wśród których od samego początku zawodowo wyrastała. Jako kobieta, a więc osoba nie pozbawiona sentymentów, żałowała utraty bezpośredniego kontaktu z komputerami i ich obsługą, do czego zdążyła już przywyknąć. Twierdzi jednak, że główną motywacją pozytywną była szansa powrotu do obsługi użytkowników, choć nawet z innej pozycji niż to kiedyś czyniła. Ale też i zakres tej obsługi stał się nieporównanie większy.

Po dwóch latach pracy na nowym stanowisku Pani Dyrektor nie waha się stwierdzić, że odkryła nową atrakcyjną cechę działania, bliską jej własnej osobowości — możliwość podejmowania ryzyka i satysfakcję jego wygrywania. I daje przykład, kiedy to w końcu ubiegłego roku założyła dokonanie w krótkim czasie 16 wdrożeń systemu ewidencji i rozliczania pracy maszyn budowlanych. Potem Centrum ETOB wydało polecenie pięciu wdrożeń w terminie niezgodnym z pierwszą piątką w ramach szesnastu. Pani Dyrektor podjęła decyzję hazardową: Wdrożono system u 16 użytkowników według ustalonych z nim harmonogramów, ale nie udało się wykonać formalnego polecenia Centrum. Pytania o słuszność tej decyzji jest raczej retoryczne.

Współpracownicy zaakceptowali nowego zastępcę dyrektora bez zastrzeżeń. Bądźmy jednak dokładni: zrobili to wszyscy mężczyźni i prawie wszystkie kobiety. To „prawie” dyr. Judka skwitował lapidarnie: — Kto słyszał, aby wszystkie kobiety zaakceptowały jedną? Musiałyby to być niezbyt normalna kobieta. Powszechnie się uważa, że swój autorytet zbudowała „pani Ewa” blisko 20-letnią pracą w ETOB-ie i jest on wsparty godnym uznania jej charakterem. Bardzo poznańskim charakterem — co bezstronnie dodać należy — w którym pracowitość, solidność, odpowiedzialność, a może nawet i skromność, są cechami dominującymi. Jakże bowiem inaczej można zrozumieć stwierdzenie naszej bohaterki, że nawet przy bardzo ambitnych zadaniach, główna satysfakcja zawodowa to wykonywanie tych zadań zgodnie z przyjętymi uprzednio założeniami, a podstawowym warunkiem osiągnięcia tego celu jest należyta organizacja pracy? I zaraz potem doda, że sprawą nie mniej ważną, to być dobrą żoną i matką. Naturalność wypowiedzenia tych stwierdzeń nie pozwala wątpić, że są to poglądy prawdziwe.

Swą przyszłość zawodową pani dyr. E. Frąckowiak traktuje też całkiem zwyczajnie. Po prostu uważa, że — tak jak dotychczas — najnormalniejszą w świecie rzeczą jest robienie stale czegoś nowego: usprawnianie technologii przetwarzania (już wdrożyła GEORGE'a 3), zdalny wielodostęp, wkrótce zapewne RIAD-32. A przede wszystkim — coraz lepsza i bogatsza obsługa użytkowników. Załuje tylko, choć już coraz mniej, że nigdy nie wróci do tego, co ją zawodowo pociągało najbardziej — do śpiewania.

(wład)



## Klub Użytkowników Minikomputera MERA 400

W ramach Specjalistycznego Ośrodka Współpracy i Koordynacji Branżowej ds. ETO w Projektowaniu Budownictwa powołano Klub Użytkowników Minikomputera MERA 400 w Projektowaniu.

Celem działania Klubu jest koordynacja wszystkich prac związanych z wdrażaniem minikomputera i jego oprogramowaniem. Cel ten będzie realizowany poprzez:

- organizację i koordynację prac nad oprogramowaniem użytkowym MERA 400 na potrzeby biur projektów
- ułatwianie wymiany programów pomiędzy użytkownikami
- organizację szkolenia w zakresie wykorzystania w opracowywanych projektach istniejącego oprogramowania
- ułatwianie przepływu informacji od użytkownika do producenta, ukierunkowanej pracy badawczo-rozwojowej związanej z oprogramowaniem podstawowym.

Potrzeba powołania Klubu wynika z troski użytkowników o jak najpełniejsze i najefektywniejsze wykorzystanie MERY 400 w procesie projektowania. Sprzęt ten jest jednak słabo wyposażony w programy dla obliczeń inżynierskich. Dzięki koordynacji sprawowanej przez Klub uniknie się opracowywania tych samych problemów przez różnych użytkowników, a tym samym przyspieszy się przygotowanie najpotrzebniejszych programów.

Członkiem Klubu Użytkowników Minikomputera MERA 400 może zostać każda organizacja gospodarcza, która jest zainteresowana zastosowaniem tego systemu w pracach projektowych. Przynależność do Klubu jest bezpłatna. Obecnie Klub liczy 14 członków, w tym 8 biur projektów o profilu budowlanym:

Zebranie inauguracyjne Klubu, które odbyło się w lokalu Centralnego Ośrodka Badawczo-Projektowego Budownictwa Przemysłowego (COBPBP) „BISTYP”, objęło:

- zwięzłą charakterystykę celów i zadań Klubu, przedstawioną przez zastępcę dyrektora ds. naukowych COBPBP „BISTYP”, dr inż. M. Robakiewicz

- prezentację obecnych na zebraniu użytkowników minikomputerów MERA 400, którzy scharakteryzowali posiadany sprzęt oraz zakres jego zastosowań

- omówienie i przyjęcie regulaminu Klubu, do którego zgłosiły się wszystkie reprezentowane na zebraniu biura projektów oraz Warszawskie Przedsiębiorstwo Geodezyjne; regulamin ten wyróżnia się szczególnie starannym sformułowaniem i wszechstronnym ujęciem zagadnień, uwzględniającym dotychczasowe doświadczenia innych klubów użytkowników.

- wybór Zarządu Klubu, w skład którego weszły następujące osoby: mgr inż. K. Kociakiewicz z Zakładu Nowych Metod Projektowania COBPBP „BISTYP” jako przewodniczący, mgr inż. Włodarski z Biura Projektów Przemysłu Lekkiego w Łodzi i mgr

inż. J. Zienkiewicz z poznańskiego Biura Projektów Budownictwa Przemysłowego jako dwaj wiceprzewodniczący oraz mgr A. Kchanowska z Zakładu Nowych Metod Projektowania COBPBP „BISTYP” jako sekretarz

- przedyskutowanie projektu planu pracy Klubu na 1979 r., w którym postanowiono uwzględnić następujące, zgłoszone w toku dyskusji wnioski: 1) przygotowanie „listy życzeń” dla producenta w odniesieniu do systemu operacyjnego MERY 400, 2) ogłoszenie w prasie o powstaniu Klubu oraz 3) udostępnianie źródłowych wersji programów.

Z przykrością odnotowano nieobecność zaproszonego przedstawiciela Biura Generalnych Dostaw Centrum Naukowo-Produkcyjnego Techniki Komputerowej i Pomiarów, który miał poinformować zebranych o stanie obecnym i planach rozwoju minikomputerów MERA 400.

Szczegółowo sformułowany wielopunktowy plan pracy Klubu na rok bieżący ujmuje następujące podstawowe tematy:

- wymianę informacji i doświadczeń w zakresie sprzętu oraz oprogramowania podstawowego i usługowego
- wymianę informacji i koordynację prac w zakresie oprogramowania użytkowego
- współpracę w zakresie szkolenia
- organizowanie bezpośredniej współpracy z producentem i innymi klubami.

Pierwszą część zebrania inauguracyjnego zamknęły wolne wnioski, spośród których w wyniku dyskusji przyjęto następujące ustalenia:

- nie zostaną wprowadzone deklaracje członkowskie
- członkowie Klubu otrzymają wzór karty informacyjnej programu
- członkowie Klubu w ciągu 10 dni przekażą Zarządowi informacje o prowadzonych pracach oraz aktualnie posiadanych programach
- członkowie Klubu będą informować Zarząd na bieżąco o wszystkich pracach prowadzonych w ramach przyjętego planu.

W drugiej części zebrania odbyła się prezentacja zastosowania MERY 400 zainstalowanej w Zakładzie Nowych Metod Projektowania COBPBP „BISTYP”. W ramach pokazu mgr inż. E. Kluzka zademonstrowała zebranym pakiet programów ŻELBET — 77M do wymiarowania przekrojów żelbetowych wg nowej normy.

Informując o powołaniu i planowanej działalności Klubu Użytkowników Minikomputerów MERA 400, redakcja INFORMATYKI pragnie pogratulować pomysłodawcom i organizatorom Klubu. Ich cenna inicjatywa powinna ożywić bardzo osłabioną ostatnio działalność innych niegdyś bardzo licznych klubów użytkowników. O bezspornych zaletach tego rodzaju inicjatyw społecznych — jako najbardziej skutecznej i sprawdzonej formy zwiększania efektywności sprzętu i zastosowań komputerów — nie trzeba nikogo przekonywać.

### LISTA CZŁONKÓW KLUBU UŻYTKOWNIKÓW MERY 400

Centralny Ośrodek Badawczo-Projektowy Budownictwa Przemysłowego „BISTYP”  
00-518 Warszawa, ul. Parkingowa 1  
mgr inż. K. Kociakiewicz, tel. 25-95-39  
mgr inż. M. Wiszomirski, tel. 25-96-18  
mgr A. Kochanowska, tel. 21-12-19

Instytut Maszyn Matematycznych  
02-078 Warszawa, ul. Krzywickiego 34  
mgr L. Partyka, mgr W. Soltys, tel. 21-84-41 w. 317

Warszawskie Przedsiębiorstwo Geodezyjne  
00-497 Warszawa, ul. Nowy Świat 2  
mgr inż. E. Pianko, tel. 21-44-61 w. 53

Biuro Projektów Przemysłu Mięsnego  
00-791 Warszawa, ul. Chocimska 28  
mgr inż. Barbara Jung, mgr inż. T. Tomkiewicz, tel. 49-80-10

Poznańskie Biuro Projektów Budownictwa Przemysłowego  
61-815 Poznań, ul. Ratajczaka 10/12  
mgr inż. Z. Kulas, tel. 6-22-31 w. 151

BPBBO „Miastoprojekt” — Łódź Miasto  
90-113 Łódź, ul. Traugutta 21  
mgr inż. M. Cichoński, tel. 3-81-19 lub 2-89-20 w. 137 lub w. 248

Biuro Projektów Przemysłu Lekkiego  
90-312 Łódź, Plac Zwycięstwa 2  
mgr inż. E. Włodarski, tel. 6-33-44 w. 113

Gdańskie Biuro Projektów Budownictwa Przemysłowego  
80-894 Gdańsk, ul. Garncarska 23  
mgr inż. B. Pamięta, tel. 31-28-31 w. 17

RSiPE „ENERGOPROJEKT”  
61-823 Poznań, ul. Piekary 19  
mgr inż. A. Czyżak, tel. 22-20-11 w. 186

Wrocławskie Biuro Projektów Budownictwa Przemysłowego  
50-066 Wrocław, ul. Świdnicka 19  
mgr inż. W. Czarnecki, mgr inż. I. Grabkowski, tel. 3-80-41 w. 136

PPiM „MERAL”  
02-456 Warszawa, ul. Czereśniowa 96  
mgr inż. R. Koropacki, mgr inż. M. Bieracki, tel. 23-76-01 w. 245

Instytut Dróg i Mostów, Politechnika Warszawska  
00-611 Warszawa, ul. Filtrowa 1  
mgr inż. M. Winiarski, tel. 25-80-16

Miastoprojekt — Katowice  
40-036 Katowice, ul. Wita Stwosza 2  
mgr inż. M. Grymel, tel. 51-20-41 w. 308

Biuro Projektów Budownictwa Komunalnego  
30-107 Kraków, Pl. Na Stawach 1  
mgr Antoni Przytocky, tel. 587-00

W. KLEPACZ

## Udane, tyle że nie „targi”



Ceremonię zamknięcia „softargów” uświetnili swoją obecnością: sekretarz KW PZPR w Katowicach, Zdzisław Grudzień, i Minister Przemysłu Maszynowego, dr inż. Aleksander Kopeć

Zdjęcie: Jacek Tomczek

Nie licząc różnego typu „zaściankowych” giełd systemów, po raz pierwszy zdobyliśmy się na solidny przegląd oprogramowania zrealizowanego ostatnio przez rozmaite ośrodki informatyki. W dniach 26–31 marca br. w katowickim Ośrodku Postępu Technicznego ponad 120 instytucji zaprezentowało swoje najcenniejsze systemy.

Wprawdzie zamiysł zorganizowania ekspozycji powstał w związku z dorobkiem software'owym ośrodków Ministerstwa Przemysłu Maszynowego, ale później, na etapie konkretyzacji projektu, generalny realizator — Instytut Organizacji Przemysłu Maszynowego — zdecydował się uwzględnić zgłoszenie instytucji spoza własnego resortu, głównie wyższych uczelni, instytutów MNSzWiT, PAN, ale także ośrodków z innych resortów. W sumie reprezentowanych było 7 resortów.

W ten sposób na wystawie zgromadzono w postaci oferty do kupna i wymiany ponad 700 pakietów, przy czym zbiór był dość niejednorodny, albowiem w jego skład wchodziły (zgodnie zresztą z założeniami), zarówno programy sterowania procesami technologicznymi, jak i: programy obliczeń inżynierskich, programy i pakiety programów technologicznych i użytkowych, programy szkolenia w zakresie nowoczesnych technik programowania. Nadto dopuszczalne było

prezentowanie osiągnięć z dziedziny projektów organizacyjnych i technologicznych ośrodków komputerowych i wszelkich pomysłów racjonalizatorskich z dziedziny przetwarzania informacji.

Po zwiedzeniu wystawy Minister Przemysłu Maszynowego, dr inż. Aleksander Kopeć, powiedział: *Największym osiągnięciem jest sam fakt, że stać nas już na zorganizowanie tak bogatej ilościowo i tak dobrej jakościowo ekspozycji.*

Po latach mozolnej produkcji najprostszych systemów ewidencyjnych — częstokroć tych samych w różnych ośrodkach — w Katowicach można było nabrać przekonania, że informatyka wkracza nareszcie w dziedzinę, w których jest najbardziej potrzebna i oczekiwana, w których z jej stosowania płyną największe korzyści.

Nie chcę przez to powiedzieć, że różne gospodarki materiałowe, listy płac i bilanse siły roboczej nie zasługują na zaangażowanie komputera, ale zbyt często byliśmy świadkami „uników” ośrodków obliczeniowych przed trudnymi tematami, podczas gdy na drogim, wysokiej klasy komputerze „tłuczono” przez trzy zmiany na okrągło to, z czym równie dobrze poradziłyby sobie maszyna księgująca lub licząco-analityczna. Dla której z kolei nie jest dostępny np. system modelowania procesów ciągłych (brązowy medal), czyli pakiet programów pracujących pod OS/JS, a przeznaczonych do

rozwiązywania zagadnień opisanych równaniami różniczkowymi lub analogowymi schematami blokowymi. Ponieważ powyższy przykład wybrałem z katalogu produktów Centrum Naukowo-Produkcyjnego Systemów Sterowania w Katowicach, trudno pominąć że ta renomowana firma była jednym z największych wystawców na „softtargach” (21 produktów), a prezentowane przez nią programy zdobyły kilka medali.

Sporo zastrzeżeń budzi natomiast oficjalna formuła wystawy — „targi”. Wydaje się, po pierwsze, że na „targi” to u nas jeszcze za wcześnie, po drugie, że organizatorzy chcieli zignorować to, co im wiadomo, a co zamieścili w „Informacji prasowej”, opracowanej na okoliczność „softtargów”:

*...Jednocześnie jednak obserwuje się u nas szereg ujemnych zjawisk w produkcji oprogramowania. Są one następujące: brak wystarczającej informacji o produktach programowych oraz projektowanych w tym zakresie przedsięwzięciach, (...) brak ścisłych zasad obrotu oprogramowaniem, (...) brak działalności marketingowej.*

Otóż! Nie były to targi — gdyż nie na nich nie utargowano. Dodajmy — gdyż nie można było utargować.

Informacja nie była dokładna. Wiele stoisk nie dysponowało zbiorczym katalogiem eksponowanych systemów. Wydano wprawdzie katalog całej wystawy, ale rozszedł się błyskawicznie, tak szybko, że nawet na konferencji prasowej nie udało się pokazać bodaj jednego egzemplarza (Minister Aleksander Kopeć: *Należy natychmiast zwiększyć nakład*). Informacje o prezentowanych systemach nie zawierały żadnej informacji o cenie produktu, sposobie przejęcia i wdrażania etc.

Ostatni zarzut tłumaczy się... „brakiem ścisłych zasad obrotu oprogramowania”. Jak długo zasad tych się nie sformułuje — nie ma mowy o targach. Zaś „brak działalności marketingowej” również wyklucza formułę targową. Choćby dlatego, że system komputerowy nie jest parą butów, które wystarczy na targach obejrzeć i... zakontraktować całą ich partię. System trzeba dobrze zbadać w działaniu. W konkretnym przypadku wymagałoby to albo zainstalowania sprzętu informatycznego na terenie targów, albo — wcześniejszych badań wzajemnych potrzeb.

Ale to nie formuła imprezy katowickiej wymaga korekty. Targi są potrzebne. Aby więc mogły nosić w przyszłości tę nazwę, należy stworzyć po temu warunki, o których mowa wyżej. O istniejące niedomogi mogą się rozbić liczne intencje „kupna — sprzedaży”, których — jak zapewnili organizatorzy — objawiło się w czasie wystawy niemało. Sprawa wyceny systemów jest tym bardziej istotna i pilna, że przyszłoroczne softtargi zapowiadają się jako impreza RWPG.

Mimo powyższych uwag „softargi” należy ocenić wysoko. Splendoru imprezie dodał niewątpliwie protektorat Ministra Przemysłu Maszynowego, udział w ceremonii otwarcia wiceministra MPM, prof. dr hab. inż. Stanisława Paszkowskiego, a w ceremonii zamknięcia — gospodarza województwa, Zdzisława Grudnia, który autorem wyróżnionych 40 pakietów wręczył przyznane przez jury złote, srebrne i brązowe medale Ministra MPM.

Oto nagrodzeni:

#### Medal złoty

1) Centrum Naukowo-Produkcyjne Systemów Sterowania MERA-STER w Katowicach — za system oprogramowania MISS-80

2) Zakłady Urządzeń Komputerowych MERA-ELZAB w Zabrze — za system oprogramowania VIDEO-ELZAB  
3) Fabryka Samochodów Małolitrażowych w Bielsku-Białej — za system oprogramowania KADRA

4) Instytut Organizacji Przemysłu Maszynowego w Warszawie — za system oprogramowania HANDEL ZAGRANICZNY

#### Medal srebrny

1) Centrum Naukowo-Produkcyjne Systemów Sterowania MERA-STER w Katowicach — za system oprogramowania SKARB

2) Centrum Komputerowych Systemów Automatyki i Pomiarów MERA-ELWRO we Wrocławiu — za system oprogramowania WINT

3) Instytut Maszyn Matematycznych w Warszawie — za kompilator języka PASCAL

#### Medal brązowy

1) Centrum Naukowo-Produkcyjne Systemów Sterowania MERA-STER w Katowicach — za system oprogramowania MODEL

2) Fabryka Samochodów Osobowych w Warszawie — za system oprogramowania GESTOR

3) Instytut Elektrotechniki w Warszawie — za system oprogramowania SILNIKI.

Odnotujemy też wiele imprez towarzyszących — m.in. symposium naukowe z bogatym i zróżnicowanym programem, wystawę książek o tematyce informatycznej i różne drobne pokazy, jak np. prezentacja... wieszaka do taśm magnetycznych — urządzenia taniutkiego, ale jakże potrzebnego w warunkach powszechnej ciasnoty.

Wystawa cieszyła się stałym powodzeniem. Przez pawilon OPT przewijały się setki ludzi, co w końcu dowodzi i wzrostu zastępów informatyków i utrwalaniu się informatyki w świadomości społecznej.

Krystyn BERNATOWICZ

## Szkoła Naukowa „Sieci Komputerowe”

W dniach 23—28 marca br. odbyła się w Karpaczu III Zimowa Szkoła Naukowa pod nazwą „Sieci komputerowe”. Tegoroczna Szkoła poświęcona była komunikacji w sieciach komputerowych.

Celem Szkoły było pogłębienie wiedzy, wymiana doświadczeń i konfrontacja poglądów w dziedzinie modelowania, weryfikacji projektowania i eksploatacji podsieci komunikacyjnej.

W szkole wzięli udział przedstawiciele nauki i specjaliści w dziedzinie projektowania i badań sieci komputerowych z ośrodków badawczo-naukowych Czechosłowacji, Francji i Polski. Oprócz środowiska wrocławskiego reprezentowane były następujące ośrodki krajowe: warszawski (IPI PAN, WAT, Centrum Informacji Energetycznej, OBRI), śląski (Politechnika Śląska), gdański (Politechnika Gdańska), krakowski (AGH) i poznański (Politechnika Poznańska).

Szkolę zorganizował Instytut Cybernetyki Technicznej Politechniki Wrocławskiej przy współpracy Instytutu Podstaw Informatyki PAN oraz Instytutu Cybernetyki Stosowanej (IAC) w Bratysławie. Program Szkoły został opracowany pod patronatem Rady Programowej, a w szczególności: doc. M. Bazewicza (Politechnika Wroclawska), prof. J. Seidlera (IPI PAN Warszawa) i dra R. Porizka (IAC Bratysława).

Zajęcia prowadzono w formie wykładów i seminariów problemowych. Na seminariach omawiano zagadnienia poruszane dnia poprzedniego na wykładach. Oto tematyka poszczególnych se-

sji oraz przewodniczący kolejnych dyskusji seminaryjnych:

1) projektowanie węzła sieci komputerowej (L. Niemczycki — Centrum Informatyki Energetycznej, D. Rutkowski — Politechnika Gdańska)

2) komunikacja na poziomie modemów (M. Rojewski — Politechnika Gdańska, W. Mochnacki — Politechnika Wroclawska)

3) sterowanie przepływem i kontrola błędów (J. Pużman — FMI Praga, A. Badach — Politechnika Gdańska)

4) architektura sieci (R. Pożizek — IAC Bratysława, J. Barchański — ICT Politechniki Wroclawskiej).

Na uwagę zasługuje duży wkład prof. J. Seidlera oraz zespołu z Politechniki Gdańskiej w pogłębienie wiedzy w zakresie metodologii i tworzeniu podstaw teoretycznych i projektowania sieci komputerowych. Zaprezentowali oni sposób podejścia do projektowania sieci komputerowych ze szczególnym uwzględnieniem zasad projektowania i doboru parametrów pojedynczego węzła sieci oraz organizacji przesyłania pakietów. Zespół IPI PAN prezentował wyniki prac dotyczących pomiarów statystycznych podsieci komunikacyjnej. Zespół z IAC Bratysława zapoznał słuchaczy z metodami weryfikacji i oceny protokołów komunikacyjnych, natomiast ośrodek Sekcji Transmisji Danych Federalnego Ministerstwa Techniki i Inwestycji z Pragi zajął się sterowaniem błędami i przepływem. Przedstawiciel Uniwersytetu w Pradze przedstawił tematykę związaną z transmisją danych w kanałach radiowych, poruszył

zagadnienia metod statystycznych i pomiarów sieci, a reprezentant Akademii Nauk z Pragi podjął problem parametrów stanowiących podstawę do oceny wydajności w komunikacji komputerowej.

Poza programem przedstawiciel Międzyuniwersyteckiego Centrum Obliczeniowego w Grenoble omówił rozwiązania problemów komunikacji we francuskiej sieci „CYCLADES” oraz oprogramowanie telekomunikacyjne i sieciowe komputerów przyłączonych do tej sieci, opracowane w wymienionym Centrum.

Ośrodek Politechniki Wrocławskiej zaprezentował zagadnienia detekcji i korekcy błędów w sieciach z zastosowaniem protokołów liniowych, problemy modelowania i symulacji w sieciach teleinformatycznych oraz strategię retransmisji w sieciach z komutacją pakietów. Omówiono również architekturę logiczną sieci systemów otwartych, zagadnienia całościowej oceny funkcji użytkowych sieci oraz zarządzanie buforowaniem w węzle sieci komputerowej.

Treść wykładów i komunikatów ogłoszonych w ramach tegorocznej Szkoły Zimowej zostanie opublikowana w Polsce przez wydawnictwo Politechniki Wrocławskiej oraz w Czechosłowacji przez wydawnictwo Akademii Nauk. Będzie to druga część skryptu pt. „Sieci komputerowe” pod redakcją M. Bazewicza (pierwsza część skryptu poświęcona została problemom protokołów i ukazała się na początku bieżącego roku w bibliotece WASC Politechniki Wrocławskiej).

M.B.

## Uczelniane ośrodki obliczeniowe

Dzisiaj prawie nie można sobie wyobrazić szkoły wyższej bez urządzeń do elektronicznego przetwarzania danych — czytamy w informatorze o zasadach korzystania z ośrodka obliczeniowego uniwersytetu w Tybindze. Słowa te najlepiej chyba określają obecny status nauk informatycznych w krajach Europy Zachodniej. Takie podejście sprzyja tworzeniu dobrze wyposażonych i zorganizowanych uczelnianych ośrodków obliczeniowych. Dla nas, studentów informatyki, wizyty w takich ośrodkach były szczególnie interesujące.

Największe różnice między tym, co zobaczyliśmy w trakcie wycieczki, a tym, do czego przywykliśmy „na własnym podwórku”, polegają przede wszystkim na:

— znacznie większej mocy obliczeniowej komputerów, a dzięki temu — krótszym średnim czasie rozwiązywania zadania

— olbrzymiej pojemności pamięci zewnętrznej, przydzielanej użytkownikom prawie bez ograniczeń

— pracy interakcyjnej jako podstawowej formie nauczania; przykładowo zajęcia z laboratorium programowania prowadzone są w salach wyposażonych w kilkanaście terminali — i wreszcie

— na wielu różnych nie tyle kosztownych co pomysłowych rozwiązaniach, bardzo ułatwiających pracę (przodował w tym ośrodek obliczeniowy w Lozannie).

### TYBINGA

#### — SYSTEM SOHO

Do Tybingi przyjechaliśmy na zaproszenie prof. Graefa, dyrektora ośrodka obliczeniowego tamtejszego uniwersytetu.

Ośrodek jest wyposażony w dwa komputery: CDC 3300 z 112 K słów i TR 440 z 384 K słów pamięci operacyjnej. W ciekawy sposób rozwiązano tutaj pracę konwersyjną, a ściślej dostęp do terminala do komputera. Do maja 1977 roku obie maszyny były wykorzystywane w sposób tradycyjny, tzn. jako dwie odrębne jednostki obliczeniowe. Potem zastosowano oprogramowanie umożliwiające połączenie obu komputerów. Głównym celem tego połączenia było stworzenie warunków swobodnego wyboru dowolnej maszyny dla użytkownika terminala. Oznacza to, że terminal, który sprzętowo był kompatybilny z kom-

puterem TR 440 i TR 86S, może również służyć jako terminal CDC 3300.

Minikomputer TR 86S jest komputerem satelitarnym maszyny TR 440. Na TR 86S został zaimplementowany system SOHO, odpowiedzialny za programową realizację wspomnianego połączenia. Sprzętowo realizację zapewnia łącznie transmisji danych o prędkości przesyłania 9600 bodów, prowadzące od TR 86S do CDC 3300. Łącze to umożliwia jednocześnie korzystanie z 16 terminali, przyłączonych do CDC 3300 poprzez podsystem MCS III.

SOHO realizuje konwersję danych, kontrolę przepływu informacji oraz inne czynności związane z transmisją. Użytkownik terminala za pomocą specjalnej dyrektywy może wybrać komputer, na którym chce pracować. Od chwili, gdy połączenie także zostanie zrealizowane, musi stosować rozkazy sterujące tego właśnie komputera.

Oprócz opisanego wyżej połączenia, w przyszłości przewiduje się podobne z komputerem CYBER w Stuttgarcie. Ponieważ procedury zarządzające dla komputerów CDC 3300 i CYBER są zgodne, nie będzie prawdopodobnie żadnych technicznych przeszkód przy tworzeniu tego połączenia, natomiast trzeba będzie rozwiązać wiele problemów organizacyjnych, m.in. sprawę podziału kosztów eksploatacji. Do tego czasu przesyłanie obliczeń do Stuttgartu i otrzymywanie wyników na urządzeniach wyjściowych w Tybindze możliwe będzie tylko w trybie wsadowym, za pomocą zainstalowanego w Tybindze urządzenia CD 200-UT Remote-Job-Entry Station Batch-Jobs.

Rozwiązanie zrealizowane w Tybindze umożliwia jednolity dostęp do terminala do zbioru heterogenicznych komputerów, co ułatwia stworzenie sieci komputerowej łączącej różne ośrodki obliczeniowe na bazie istniejącego sprzętu. Stworzenie takiej sieci planuje się np. w Zurichu, gdzie prowadzone są obecnie prace nad włączeniem ośrodka uczelnianego do międzynarodowej sieci komputerowej, obejmującej Paryż, Medialan, Istria i Londyn.

### ZURICH — SYSTEM ZE WSPÓLNĄ PAMIĘCIĄ

Jadąc do Zurichu mieliśmy nadzieję, że poznamy osobiście prof. Wirtha, o którym słyszeliśmy już wtedy, gdy stawialiśmy pierwsze kroki jako adepci wiedzy informatycznej. Niestety był okres urlopow, dlatego też naszym przewodnikiem po Instytucie Infor-

matyki ETH (odpowiednik politechniki) byli asystenci profesora.

W Zurichu szczególnie zainteresował nas sposób organizacji pracy maszyn wchodzących w skład wyposażenia tamtejszego ośrodka obliczeniowego. Są to komputery: CDC 6400 (czyli taki jak u nas w Świerku, lecz tam uważany za maszynę małą), CDC 6500 i CYBER 174. Komputery te pracują niezależnie od siebie, korzystając jednak ze wspólnego obszaru pamięci dyskowej. W celu zorganizowania wspólnej pracy tych maszyn stworzono nowy system operacyjny. Jest to połączenie systemu SCOPE 3.4.1. (znane go użytkownikom komputera w Świerku), LONDPES (LONDON Permanent File System) oraz QMAN (Queene MANager). Najważniejszą częścią tego systemu jest moduł zarządzający stałymi zbiorami danych. System zapewnia synchronizację dostępu do danych i ochronę przechowywanej informacji (przed zniszczeniem i niepowołanym dostępem).

Konfiguracja taka pozwala na znaczne usprawnienie wszelkich działań związanych z pobieraniem i zapisywaniem informacji dla pamięci zewnętrznej. Ponieważ właśnie dostęp do pamięci masowej jest najczęściej wąskim gardłem systemów komputerowych, zwłaszcza w przypadku częstych operacji na bazach danych, rozwiązanie przyjęte w Zurichu zwiększa efektywność całego systemu i pozwala lepiej wykorzystać możliwości obliczeniowe poszczególnych komputerów.

Organizacja wspólnej pamięci wewnętrznej dla kilku komputerów ma wiele zalet: umożliwia jednolity dostęp do wspólnej bazy danych systemu (w szczególności eliminuje redundancję przechowywanych informacji), a także bardzo łatwe komunikowanie się zadań przetwarzanych na różnych komputerach (np. zbiór wyjściowy procesu przetwarzanego na CDC 6500 może służyć jako „wejście” dla procesu przetwarzanego na maszynie CYBER).

Podobne rozwiązanie planuje się w Monachium, gdzie dwa komputery CYBER 175, stanowiące część wyposażenia ośrodka obliczeniowego Bawarskiej Akademii Nauk, będą korzystały nie tylko ze wspólnej pamięci wewnętrznej (którą w tym przypadku będzie pamięć typu *cartridge storage*), lecz także z rozszerzonej pamięci rdzeniowej (*extended core memory*).

## Marzenia szalonego studenta

Znalazłem błąd w programie i musiałem go poprawić. Przyszedłem więc rano do ośrodka obliczeniowego. Przy urządzeniach końcowych było jak zwykle pusto. Szybko nawiązałem łączność z systemem, napisałem poprawki i przyłączyłem bibliotekę z dysku. Wprowadziłem poprawki, skompilowałem program, obejrzałem na monitorze listing i uruchomiłem program dla próbnych danych. Ponieważ wynik wymagał głębszej analizy, z ołówkiem w rękę, skopiowałem go na kopiarce i poszedłem obok do pokoju cichej pracy.

Po sprawdzeniu okazało się, że poszło dobrze i mogłem uruchomić program dla danych rzeczywistych. Ponieważ kart z danymi było sporo, program miał trwać długo, a spodziewane wyniki powinny zająć kilka tysięcy wierszy, zdecydowałem się na pracę usadową. Dane miałem już przepisane, ale musiałem dopisać karty sterujące. Poszedłem do sali dziurkarek. Sytuacja trudna — dziurkarek powinno być w eksploatacji piętnaście, ale dwie już kilka dni są remontowane, a dwie spośród pozostałych niewyraźnie opisują. Podobno mają być sprawdzone urządzenia do bezpośredniego zapisu na dysk, wystarczą wówczas znacznie mniej dziurkarek.

Wziąłem ze stojaka karty w odpowiednich kolorach, wyperforowałem, złożyłem cały pakiet i poszedłem do czytnika. Sam nie wiem, czy lepsze jest rozwiązanie z czytnikiem ogólnodostępnym, czy — tak jak było poprzednio — z przekazywaniem kart przez specjalną służbę operatorom. Wydaje się, że teraz jest wygodniej — jeżeli jednak ma się dla operatora szczególne zlecenia?

Po chwili na monitorze nad drzwiami wyświetliła się kolejka wejściowa. Ze względu na wymagany czas i rozmiar pamięci, znalazłem się na ostatniej, piątej pozycji. Przed uruchomieniem trzeciej maszyny nie można chyba liczyć na poprawę. Obecny zestaw był kupowany trzy lata temu, ma więc prawo być przeciążony. Czekałem cierpliwie.

Po dziesięciu minutach zadanie zaczęło się realizować. Trwało to około kwadransa, gdy wreszcie na monitorze stojącym w hallu zobaczyłem swoje wyniki (6 tys. wierszy) w kolejce do drukarki. Z tym przynajmniej nie było problemów — drukarka stała właśnie wolna i po pięciu minutach zakończyłem pracę. Zabrałem wydruk z odpowiedniej półki. W sumie trwało wszystko pół godziny, jednak długo w porównaniu z trybem konwersacyjnym.

Bajka? Nie, rzeczywistość oglądana w ośrodkach uczelnianych w RFN i Szwajcarii. Tak mniej więcej wygląda praca studenta z komputerem.

A oto garść informacji dodatkowych.

Tryb interakcyjny jest rzeczywiście powszechnie stosowany. Komputery są wyposażone w ekranowe urządzenia końcowe, zgrupowane w specjalnych salach (Lozanna) lub rozsianych na całym terenie uczelni (Berno, Zurich). Do utrwalenia wyników pracy służą specjalne kopiarki. Jest to coś w rodzaju kserografu połączonego przewodem z terminalem — po naciśnięciu odpowiedniego przycisku z urządzenia tego wysuwa się arkusz papieru zawierający kopię informacji aktualnie wyświetlanej na ekranie.

W niektórych ośrodkach (Berno) używane są monitory ekranowe graficzne, umożliwiające wprowadzenie dowolnych rysunków i wykresów. W połączeniu ze wspomnianą kopiarką daje to zaiste wspaniałe możliwości.

Wszystkie ośrodki są oficjalnie wyposażone w pamięci dyskowe. Użytkownik może z nich korzystać zupełnie swobodnie, a ewentualne ograniczenia są ustalone na bardzo wysokim poziomie pojemności (np. w Bernie — 1 MB na osobę). Skasowanie pliku ze względu na brak miejsca jest czymś zupełnie nieznanym i niepojętym.

Do doskonałym rozwiązaniem (Berno, Lozanna) są pokoje cichej pracy, ze stolikami i zbiorem podręczników, znajdujące się tuż obok sal przygotowywania danych lub pracy konwersacyjnej. Pozwala to na spokojną analizę wyników.

Przygotowywanie danych na kartkach jest nadal powszechnie stosowane ze względu na pracę w trybie wsadowym. Ośrodki są jednak wyposażone w wystarczającą liczbę dziurkarek. Wszędzie stosuje się różne kolory i nadruki kart do różnych zastosowań, np. w Zurichu i Lozannie na kartach końcach zadania znajdowała się lista instrukcji maszynowych CDC, w Bernie, natomiast, na kartach przeznaczonych na początek zadania (instrukcja JOB na IBM) był podany krótki opis dozwolonych parametrów. Przodował chyba Zurich, gdzie dla większości dostępnych języków programowania (ALGOL/SIMULA, COBOL, FORTRAN, BASIC, COMPASS) stosowa-

no specjalne, wyróżniające się karty. Sytuacja, w której z powodu braku normalnych kart otrzymuje się karty przeznaczone do gromadzenia danych z obserwacji meteorologicznych, w tamtych ośrodkach nie grozi...

Przy trybie wsadowym kontakt użytkownika z komputerem jest organizowany rozmaicie. W większości ośrodków (Tybinga, Zurich, Berno) należy wrzucić karty wraz z ewentualną instrukcją oraz taśmami magnetycznymi i pakietami dyskowymi do specjalnej służby, która jest natychmiast opróżniana przez operatorów. W Lozannie natomiast czytnik (z poziomym prowadzeniem kart) stoi w pomieszczeniu dostępnym dla studentów i jest przez nich bezpośrednio obsługiwany.

Przeływ zadania przez system w Zurichu i Lozannie można obserwować na monitorach ustawionych w różnych newralgicznych punktach ośrodka (sale dziurkarek, korytarze itp.). Monitory te cyklicznie wyświetlają wszystkie kolejki systemowe. Czas wykonania zadania przy pracy wsadowej nie jest zazwyczaj duży, a pół godziny uznane jest za wartość niemal nie do przyjęcia.

Wydrukowane zadania są na ogół rozkładane przez operatorów na półki. Np. w Lozannie były to specjalne stojaki, na których wydruki były układane według alfabetu. W Bernie ściana dzieląca użytkowników od Sali komputera zawierała szereg półek dostępnych z obu stron, przy czym każda półka była przypisana określonej grupie użytkowników. Operator wkładał wydruk z jednej strony, a użytkownik mógł go zabrać z drugiej strony półki.

Wszystko to znakomicie ułatwia pracę studenta z komputerem. Co więcej, wiele rozwiązań (półki, czytnik ogólnie dostępny, kolorowe karty) może być zrealizowanych właściwie wszędzie, bez większych nakładów, a jedynie przy dobrych chęciach i sprawnej organizacji. Inne rozwiązania, jakkolwiek kosztowne (monitory wyświetlające kolejki), poprawiają wydajność pracy, umożliwiając lepsze wykorzystanie komputera. Sądzę więc, że warto by się zastanowić nad wprowadzeniem podobnych rozwiązań również w naszych ośrodkach.

Jarosław DEMINET

## EuroIFIP 79

Ostatni światowy kongres IFIP odbył się w Toronto (1977 r.), następny odbędzie się w Tokio i Melbourne (1980 r.), czyli również poza Europą. W poważnym stopniu ogranicza to udział w kongresach przedstawicieli naszego kontynentu. Prezydium IFIP postanowiło więc zorganizować w bieżącym roku EuroIFIP 79.

EuroIFIP 79 odbędzie się w dniach od 25 do 28 września br. w Londynie, na terenie nowego ośrodka konferencyjnego Wembley (Wembley Conference Center). Bezpośrednim organizatorem EuroIFIP 79 jest stowarzyszenie informatyków brytyjskich BCS (British Computer Society). Konferencji będzie towarzyszyć wielka wystawa produktów i usług informatycznych.

Głównym tematem konferencji będą praktyczne i ekonomiczne aspekty zastosowań komputerów. Równoległe będą prezentowane nowe osiągnięcia w dziedzinie teoretycznych podstaw zastosowań, jak również cele, osiągnięcia i zamierzenia takich agend IFIP, jak IAG (Applied Information Processing Group), Komitety Techniczne IFIP (IFIP Technical Committees) oraz ich grupy robocze (Working Groups).

Zgodnie z tradycją, oprócz referatów nadesłanych, organizatorzy zaprosili do wygłoszenia referatów kilku największych specjalistów. W referatach i dyskusjach zostaną szczególnie uwypuklone związki między poziomem rozwoju technologicznego a rozwojem gospodarczym oraz współzależności w rozwoju informatyki różnych krajów europejskich.

Organizatorzy spodziewają się, że 4-dniowe obrady i dyskusje przyczynią się do lepszego zrozumienia problemów występujących na obecnym etapie rozwoju informatyki oraz kierunków jej rozwoju.

Komitet programowy konferencji:

przewodniczący — S. A. Overgaard (Norwegia), zastępca przewodniczącego — D. Firnberg (W. Brytania), członkowie — A. W. Goldsworthy (Australia), T. Kitagawa (Japonia), H. Kopetz (Austria), K. H. Müller (NRD), M. Paul (RFN), D. Szmylewski (ZSRR), P. Sorensen (Dania), J. Vlietstra (Holandia), H. Zimmermann (Francja) oraz PASomet (BCS) jako redaktor materiałów konferencyjnych.

Przedmiotem obrad siedmiu sekcji tematycznych będą następujące problemy:

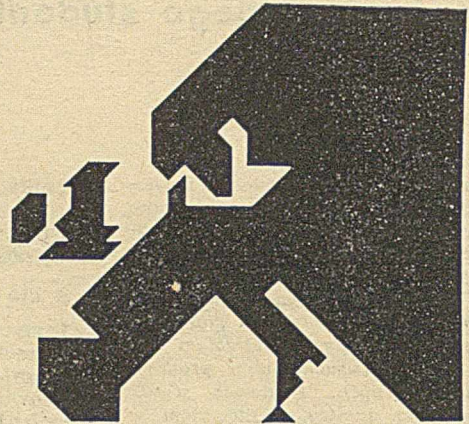
1) Rozwój, eksploatacja i zarządzanie systemami informatycznymi (Development, operation and management of computer-based systems)

— organizacja i zarządzanie projektowaniem, projektowanie bazy danych oraz środków przetwarzania, kryteria i metody wyboru sprzętu, rozwiązania scentralizowane i zdecentralizowane, instalacje pochodzące od wielu dostawców, planowanie i kierowanie opracowaniem oprogramowania, narzędzia do testowania i kontroli jakości programów, normy w programowaniu, kierowanie i konserwacja złożonych systemów oprogramowania, analiza kwalifikacji zawodowych, pomiar i poprawa efektywności wykorzystania komputera oraz poziomu usług ośrodka obliczeniowego, metody księgowania w warunkach pracy wieloprogramowej i konserwacyjnej, efektywność wielkich usługowych ośrodków obliczeniowych oraz rozproszonych systemów mini- lub mikrokomputerowych

2) Wspomagane komputerem procesy zarządzania w gospodarce i administracji państwowej (Computer-supported administrative processes in business and government)

— projektowanie procedur dla zastosowań komputerów, koncepcje, teoria i praktyka organizacji stosującej komputer, procesy programowane i nieprogramowalne, zarządzanie czynnościami informatycznymi, zależność od poprawnego funkcjonowania procedur informatycznych, kierowanie i kontrola księgowania, centralizacja i decentralizacja czynności decyzyjnych i wykonawczych, wpływ pracowników oraz warunki pracy, bazy danych oraz dostępność danych w procesach administracyjnych, przetwarzanie tekstów, archiwowanie i komunikacja w skomputeryzowanym biurze

3) Sterowanie komputerem procesów technologicznych i środowiska naturalnego człowieka (Computer control of technical and environmental processes)



— sterowanie przemysłowymi procesami produkcyjnymi (sterowanie laboratorium), robotyka, sterowanie pojazdami i ruchem ulicznym, sterowanie komputerowe telekomunikacją, sterowanie komputerowe eksploatacją systemów komputerowych (systemy operacyjne), modele środowiska, nadzorowanie zasobów naturalnych oraz zużycia materiałów i energii, kontrola zanieczyszczeń, nadzorowanie i przewidywanie zjawisk naturalnych (pogody, powodzi, zjawisk sejsmicznych itp.), nadzorowanie procesów biologicznych łącznie z medyczną opieką nad pacjentem, modele biologiczne, modele i rzeczywistość

4) Komunikacja i koordynacja komputerowa (Computer communication and coordination)

— techniki, dostępność i koszty usług transmisji danych, wpływ nowych technik transmisji danych, łączenie sieci, operacje koordynujące w komunikowaniu się komputerów, zasadność sieci komputerowych, zarządy łączności i przemysł usług transmisji danych — współzawodnictwo czy współpraca? usługi transmisji danych w epoce sieci

5) Wspomaganie komputerowe procesów twórczych i analitycznych (Computer aid to the creative and analytical mind)

— projektowanie wspomagane komputerem, grafika komputerowa, przetwarzanie obrazów, metody symulacyjne i szacowanie, modele programowania gospodarczego, komputery w naukach społecznych, lingwistyka i tłumaczenia wspomagane komputerem, komputer narzędziem twórczości artystycznej, symulacja komputerowa procesów intelektualnych, języki i algorytmy do opisywania procesów, informatyka rozrywkowa

6) Komputery w społeczeństwie (Computer in society)

— problemy etyczne ery informacji, ustawodawstwo i porozumienia na temat ochrony i użycia danych, techniki ochrony danych, metody kryptograficzne, przepływ danych przez granice państwowe, słabości społeczeństwa wysoko skomputeryzowanego, zwalczania przestępstw komputerowych, ochrona praw w odniesieniu do metod, programów i danych, warunki pracy w środowisku sterowanym komputerem, komputery w szkole i w domu, wpływ kultury komputerów, problemy, cele i przyszłość europejskiego przemysłu komputerowego, Europa i światowy rynek usług informatycznych, planowanie państwowe dla skomputeryzowanego społeczeństwa przyszłości

7) podstawowe narzędzia i metody zaspokajania potrzeb użytkowników informatyki (The basic tools and fundamental methods to meet users needs in data processing)

— specjalizowane i uniwersalne systemy komputerowe, nowe technologie i metody programowania, wpływ mikrokomputerów, procesy przebiegające równoległe i ich koordynacja, struktury danych, pamięć i zarządzanie, inżynieria programowania i stosowane narzędzia, wydajność i niezawodność sprzętu, poprawność i zdolność konserwacyjna programów.

Wiele z powyższych tematów wymaga szczególnego zainteresowania ze strony polskich placówek badawczych.

## Największy na świecie

Ostatnio słyszy się wiele o superkomputerach. Należy do nich zaliczyć maszyny CYBER 203, CRAY 1 i DAP. Najpotężniejszą wydaje się CYBER 203 firmy Control Data Corporation (CDC).

Od ok. 30 lat firma CDC oferuje komputery uważane za największe w momencie wypuszczenia ich na rynek (modele 1601, 3600, 6600, 7600). Więcej uwagi projektanci przykładali jednak do architektury niż do technologii.

Badania nad superkomputerami o strukturze równoległej doprowadziły do zbudowania komputera STAR 100, którego 4 egzemplarze są aktualnie eksploatowane. Opanowanie technologii LSI pozwoliło firmie CDC na wprowadzenie ulepszeń zarówno w architekturze, jak i w technologii komputerów. Tak powstał CYBER 203. Jako komputer kompatybilny ze STARTEM 100 jest od niego 6 razy szybszy w przetwarzaniu tablic i ciągów znaków. Pamięć operacyjna sięga 2 milionów słów 64-bitowych (16 MB). Obecnie prowadzone są prace mające na celu zwiększenie pamięci operacyjnej do

8 milionów słów (chodzi o zastąpienie mikromodułów o pojemności 1 K przez mikromoduły 4 K).

Jednostka centralna CYBERA 203 zawiera (zgodnie z terminologią CDC) procesor skalarny, procesor wektorowy, pamięć operacyjną i podsystem wejścia/wyjścia. Szybkość wykonywania obliczeń przez procesor skalarny wynosi 50 milionów operacji zmiennoprzecinkowych na sekundę. 256 ogólnych rejestrów umożliwia 2 odczyty i jeden zapis w jednym cyklu. Procesor wektorowy, wykorzystywany do przetwarzania tablic, osiąga szybkość 100 milionów operacji na sekundę, wykonywanych równoległe z przetwarzaniem skalarnym. Podsystem wejścia/wyjścia zawiera 12 kanałów o przepływności do 40 Mbitów/s i umożliwia użycie CYBERA 170 jako procesora człowiego.

Cena komputera będzie wahać się między 5,8 a 11,7 milionów dolarów. CDC spodziewa się sprzedać około 200 takich komputerów, z czego 80 w Europie Zachodniej.

CYBER 203 znajduje zastosowanie między innymi w prognozowaniu meteorologicznym, wymagającym bardzo złożonych i szybkich obliczeń (według amerykańskich specjalistów komputer ten 5-krotnie przyspieszy uzyskiwanie wyników), oraz w symulacji zjawisk aerodynamicznych, eksploatacji złóż naftowych i badań nuklearnych.

Zamówiono już 3 egzemplarze CYBERA 203 dla sił lotniczych USA oraz jeden dla sieci CYBERNET, w której maszyna ta zastąpić ma STARA 100.

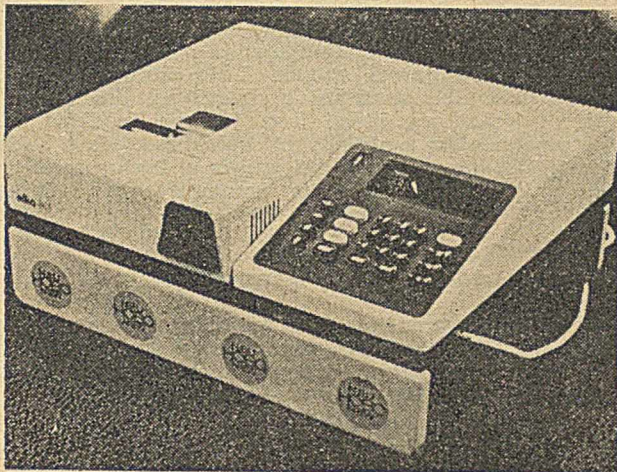
Wydaje się, że rosnące wymagania użytkowników spowodują zapotrzebowanie na jeszcze szybsze komputery, o szybkości rzędu 1 miliarda operacji na sekundę. Na badania związane z produkcją modelu 203 firma CDC wydała około 100 milionów dolarów. Badania są nadal prowadzone. Nowy model ma być 4—8 razy szybszy niż CYBER 203...

Oprac. A. NAWROCKI na podstawie ZERO-UNIFORMATIQUE N° 525, styczeń br.

### ELEKTRONICZNA KASA REJESTRUJĄCA ELKA 80

Elektroniczna rejestracja kasowa to usprawnienie Waszej działalności handlowej

Elektroniczna kasa rejestrująca ELKA 80 przeznaczona jest do szerokiego zastosowania we wszystkich rodzajach obsługi handlowej: zapewnia szybką, wygodną i bezbłędną pracę oraz całkowitą dokładność operacji pieniężnych



#### Charakterystyka techniczna:

- 3 rejestry akumulujące dla grup towarowych
- 1 rejestr akumulujący dla niezależności anulowanych
- 4 liczniki — po jednym dla każdego rejestru
- akumulacyjny i dodatkowy tryb pracy
- sygnał dźwiękowy w przypadku prawidłowo zakończonej operacji
- drukarka SEIKO — 2,5 wiersza/s
- kasowa rejestracja taśmy czekowej i kontrolnej; drukuje datę i numer kasy
- automatyczne podliczenie rachunku
- mnożenie, klawisz „korekcja”, odczyt i zerowanie rejestrów akumulujących
- zachowanie informacji nie dłużej niż miesiąc przy wyłączeniu zasilania z sieci
- dwa wskaźniki sześciocyfrowe
- wymiary: 460 × 400 × 180 mm, łącznie z sejfem
- masa: 19 kg

**Eksporter:**  
Przedsiębiorstwo  
Handlu Zagranicznego

**Isotimpex**

Bulgaria, Sofia  
ul. Czapajewa 51  
Telefon: 73-61  
Teleks: 022731, 022732

WCT/668/K/79

# Zastosowanie pakietu programów SOD w CPN Poznań

W Przedsiębiorstwie Obrotu Produktami Naftowymi CPN Poznań od 1975 r. przystąpiono do intensywnego wdrażania systemów informatycznych. Celem maksymalnego skrócenia czasu wdrożenia zakupiono początkowo trzy systemy już eksploatowane w innych przedsiębiorstwach (gospodarki środkami trwałymi, materiałami, finansowo-księgowej). W 1976 r. opracowano już we własnym zakresie i wdrożono system gospodarki towarami.

Działania te umożliwiły likwidację zaległości w pracach służb rachunkowości, wyeliminowały pracę w nadgodzinach, zahamowały przyrost zatrudnienia oraz znacznie zwiększyły operatywność kontroli, mimo corocznego 10-procentowego wzrostu obrotów przedsiębiorstwa.

Eksploatacja systemów informatycznych pozwoliła również zebrać opinię o ich bieżącej i perspektywicznej przydatności dla CPN. Stwierdzono m.in., że poszczególne systemy przetwarzania danych mają następujące wspólne cechy:

- kontrola danych źródłowych, redagowanie i zakładanie zbiorów roboczych oraz ich sortowanie
- aktualizacja danych stałych (części indeksowej)
- modyfikacja danych zmiennych
- redagowanie i drukowanie tabulogramów wynikowych.

Analiza trudności powstających w czasie eksploatacji wymienionych systemów, a wynikających głównie z konieczności stosowania różnorodnej dokumentacji projektowo-programowej, pozwoliła ustalić, że istotna różnica między poszczególnymi systemami EPD nie tkwi w stosowaniu dokumentów źródłowych o różnej strukturze i tworzeniu różnych struktur zapisów (rekordów) na nośnikach magnetycznych, lecz tylko w zakresie redagowania i drukowania tabulogramów. Zasady kontroli danych źródłowych, aktualizacji danych stałych oraz przetwarzania danych źródłowych opartego na użyciu podstawowych operacji arytmetycznych są praktycznie jednakowe. Wszystkie te operacje mają na celu zapewnienie prawidłowej dwustronnej (*Wnien* i *Ma*) księgowej ewidencji ilościowo-wartościowej lub wartościowej.

W konsekwencji nasunął się wniosek, że całkowicie możliwa jest integracja wszystkich eksploatowanych systemów na potrzeby rachunkowości. Jednocześnie zwrócono uwagę na udostępniany przez ELWRO pakiet programów SOD (System Operowania Danymi), pozwalający użytkownikowi w najprostszy sposób przetwarzać dane na komputerach ODRA 1304 lub 1305.

## Organizacja przetwarzania za pomocą systemu SOD

Podjęte w 1976 r. i kontynuowane dotąd prace pozwalają stwierdzić, że integracja całości przetwarzania danych na potrzeby rachunkowości w oparciu o system SOD jest nie tylko możliwa, ale z uwagi na niewątpliwe efekty — konieczna. W chwili obecnej zintegrowany system informatyczny w CPN Poznań obejmuje:

- gospodarkę środkami trwałymi
- gospodarkę przedmiotami nietrwałymi
- gospodarkę materiałami
- gospodarkę finansowo-księgową
- sporządzanie list płac dla pracowników o stałym wynagrodzeniu.

Realizację tego systemu oparto na:

- zastosowaniu standardowych programów SOD oraz jednolitych planów/operacyjnych przetwarzania
- zastosowaniu powszechnie używanych dokumentów źródłowych
- przyjęciu jednolitej struktury rekordów zbioru głównego

- oparciu identyfikacji na Typowym Planie Kont, Kodzie Towarowo-Materiałowym (KTM, SWW) oraz GUS-owskim klasyfikatorze środków trwałych
- zastosowaniu kart dziurkowanych i taśm magnetycznych.

Wspomniane programy standardowe zostały dostarczone przez ELWRO na bibliotecznym taśmie magnetycznym w postaci gotowej do eksploatacji. Wszystkie niezbędne adaptacje tych programów odbywają się za pomocą parametrów wczytywanych z kart dziurkowanych.

## Dokumenty źródłowe

Eksploatacja systemu w CPN Poznań nie wymaga zmiany powszechnie stosowanych dokumentów źródłowych. Dane z tych dokumentów pracownicy rachunkowości zapisują na pomocniczym dokumencie — poleceniu księgowania DZ-016, który umożliwia bezinstrukcyjne przygotowanie kart dziurkowanych. Dokument DZ-016 zawiera w jednym wierszu (jednej karcie dziurkowanej) dwa identyfikatory główne dla strony *Wnien* i *Ma*. Umożliwia to utworzenie w komputerze dwóch jednakowych rekordów, zapewniających zgodny zapis księgowy po stronach *Wnien* i *Ma*.

W chwili obecnej DZ-016, podobnie jak karta dziurkowana, jest dokumentem pośrednim między tradycyjnym dokumentem źródłowym a komputerem. W przyszłości przewiduje się zmianę obecnych dokumentów źródłowych na nowe, przystosowane do bezinstrukcyjnego przenoszenia danych, co wyeliminuje lub w zasadniczy sposób zmniejszy użycie dokumentów DZ-016.

Oprócz DZ-016 system wymaga sporządzania dokumentu DZ-014, przeznaczonego do aktualizacji części stałej (indeksowej) zbioru głównego. Za pomocą tego dokumentu wprowadza się nowe lub aktualizuje już istniejące rekordy na taśmie magnetycznej. W rekordach tych zawarte są takie informacje, jak: identyfikator główny, nazwa lub nazwisko, adres, ceny, jednostki miary, stan na początek roku.

Poza wymienionymi dokumentami w przedsiębiorstwie zaprzestano prowadzenia ewidencji ilościowo-wartościowej (rejstry, dzienniki, zestawienia, kartoteki ilościowo-wartościowe i finansowe, księgi inwentarzowe środków trwałych i przedmiotów nietrwałych), którą zastąpiono tabulogramami. Zbędne maszyny księgujące i fakturujące przekazano przedsiębiorstwom nie korzystającym z komputerów.

## Struktura rekordu w zbiorze głównym

Dla każdego szczegółowo rozliczanego czynnika gospodarczego (środka trwałego, materiału, towaru, pracownika) utworzono znormalizowany rekord o maksymalnej pojemności 2048 znaków (512 słów 24-bitowych). Całkowicie zabezpiecza to obecnie potrzeby systemu.

Program PROVE X680 (kontrola, redagowanie i zapisywanie danych źródłowych na taśmie magnetycznej)

Program PROVE realizuje następujące funkcje:

- wczytuje opracowane przez użytkownika parametry, sprawdza ich poprawność ze standardowymi założeniami programu oraz je drukuje; w razie błędów drukuje komunikaty wykazujące wykryte błędy
- wczytuje i drukuje dane źródłowe, wskazując wykryte błędy
- zamienia niestandardowe kody perforacji na standardowy kod komputera
- rozróżnia rodzaje dokumentów źródłowych, umożliwiając kontrolę poprawności danych wejściowych oraz tworzenie rekordów na taśmie magnetycznej



- sprawdza poprawność wczytanych informacji w każdym polu, np. czy pole zawiera wymagane znaki alfabetyczne (FIXA), alfanumeryczne (FIXC), cyfry (FIXP) oraz czy nie przekracza założonych wielkości, np. od 130 do 200
- prawidłowe dane wejściowe przekształca na format wewnętrzny i zapisuje na taśmie magnetycznej
- zamienia pola numeryczne zapisane znakowo na pole binarne
- oblicza i drukuje sumy kontrolne dla każdego typu danych
- umożliwia włączenie do programu segmentów własnych użytkownika, np. podprogramu kontrolującego poprawność identyfikatora za pomocą cyfry kontrolnej
- łączy (dopisuje) zbiory dokumentów źródłowych na jednej taśmie magnetycznej.

Program UPDATE X68F (aktualizacja danych stałych zbioru głównego)

- Program UPDATE realizuje następujące funkcje:
- wczytuje do pamięci operacyjnej opracowane przez użytkownika parametry, sprawdza ich poprawność ze standardowymi założeniami programu oraz je drukuje; w razie błędów drukuje odpowiednie komunikaty
  - przetwarza dane zapisane w zbiorze roboczym PRO-VEK3X680 i zbiorze głównym (stara generacja) oraz tworzy zaktualizowany zbiór główny (nowa generacja) i zbiór zmian C
  - drukuje tabulogram zawierający dotychczasową i nową treść zbioru głównego.

Program GAMP X68A (modyfikacja danych zmiennych)

- Program GAMP realizuje następujące funkcje:
- kontroluje parametry zgodnie z zasadami obowiązującymi w programach PROVE i UPDATE
  - przetwarza dane znajdujące się w zbiorach wejściowych głównym i roboczym
  - dokonuje operacji na polach zbioru głównego, tworzy jego nową generację, redaguje oraz tworzy podzbiory do wykorzystania w dalszym przetwarzaniu
  - przeprowadza badanie na trzech poziomach modyfikacji
  - przygotowuje na taśmie magnetycznej treść tabulogramów wynikowych.

Program REPORT X68G (redagowanie i drukowanie kartoteki ilościowo-wartościowej)

Działanie programu REPORT różni się od innych programów SOD tym, że parametry muszą być poddane dodatkowemu przebiegowi kompilacji. Otrzymuje się program w języku PLAN, zapisany na taśmie magnetycznej, który realizuje następujące funkcje:

- łączy dane z maksymalnie trzech taśm magnetycznych
- zapewnia drukowanie części stałych: a) na początku przebiegu, b) na początku każdej strony, c) w tekście, d) na końcu każdej strony, e) w przerwach kontrolnych, f) na końcu przebiegu

- wyznacza lub pomija rekordy uwzględniane w obliczeniach
- wykonuje podstawowe operacje arytmetyczne
- drukuje tabulogramy wynikowe.

Program FIND-2 X63D (redagowanie i drukowanie zestawienia obrotów i sald oraz księgi inwentarzowej środków trwałych)

Mimo że program FIND-2 nie należy do zestawu „Systemu Operowania Danymi”, opracowywanie dla niego parametrów nie odbiega od zasad obowiązujących dla programów UPDATE i GAMP. Program ten realizuje następujące funkcje:

- sprawdza parametry zgodnie z zasadami obowiązującymi dla poprzednich programów
- bada (przeszukuje) zbiór wejściowy, np. zbiór główny, oraz tworzy nową taśmę magnetyczną zawierającą tylko wybrane rekordy
- zapewnia wykonanie zadań omówionych w programie REPORT
- umożliwia sporządzanie różnych typów sprawozdań wynikowych (PRINT, LIST, TOTAL, TABLE).

## Zalety i Wady Pakietu Sod

System Operowania Danymi wyróżnia się następującymi zaletami:

- znacznie ułatwia i ujednocza projektowanie dzięki zastosowaniu rozwiązań typowych dla przetwarzania danych gospodarczych
  - prawie całkowicie eliminuje błędy programowania
  - pozwala wiązać bezpośrednio działanie różnych programów
  - zapewnia łatwe opanowanie sposobu opracowywania słowników i parametrów dla konkretnego systemu przetwarzania danych
  - pozwala użytkownikowi włączać do programów standardowych własne segmenty; zwiększa to elastyczność programów i daje znaczną swobodę programowania szczegółów
  - nie powoduje konieczności wprowadzania zmian w dotychczas stosowanych dokumentach źródłowych
  - umożliwia korzystanie z istniejących już zbiorów danych na różnych nośnikach maszynowych.
- Najbardziej odczuwalnymi wadami SOD są:
- przystosowanie niektórych programów wyłącznie do korzystania z taśm magnetycznych
  - zmniejszenie efektywności wykorzystania czasu pracy komputera
  - ograniczenie wielkości rekordu do 512 słów.

W naszej praktyce wady te okazały się jednak mało znaczące i są całkowicie rekompensowane prostotą i łatwością stosowania SOD.

Mgr Rafał LYKOWSKI  
Przedsiębiorstwo Obrotu  
Produktami Naftowymi CPN  
Poznań

### Tenis po raz drugi

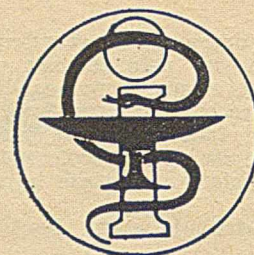
Zapoczątkowany w ubiegłym roku cykl dorocznych mistrzostw informatyków w tenisie będzie kontynuowany na kortach Spartakusa w Jeleniej Górze w dniach 5-9 września br. Oprócz sprawdzenia sportowych kwalifikacji uczestników, tak jak uprzednio i tym razem organizatorom — Zjednoczeniu Informatyki, Federacji

Sportowej Energetyki i Zakładowi ETO w Jeleniej Górze — przyświeca cel dalszej integracji środowiska.

Ponieważ już w roku ubiegłym chętnych było więcej niż miejsc, organizatorzy ponownie byli zmuszeni ograniczyć udział w mistrzostwach. W sumie na korta Jeleniej Góry zmieści się 16 pań i 32 panów. Selekcja chętnych będzie zapewne sprawą niełatwą, zwłaszcza że impreza będzie miała charakter otwarty — to znaczy, że mogą

brać w niej udział także informatycy spoza sieci Zjednoczenia. Warto podkreślić, że organizatorzy słusznie skorzystali z doświadczeń poprzednich mistrzostw — turniej przedłużono o 1 dzień i poszerzono o 1 konkurencję (gra mieszana). Termin zgłoszeń upływa z dniem 1 lipca. (KB)

## Informatyka na Farmacji



Liczne już i wielokierunkowe zastosowania informatyki w farmacji skłaniają do rozważań nad przygotowaniem informatycznym absolwentów wszystkich kierunków studiów farmaceutycznych<sup>1)</sup>.

Niestety problem ten zbyt długo był niedostrzegany i do dziś nie został całkowicie rozwiązany. Nie znaczy to jednak, że nic w tym kierunku nie zostało zrobione. Pierwsze próby nauczania informatyki studentów farmacji mają już za sobą Wydziały Farmacji w Gdańsku, w Poznaniu i w Warszawie.

W programie studiów farmaceutycznych znajduje się co prawda przedmiot „Ekonomiczne i organizacyjne zagadnienia farmacji”, ale np. na warszawskim Wydziale Farmacji przedmiot ten przewidziany jest jedynie dla studentów kierunku aptecznego (jeden semestr na IV roku<sup>2)</sup>).

Wobec braku obowiązującego, ministerialnego programu nauczania „Ekonomicznych i organizacyjnych zagadnień farmacji”, Samodzielna Pracownia Organizacji i Ekonomiki Służb Farmaceutycznych AM w Warszawie opracowała własny program. W ra-

mach wykładów i ćwiczeń z tego przedmiotu studenci zapoznają się z podstawowymi zagadnieniami informatyki.

Zagadnienia prezentowane studentom nie są jednak jeszcze dopracowane i rozwnięte; autorzy programu wykazują zdecydowanie lepszą znajomość zagadnień sprzętowych niż zagadnień oprogramowania i zastosowań.

Celem programu winno być nie tyle nauczenie farmaceutów informatyki, co ukształtowanie w nich świadomości, że jest ona w ich działalności niezbędna, a zwłaszcza czemu ma służyć i jakie są jej możliwości teraz i w najbliższej przyszłości. Chodzi tu głównie o przygotowanie absolwentów wydziałów farmaceutycznych do nieuniknionej roli użytkowników systemów informatycznych.

Oprócz treści informatycznych o charakterze encyklopedycznym warto nauczyć przyszłych farmaceutów również nowoczesniejszego myślenia w zakresie problematyki organizacji.

Kształcenie informatyczne studentów farmacji powinno w większej mierze uwzględniać zagadnienia projektowania i wdrażania systemów informatycznych niż często zbyt szczegółowe opisy sprzętu i programowania komputerów.

Program nauczania informatyki nie może być oczywiście programem zamkniętym. Aby nadążyć za niezwykle dynamicznym rozwojem tej gałęzi wiedzy, powinien być często aktualizowany, zwłaszcza pod kątem możliwości krajowych oraz światowych osiągnięć i trendów rozwojowych.

\* \* \*

Wydaje się, że proces nauczania informatyki studentów farmacji w dotychczasowej, prawie zaimprovizowanej formie nie może przynieść oczekiwanych efektów. Brak autoryzowanego programu o sprecyzowanej treści i formie, brak ciągłości i konsekwencji w jego realizacji, związany głównie z brakiem odpowiednio przygotowanej kadry wykładowców, brak własnego sprzętu komputerowego i trudności z dostępem do takiego sprzętu powinny zainteresować właściwe instancje uczelniane i resortowe. Pozostawienie rozwoju sytuacji nielicznym inicjatywom entuzjastów unowocześniania farmacji i całej służby zdrowia niesie poważne niebezpieczeństwo, którego bolesne skutki odczujemy w najbliższych latach. Potrzebne są pilnie konkretne propozycje w zakresie programów nauczania na wszystkich wydziałach kierunkach studiów farmaceutycznych w Polsce, a także kursów dla studiów podyplomowych przeznaczonych dla absolwentów czynnych już zawodowo.

Wskazane byłoby również powołanie Sekcji Informatyki przy Zarządzie Głównym i Oddziałach Polskiego Towarzystwa Farmaceutycznego w celu popularyzowania najbardziej efektywnych zastosowań informatyki w farmacji, a także włączenie zagadnień informatyki stosowanej do zakresu tematyki farmaceutycznych prac naukowych.

**Maciej PAPROCKI**

<sup>1)</sup> Wydziały Farmacji istnieją na ośmiu akademiach medycznych w Polsce (a jest ich dziesięć). Program studiów na ostatnich dwóch latach — czwartym i piątym — przewiduje sześć kierunków specjalizacji

<sup>2)</sup> 45 wykładów, 15 godzin ćwiczeń

Centrum Informacji Naukowej, Technicznej i Ekonomicznej, Al. Niepodległości 186, 00-950 Warszawa

**ZAKUPI PILNIE DZIURKARKI TAŚMY PAPIEROWEJ OŚMIOKANAŁOWEJ W KODZIE MASZYN JEDNOLITEGO SYSTEMU.**

Zgłoszenia prosimy kierować pod wyżej wymienionym adresem lub telefonicznie: 25-61-78 lub 25-12-41 wewn. 272.

## Czekając na Napoleona

Rozwój zastosowań systemów informatycznych w naszym kraju limitują:

- sprzęt i oprogramowanie
- koordynacja rozwoju informatyki
- poziom organizacji i zarządzania w jednostkach wdrażających systemy informatyczne.

### Sprzęt i oprogramowanie

Perspektywy dotyczące wyposażenia w podstawowy sprzęt komputerowy wydają się być pocieszające. Uruchomiliśmy też produkcję urządzeń do zapisu magnetycznego na taśmie magnetycznej (MERA 9150) oraz urządzeń do zapisu na dyskach elastycznych, co pozwala na sukcesywne eliminowanie kart dziurkowanych jako podstawowego nośnika danych wejściowych. Wymagania bardziej zaawansowanych użytkowników w niedalekiej już przyszłości powinna zaspokoić rodzina drugiej generacji komputerów Jednolitego Systemu (RIAD-2). Ponadto, zgodnie z trendami światowymi, opracowano urządzenia i systemy operacyjne pozwalające na teleprzetwarzanie. Stwarza to możliwości tworzenia dużych centrów komputerowych, połączonych z licznymi urządzeniami końcowymi do zdalnego przetwarzania danych. Natomiast powszechnie odczuwa się brak minikomputerów kompatybilnych sprzętowo i programowo z dużymi maszynami oraz dostosowanych do pełnienia funkcji inteligentnych terminali. Brak odpowiednich konstrukcji minikomputerowych ograniczył również ich zastosowanie do sterowania procesami produkcyjnymi i zapewne dlatego jesteśmy tak opóźnieni w tej dziedzinie w stosunku do wysoko uprzemysłowionych krajów zachodnich.

Istotną przeszkodą w rozwoju instalacji komputerowych są trudności w uzyskaniu pomieszczeń i materiałów wyposażeniowych dla nowo powstających ośrodków obliczeniowych. Ze względu na znane kłopoty z serwisem technicznym i częściami zamiennymi problemem jest również utrzymanie w należytej sprawności technicznej już eksploatowanego sprzętu.

Generalnie można jednak powiedzieć, że aktualny stan rozwoju sprzętu oprogramowania osiągnął poziom pozwalający na dalszy rozwój zastosowań informatyki w planowaniu i zarządzaniu.

### Koordynacja rozwoju informatyki

Kolejnym czynnikiem w największym chyba stopniu wpływającym na rozwój zastosowań, projektowania i wdrażania systemów informatycznych jest ogólnokrajowa koordynacja tego rozwoju. Taka koordynacja praktycznie w ogóle nie istnieje.

Za produkcję sprzętu komputerowego oraz politykę importową i zakup licencji odpowiada Zjednoczenie „MERA”. Koordynacją działalności w zakresie oprogramowania, wdrażania i szkolenia kadr zajmuje się Ministerstwo Nauki, Szkolnictwa Wyższego i Techniki wraz z Sekretariatem Komitetu Informatyki, obsługującym Komitet Informatyki. W wielu województwach, w tym również stołecznym, przy urzędach wojewódzkich działają rady programowe informatyki, zajmujące się rozwojem i koordynacją informatyki na terenie własnych województw. Organem koordynującym jest jeszcze Zjednoczenie Informatyki, nadzorujące pracę ośrodków ZETO na terenie całego kraju.

Gdybym jednak miał ochotę np. na opracowanie na komputer ODRA 1305 jakiegoś systemu gospodarki materiałowej, to po pierwsze żadna z ww. instytucji nie musiałaby o tym wiedzieć, a gdyby nawet dowiedziała się, to nie znam trybu, w którym można by zakazać wykonywania zbędnego oprogramowania.

Tak więc nie ma żadnych informacji o projektowanych i użytkowanych w kraju systemach informatycznych, bo nie ma też skutecznego systemu ogólnokrajowej koordynacji rozwoju informatyki.

Przyczyna tego stanu rzeczy jest jedna, a mianowicie brak odpowiednich narzędzi działania w instytucjach, które powinny zajmować się tą koordynacją. Zawsze bowiem in-

formatyczne ośrodki zakładowe, branżowe, czy resortowe będą stawiały wyżej swoje partykularne interesy, ponieważ ich fundusze płac, etaty, środki inwestycyjne, pomieszczenia itp. są przydzielane przez bezpośrednich zwierzchników, którzy, powiedzmy delikatnie, nie zawsze rozumieją potrzeby informatyki w swojej branży, nie mówiąc już o spojrzeniu szerszym, a zwłaszcza w skali całego kraju.

Sprawa ta wymaga pilnego rozwiązania. Nie można się tłumaczyć brakiem koordynacji, nie można też pozwolić na niepotrzebne marnowanie i tak skromnego jeszcze potencjału naszej informatyki. Jedną z pierwszych jaskółek jest działalność wyżej wspomnianej Rady Programowej Informatyki województwa stołecznego. Ale jedna jaskółka wiosny nie czyni...

### Poziom organizacji i zarządzania

Trzecim i ostatnim czynnikiem, który obok sprzętu i systemu koordynacji wpływa na niedostateczny rozwój zastosowań systemów informatycznych, jest poziom organizacji i zarządzania w jednostkach wdrażających te systemy.

Wprowadzenie systemów informatycznych w jednostkach o złej organizacji i złym modelu zarządzania nie spowoduje oczywiście istotnej jakościowo poprawy. Informacje, które w systemie tradycyjnym nie były właściwie wykorzystywane i przekazywane, przy zastosowaniu elektronicznej techniki obliczeniowej będą tylko szybciej krążyć między poszczególnymi szczeblami organizacji. Stąd czasami słyszy się narzekania na bałagan spowodowany rzekomo wprowadzeniem systemów epd. Ale to nie komputery są winne.

Bezkrytyczne stwierdzenie, że informatyka to lekarstwo na wszystkie kłopoty, jest z gruntu fałszywe. Informatyka jest tylko narzędziem w realizacji celów. Komputer nie zapewni rytmiczności dostaw materiałów i surowców, nie stworzy prawidłowej wejściowej informacji o procesie gospodarczym. Stara to prawda, że jakie będą dane wejściowe — takich należy oczekiwać wyników.

Dlatego też twierdzę, że za bardzo wybiegliśmy naprzód we wdrożeniach systemów informatycznych, nie uwzględniając poziomu organizacji i zarządzania jednostek, w których tych wdrożeń dokonujemy. Twierdząc również, że sami informatycy nie są w stanie dokonywać prawidłowych analiz w dziedzinie organizacji i zarządzania, ani proponować odpowiednich zmian w tym zakresie. Możemy być tutaj konsultantami, ale nie bezpośrednimi wykonawcami. Dlatego postuluję o zwiększenie roli oraz o nadanie wyższej rangi komórkom organizacji i zarządzania w zakładach, branżach i resortach. Należy oddzielić w tych komórkach organizację i zarządzanie od normowania pracy. Trzeba wzmocnić obsadę etatową tych komórek, zatrudnić wysoko kwalifikowanych specjalistów organizacji i zarządzania. Doskonalenie zarządzania to proces ciągły, dlatego należy podejść do problemów w nim występujących w sposób zorganizowany i systematyczny. Takie komórki byłyby więc inicjatorami i partnerami w projektowaniu i wdrażaniu systemów epd. Bez takiej współpracy projektowanie, a przede wszystkim wdrażanie, trwa zbyt długo i często kończy się niepowodzeniem.

Co do jednego jesteśmy wszyscy zgodni: nie ma alternatywy „z komputerem czy bez komputera”. Nowoczesna gospodarka i zarządzanie nie może obejść się bez informatyki. Chociaż wiele już zostało zrobione, to jednak w dziedzinie zastosowań możemy i powinniśmy rozwijać się szybko.

Wymieniłem trzy, jak się wydaje, najważniejsze czynniki limitujące ów rozwój. Wiele na ten temat zostało już powiedziane i napisane, ale środowisko nasze czeka na konkretne i efektywne działanie.

Napoleon Bonaparte powiedział kiedyś, że nie jest zwycięzcą ten, kto proponuje jakieś rozwiązanie, ale ten kto ma odwagę wprowadzić je w życie. Czy polska informatyka doczeka się swego Napoleona?

Mgr Aleksander LESZ  
Branżowy Ośrodek Informatyki „POLLENY”

Warszawa

## Teleinformatyka na półkach księgarskich

Problemy teleinformatyki interesują nie tylko informatyków. Różne aspekty lub konsekwencje rozwoju teleinformatyki interesują także specjalistów z dziedziny telekomunikacji, matematyków, działaczy gospodarczych itp. Coraz większe jest więc zapotrzebowanie na literaturę naukową w tym przedmiocie, dostosowaną do potrzeb różnych kategorii odbiorców. Dopóki, bowiem, nie uświadomimy sobie wszyscy, czym jest i czym może stać się teleinformatyka, jakie są warunki i korzyści jej rozwoju, nie można oczekiwać szerokiej aprobaty społecznej dla realizacji krajowego programu rozwoju teleinformatyki, a praktyczne działania w tej dziedzinie będą wynikać z jednej strony z niewyważonych lub przypadkowych decyzji bądź z drugiej strony będą nacechowane piętnem informatycznej fobii.

Samo pojęcie *teleinformatyka* nie jest rozumiane jednoznacznie. Prof. Jerzy Seidler, którego stanowisko w tej sprawie podzielał, definiuje je w przybliżeniu jak następuje<sup>1)</sup>: „zarówno systemy (obliczeniowe — przyp. J.L.K.) umożliwiające zdalny dostęp, jak i nadrzędne systemy obliczeniowe powstające przez sprzęgnięcie kilku przestrzennie rozłożonych systemów obliczeniowych, nazywamy ogólnie systemami teleinformatycznymi. W tym znaczeniu teleinformatyka jest dyscypliną naukowo-techniczną dotyczącą podstaw teoretycznych i zasad budowy tak rozumianych systemów teleinformatycznych.

W niektórych kręgach specjalistów z dziedziny telekomunikacji przez *teleinformatykę* rozumie się jednak teorię i technikę zastosowania metod cyfrowych w telekomunikacji; w takim sensie *siecią teleinformatyczną* staje się zatem każda sieć łączności, w której zastosowano modulację kodowo-impulsową lub komutację elektroniczną i programowane sterowanie siecią. Pogląd powyższy bierze przypuszczalnie początek z interpretacji pojęcia *teleinformatyka* w niektórych (bynajmniej nie wszystkich) publikacjach francuskich. Tak rozumiane pojęcie *teleinformatyki*, przeniesione na grunt polski, powoduje wiele nieporozumień, polegających na przykład na traktowaniu przyszłej sieci telekomunikacyjnej kraju, opartej na komutacji elektronicznej, jako substytutu sieci transmisji danych na potrzeby systemów teleinformatycznych. Należy zatem wyraźnie rozróżniać pojęcia: systemu teleinformatycznego (STI) od sieci transmisji danych (STD), stanowiącej medium łączności STI, a oba te pojęcia — od cyfrowego systemu telekomunikacyjnego (CST), który technikę cyfrową wykorzystuje dla celów realizacji fizycznych procesów transmisji informacji (dowolnego rodzaju i przeznaczenia) i sterowania ruchem telekomunikacyjnym.

To robocze rozgraniczenie pojęć umożliwi nam orientację w sytuacji wydawniczej kilku blisko ze sobą związanych, wzajemnie komplementarnych dyscyplin naukowo-technicznych.

W dziedzinie CST na szczególne wyróżnienie zasługuje wieloletnia działalność Wydawnictw Komunikacji i Łączności, szczególnie rozwijana w ramach serii pn. „Problemy elektroniki i telekomunikacji”, adresowanej do szerokiego kręgu odbiorców o przygotowaniu technicznym. W serii tej ukazały się dotychczas trzy pozycje poświęcone CST<sup>2)</sup>, w przygotowaniu są jednak następne, szerzej omawiające problematykę STD.

W ramach tejże serii ukazała się ostatnio praca P. Ostrowskiego i M. Dąbrowskiego<sup>3)</sup>, poświęcona proble-

matyce STD z pewnymi elementami STI. Praca ta zawiera znaczny ładunek praktycznej wiedzy inżynierskiej, bardzo przydatnej zarówno specjalistom z dziedziny telekomunikacji, jak i informatyki. Zasługuje też na wyróżnienie staranna szata graficzna książek ukazujących się w serii „Problemy Elektroniki i Telekomunikacji”.

W tym samym wydawnictwie, poza serią, wydano niedawno pracę zbiorową pod redakcją doc. dr hab. Wojciecha Sobczaka, poświęconą problematyce STI<sup>4)</sup>. Publikację tę należałoby zalecić każdemu informatykowi o zainteresowaniach systemowych ze względu na przystępne, przeglądowe ujęcie tematu, oparte na licznych przykładach rozwiązań praktycznych znanych z literatury obcej, jak również odwołanie się do opracowań krajowych (głównie teoretycznych), zwłaszcza zrealizowanych w środowisku gdańskim, z którego wywodzi się grupa współautorów tej wartościowej publikacji. Powinna ona z powodzeniem zastąpić trudno już dziś dostępne (i nieaktualne w stosunku do obecnego poziomu techniki światowej) publikacje książkowe z dziedziny STI, wydane w ubiegłych latach przez Wydawnictwo MON<sup>5)</sup> lub WNT<sup>6)</sup>.

Czytelnik interesujący się zagadnieniami systemów wielodostępnych (*wielodostępowych* — jak z uporem utrzymują autorzy ze „szkoły gdańskiej”) ma do wyboru stosunkowo duży zbiór publikacji: poczynając od dawniejszych tłumaczeń, które problemy te ujmowały jeszcze marginesowo<sup>7)</sup>, i trudno dostępnych skryptów akademickich<sup>8)</sup> do dwóch danych publikacji poświęconych w całości systemom wielodostępny: tłumaczeniu pracy U. E. Fischera<sup>9)</sup> oraz oryginalnej pracy doc. dra Mieczysława Bazewicza<sup>10)</sup>, poświęconej w dużym stopniu systemom wielodostępny zrealizowanym w oparciu o sprzęt informatyczny produkcji krajowej.

Na marginesie publikacji na temat systemów wielodostępnych należałoby wspomnieć o przetłumaczeniu na język polski i wydaniu przez WNT, w ramach serii „Informatyka”, znanej pracy pod red. Philipa J. Enslowa<sup>11)</sup>, poświęconej cyfrowym systemom wieloprocesorowym. Systemów wieloprocesorowych nie można, oczywiście, utożsamiać z systemami teleinformatycznymi: relacja między tymi pojęciami jest relacją częściowego pokrywania się. System informatyczny jest systemem wieloprocesorowym (SWP) jeśli składa się z dwóch lub więcej procesorów współpracujących ze wspólną pamięcią lub urządzeniami wejścia/wyjścia i podporządkowanych wspólnemu systemowi operacyjnemu. W szczególnym przypadku może to być system teleinformatyczny. Do kategorii STI zaliczamy jednak także systemy wielodostępne z jednym procesorem, posiadające gwiazdową konfigurację połączeń terminali z jednostką przetwarzającą, jak również systemy wielomaszynowe, w których każda maszyna podporządkowana jest własnemu systemowi operacyjnemu, natomiast komunika-

<sup>1)</sup> Sobczak W. (red.): Wprowadzenie do teleinformatyki. WKiŁ, Warszawa 1979

<sup>2)</sup> Hącel E., Puchalski F., Sieradzan R.: Transmisja danych w systemach informatycznych. Wyd. MON, Warszawa 1974

<sup>3)</sup> Martin J.: Wprowadzenie do transmisji danych. WNT, seria „Informatyka”. Warszawa 1976

<sup>4)</sup> Eadie D.: Nowoczesne maszyny i systemy cyfrowe. WNT, seria „Informatyka”. Warszawa 1975

<sup>5)</sup> Martin J. T.: Programowanie maszyn cyfrowych w systemach uwarunkowanych czasowo. WNT, seria „Informatyka”. Warszawa 1970

<sup>6)</sup> Huzar Zb. (red.): Organizacja maszyn cyfrowych. Cz. II — Systemy cyfrowe. Wyd. Politechniki Wrocławskiej, Wrocław 1976

<sup>7)</sup> Fischer U. E.: Systemy abonentkie. WNT, Warszawa 1977

<sup>8)</sup> Bazewicz M.: Wielodostępne systemy informatyczne. PWN, Warszawa 1977

<sup>9)</sup> Enslow jr Ph. H. (red.): Systemy cyfrowe wieloprocesorowe. WNT, seria „Informatyka” Warszawa 1978

<sup>1)</sup> Seidler J.: Analiza i synteza sieci łączności dla systemów teleinformatycznych. PWN, seria „Biblioteka informatyki”. Warszawa 1979

<sup>2)</sup> Majewski W. (red.): Systemy sieci zintegrowanej. WKiŁ, seria „Problemy elektroniki i telekomunikacji”. Warszawa 1978

<sup>3)</sup> Majewski W., Miłek J. (red.): Teletransmisyjne systemy cyfrowe. WKiŁ, seria „Problemy elektroniki i telekomunikacji”. Warszawa 1976

<sup>4)</sup> Sochacki J. (red.): Transmisja danych. WKiŁ, seria „Problemy elektroniki i telekomunikacji”. Warszawa 1966

<sup>5)</sup> P. Ostrowski, M. Dąbrowski „Systemy i sieci teleinformatyczne” WKiŁ, seria „Problemy Elektroniki i Telekomunikacji”. Warszawa 1979.

cja międzymaszynowa odbywa się za pośrednictwem kanałów wejścia/wyjścia.

W grupie publikacji poświęconych STD na osobne omówienie zasługuje wspomniana już obszerna monografia naukowa prof. dra Jerzego Seidlera<sup>1)</sup>, wydana przez PWN w ramach serii „Biblioteka informatyki”. Książka stanowi niejako kontynuację szeregu dzieł tego autora, z których na przypomnienie zasługuje zwłaszcza praca<sup>2)</sup> poświęcona teoretycznym podstawom transmisji danych cyfrowych. Obie prace charakteryzuje podobny sposób ujęcia tematu: spojrzenie na problematykę techniczną poprzez pryzmat jej podstaw teoretycznych, popartych w miarę potrzeby ścisłym wywodem matematycznym.

W swej ostatniej pracy prof. J. Seidler obszernie omawia specyficzne wymagania sieci transmisji danych, jakie warunkuje efektywna współpraca z komputerami. Z punktu widzenia telekomunikacji komputer jest klientem dość kłopotliwym: stawia on ostre wymagania dotyczące dopuszczalnej stopy błędów, niezawodności połączeń, czasu realizacji połączenia i przejścia komunikatu (pakietu) przez sieć, przepustowości informacyjnej kanałów i nierównomierność ich obciążenia w czasie, rozproszenia terytorialnego użytkowników sieci, ich zróżnicowania pod względem priorytetu obsługi itp. Podstawy teoretyczne i zasady projektowania STD wyrastają zatem w odrębną dziedzinę wiedzy, posilkującą się dość złożonym aparatem formalnym i metodami wyboru rozwiązań spełniających postulaty praktyki.

Książka prof. J. Seidlera to cenna pozycja nie tylko na krajowym rynku wydawniczym. Omawia temat w sposób wielostronny i głęboki, nawiązuje do obszernej literatury zagranicznej, jak również eksponuje (wcale niemały, choć w praktyce nie wykorzystany) dorobek autorów polskich. Nie jest to jednak książka łatwa, przeznaczona dla tych, którzy STD traktują jako przedmiot swej specjalizacji zawodowej. Bardzo istotną zaletą publikacji jest m.in. próba powiązania parametrów technicznych STD ze wskaźnikami ekonomicznymi; wykorzystanie metod rachunku optymalizacyjnego wydaje się jedyną rozsądną koncepcją prowadzenia dyskusji na temat „opłacalności” lub „nieopłacalności” systemów teleinformatycznych na płaszczyźnie wolnej od subiektywizmu. Autor książki nie sugeruje zresztą, że dysponuje pewną i jednoznaczną metodą rozstrzygnięcia wszystkich pytań, jakie mogą się nasunąć w związku z re-

alizacją STD lub ze sterowaniem jej pracą; proponuje raczej sposób tworzenia takich metod, oparty na naukowych przesłankach. Z tego względu książkę prof. Jerzego Seidlera należałoby polecić w pierwszym rzędzie tym, którzy poszukują nie tyle gotowych recept, co inspiracji do twórczego działania w dziedzinie transmisji danych i systemów teleinformatycznych.

Nakładem Wydawnictw Naukowo-technicznych w ramach serii „Informatyka” wydano także pracę zbiorową pod red. N. Abramsona i F. F. Kuo<sup>3)</sup>, poświęconą problemom STI z bogatym omówieniem doświadczeń amerykańskich. Oprócz znanych polskim czytelnikom sieci ARPA lub ALOHA na temat koncepcji systemów teleinformatycznych firmy IBM oraz sieci uczelnianej Dartmouth College. Praca napisana jest przystępnym językiem inżynierskim i stosunkowo dobrze przetłumaczona (w odróżnieniu od niektórych innych publikacji tej serii, na przykład<sup>4)</sup>).

Nieco inny charakter ma publikacja opracowana pod redakcją doc. dra Mieczysława Bazewicza<sup>5)</sup>, wydana nakładem Wydawnictwa Politechniki Wrocławskiej, pierwsza z zamierzonej serii publikacji o charakterze szkoleniowym, przeznaczonych dla projektantów STI. Część I, poświęcona protokołom komunikacyjnym, zawiera materiały Szkoły Zimowej zorganizowanej w 1978 r. staraniem Politechniki Wrocławskiej i Instytutu Podstaw Informatyki PAN, przy współudziale specjalistów z CSRS. Jest to pierwsza publikacja w języku polskim poświęcona temu zagadnieniu, które zresztą nie doczekało się dotychczas właściwego (tj. kompletnego i usystematyzowanego) opracowania w literaturze światowej.

Rozwój piśmiennictwa dotyczącego teleinformatyki stwarza coraz więcej problemów terminologicznych. Próby uporządkowania słownictwa teleinformatycznego, podjęte w Instytucie Podstaw Informatyki PAN<sup>6)</sup>, noszą, niestety, zbyt liczne ślady niedopracowania merytorycznego i redakcyjnego. Warto natomiast zasygnalizować, że we wspomnianym wyżej instytucji prowadzony jest rejestr bibliograficzny krajowych publikacji z dziedziny teleinformatyki, który może być cenną pomocą dla każdego, kto ma trudności z dotarciem do odpowiedniej literatury.

<sup>1)</sup> N. Abramson, F. F. Kuo (red.): Sieci telekomunikacyjne komputerów. WNT, seria „Informatyka”. Warszawa 1978

<sup>2)</sup> Bazewicz M. (red.): Własności i funkcje sieci komputerowych. Cz. I — Protokoły. Biblioteka WASC. Politechnika Wroclawska, Wroclaw 1979

<sup>3)</sup> Rupiński M.: Sieci komputerowe. Słownictwo. Prace IPI PAN, z. 350, 1979

Juliusz Lech KULIKOWSKI

<sup>4)</sup> Seidler J.: Systemy przesyłania informacji cyfrowych. WNT, Warszawa 1972

## Warunki prenumeraty

Prenumeratę przyjmują oddziały RSW „Prasa-Książka-Ruch” i urzędy pocztowe.

Jednostki gospodarki społecznej, instytucje, organizacje i wszelkiego rodzaju zakłady pracy zamawiają prenumeratę w miejscowych oddziałach RSW „Prasa-Książka-Ruch”, a w miejscowościach, w których nie ma oddziałów — w urzędach pocztowych. Czytelnicy indywidualni opłacają prenumeratę wyłącznie w urzędach pocztowych i u doręczycieli.

Cena prenumeraty krajowej wynosi:

- kwartalna — 75 zł
- półroczna — 150 zł
- roczna — 300 zł

Przedpłaty przyjmowane są w następujących terminach:

- do 10 czerwca — na III kwartał i II półrocze
- do 10 września — na IV kwartał

- do 25 listopada — na rok następny, I kwartał, I półrocze
- do 10 marca — na II kwartał

Prenumeratę ze zleceniem wysyłki za granicę przyjmuje RSW „Prasa-Książka-Ruch”, Centrala Kolportażu Prasy i Wydawnictw, ul. Towarowa 28, 00-958 Warszawa, konto PKO nr 1531-71 — w terminach obowiązujących dla prenumeraty krajowej.

Prenumerata ze zleceniem za granicę jest droższa od prenumeraty krajowej o 50% dla zleceniobiorców indywidualnych i o 100% dla zlecających instytucji i zakładów pracy.

Exemplarze archiwalne czasopism wydawanych przez WCT NOT można nabyć w Dziale Handlowym przy ul. Mazowieckiej 12, 00-048 Warszawa, tel. 26 80 16.

**BOLESŁAW SZYMAŃSKI**

Instytut Informacji Naukowej, Technicznej i Ekonomicznej  
Warszawa

## Główne kierunki rozwoju zastosowań

Relacyjny model baz danych został po raz pierwszy zaprezentowany przez E. F. Codd'a [3] w 1970 roku. Definicja tego modelu, elegancka matematycznie i prosta pod względem ideowym, wykorzystywała terminologię matematyczną, różną od powszechnie przyjętej terminologii baz danych w ujęciu wcześniejszych modeli, takich jak hierarchiczny czy sieciowy [17]. Z tego powodu propozycja Codd'a wywołała wiele kontrowersji i dyskusji, co doprowadziło do uściślenia i rozwinięcia relacyjnego modelu [4, 7].

Bezsporną zaletą modelu relacyjnego jest wygodna dla użytkownika postać opisu bazy danych i dostępu do niej. Ta zaleta zadecydowała, że od 1974 r. zaczęły pojawiać się systemy wykorzystujące w mniejszym lub większym stopniu relacyjny model baz danych [6, 11, 13, 15].

Prezentowany artykuł stanowi próbę podsumowania doświadczeń wynikających z wykorzystania relacyjnego modelu przy realizacji baz danych. Na tej podstawie określa się również przewidywane główne kierunki rozwoju zastosowań relacyjnego modelu baz danych. Autor miał okazję do zapoznania się z omawianą problematyką podczas stażu, jaki odbył w Wielkiej Brytanii w końcu 1978 r.

### PODSTAWOWE DEFINICJE

Relacją  $n$ -tego rzędu  $R(D_1, D_2, \dots, D_n)$ , określoną na zbiorach  $D_1, D_2, \dots, D_n$  (niekoniecznie różnych), nazywa się zbiór  $n$ -tek uporządkowanych  $d_1, d_2, \dots, d_n$ , takich, że  $i$ -ty element  $n$ -tki  $d_i$  należy do zbioru  $D_i$ . Zbiory  $D_1, \dots, D_n$  nazywane są dziedzinami relacji.

Zbiór wszystkich  $i$ -tych elementów  $n$ -tek relacji  $R$ , nazywany atrybutem tej relacji, powinien mieć unikalną w relacji nazwę. Przyjmuje się, że dwie relacje różniące się tylko kolejnością atrybutów są identyczne.

Bazę danych stanowi zbiór zmiennych w czasie (poprzez działanie na bazie danych) relacji.

$N$ -tki identyfikowane są przez swoje wartości. Atrybuty lub ich minimalne kombinacje pozwalające identyfikować wystąpienia  $n$ -tki nazywane są potencjalnymi kluczami. Atrybuty wchodzące w skład potencjalnego klucza nazywa się atrybutami podstawowymi. Spośród kluczy potencjalnych wybiera się arbitralnie klucz relacji.

Należy jeszcze podkreślić, że określenie atrybutu podstawowego, klucza, klucza potencjalnego czy zależności musi się odbywać na podstawie własności opisywanych baz danych obiektów, a nie aktualnych wartości atrybutów relacji.

Relację  $n$ -tego rzędu wygodnie jest reprezentować w postaci zwykłej tablicy o  $n$  kolumnach i zmiennej w czasie liczbie wierszy. Należy tylko pamiętać, że kolejność wierszy i kolumn jest tu nieistotna.

Jako przykład rozpatrywana będzie prosta baza danych o bibliotece. Składa się ona z dwóch relacji: ZBIOR i WYP. Określone są 3 dziedziny (domeny): NAZWA, DATA, SYGNATURA. Relacja ZBIOR definiuje zasoby biblioteki, określając tytuł i autora każdej książki w bibliotece oraz jej sygnaturę; posiada więc 3 atrybuty o następujących nazwach: TYTUŁ (dziedzina NAZWA), AUTOR (dziedzina NAZWA) i SYG (dziedzina SYGNATURA). Na tym przykładzie widać różnicę między atrybutem, który ma określone znaczenie w relacji, a dziedziną, która jest zbiorem informacji jednego typu. Relacja WYP opisuje wypożyczenie z biblioteki, określając sygnaturę wypożyczonej książki, nazwisko wypożyczającego czytelnika i datę wypożyczenia.

Obie relacje przedstawione są poniżej.

Relacja ZBIOR

| SYGNATURA | NAZWA         | NAZWA       |
|-----------|---------------|-------------|
| SYG       | AUTOR         | TYTUŁ       |
| 000120    | A. MICKIEWICZ | PAN TADEUSZ |
| 000121    | A. MICKIEWICZ | DZIADY      |
| 000122    | A. MICKIEWICZ | DZIADY      |

dziedzina  
atrybut

Relacja WYP

| SYGNATURA | NAZWA       | DATA     |
|-----------|-------------|----------|
| SYG       | CZYT        | DATA     |
| 00069     | ADAMSKI A.  | 30/11/78 |
| 00120     | KOWALSKI B. | 30/01/79 |
| 00121     | KOWALSKI B. | 28/01/79 |

dziedzina  
atrybut

Relacyjny model pozwala na definiowanie języków operowania na bazie danych, zapewniających niezależność programów realizujących operacje na bazie danych od fizycznej reprezentacji danych [5]. Codd w [7] określa 5 rodzajów tego typu języków:

1. Języki dostępu „element po elemencie”  
Języki te stanowią najniższy proceduralny interfejs do relacyjnej bazy danych. Przeznaczone są w zasadzie do budowania na nich interpreterów języków wyższego rzędu. Zasadą jest, że w dowolnym momencie użytkownik ma dostęp do jednego elementu bazy danych.
2. Algebra relacyjna  
Wyszukiwanie jest realizowane poprzez tworzenie nowych relacji, które zawierają dane interesujące użytkownika. Przetwarzanie dotyczy całych relacji, dzięki czemu zapytania mogą być formułowane zwięźle i czytelnie.
3. Języki odwzorowań zbiorów  
Zapytania w tego typu językach są formułowane poprzez wybranie zbioru wartości określonego atrybutu spełniającego zadane warunki i dostarczenie zbioru wartości innych atrybutów z wybranych  $n$ -tek (wierszy). Przetwarzanie dotyczy więc całych atrybutów (kolumn) [2].
4. Rachunek relacyjny  
Języki tego typu umożliwiają wprowadzenie zdefiniowanych przez użytkownika zmiennych, których zakresem są relacje. Formułowania zapytań dokonuje się w postaci wyrażań logicznych z kwantyfikatorami. Przetwarzanie dotyczy tutaj całych  $n$ -tek (wierszy).
5. Języki pozwalające na formułowanie zapytań w języku naturalnym

W trakcie konwersacji z systemem w języku naturalnym (angielskim [6]) system zbiera wszystkie niezbędne dane do sformułowania poprawnego zapytania. Bardzo interesujące — choć trudne do zrealizowania — rozwiązanie, szczególnie atrakcyjne dla użytkowników bez znajomości programowania.

Należy podkreślić, że największą popularność zdobyły sobie algebra relacyjna i rachunek relacyjny.

# relacyjnego modelu baz danych

## IMPLEMENTACJE RELACYJNEGO MODELU BAZ DANYCH

Z punktu widzenia zastosowań jedną z najbardziej udanych implementacji relacyjnego modelu jest PRTV — *Peterlee Relational Test Vehicle* [16].

PRTV jest konwersacyjnym systemem realizującym relacyjny model bazy danych. Językiem użytkownika jest ISBL (*Information System Base Language*), który jest językiem typu algebra relacyjna. Relacje są tu traktowane jak nazwane zmienne. Do tworzenia nowych relacji z istniejących służy instrukcja podstawienia, oznaczana znakiem  $=$ , oraz sześć głównych operacji algebry relacyjnej. Są nimi: selekcja (oznaczana przez:  $\%$ ), projekcja ( $\%$ ), suma ( $+$ ), iloczyn ( $\cdot$ ) (różnica ( $-$ ) oraz połączenie ( $*$ ).

**Selekcja** ( $\cdot$ ) tworzy z jednej relacji relację wynikową, zawierającą  $n$ -tki starej relacji, spełniające pewne warunki nazywane filtrami. Filtry mogą zawierać porównania pomiędzy wartościami dwóch atrybutów lub pomiędzy atrybutem a stałą. Porównania w filtrze mogą być łączone w wyrażenia boolowskie za pomocą operatorów logicznych  $\wedge$  (and),  $\vee$  (or) oraz  $\neg$  (not) i nawiasów okrągłych. Porównania wartości numerycznych mogą wykorzystywać operatory:  $=$ ,  $\neq$ ,  $<$ ,  $\leq$ ,  $>$ ,  $\geq$ ; porównanie wartości napisowych tylko operatory  $=$  oraz  $\neq$ .

Np. wyrażenie  
ZBIOR : SYG > 100

definiuje relację zawierającą  $n$ -tki dla książek o numerze sygnatury większym niż 100, natomiast wyrażenie  
WYP : (DATA = 010279 | DATA = 010379)  $\wedge$  SYG < 100  
definiuje relację zawierającą  $n$ -tki dla osób, które wypożyczyły jakąś książkę o sygnaturze < 100 dnia 1 lutego 1979 lub 1 marca 1979.

**Projekcja** ( $\%$ ) również tworzy z jednej relacji relację wynikową. W tym przypadku do relacji wynikowej wybierane są atrybuty i ewentualnie ich nowe nazwy poprzez określenie listy zawierającej wybierane atrybuty i ich nowe nazwy. Jeżeli jedynym rezultatem projekcji ma być zmiana nazw pewnych atrybutów, to zamiast wypisywania pozostałych atrybutów relacji wystarczy użyć znaku wielokropka. Np. wyrażenie

ZBIOR % AUTOR  $\rightarrow$  PISARZ

definiuje relację 1. rzędu z jedynym atrybutem: PISARZ, natomiast wyrażenie

ZBIOR % AUTOR  $\rightarrow$  PISARZ...

definiuje relację identyczną z relacją ZBIOR tylko z nową nazwą PISARZ atrybutu AUTOR.

Należy pamiętać, że projekcja, w wyniku której usunięty zostanie atrybut podstawowy, może spowodować zmniejszenie liczby  $n$ -tek należących do relacji wynikowej, gdy w relacji nie mogą istnieć dwie identyczne  $n$ -tki. Np. w wyniku projekcji

ZBIOR % AUTOR

w relacji wynikowej trzy książki Mickiewicza reprezentowane będą przez jeden wiersz (por. rys. 1).

Operacje sumy ( $+$ ), iloczynu ( $\cdot$ ) i różnicy ( $-$ ) działają na relacjach jak na zbiorach  $n$ -tek. Obie relacje występujące w operacjach sumy i iloczynu muszą być identyczne. Suma daje w wyniku działania relację zawierającą wszystkie  $n$ -tki, które należą do jednej z relacji stanowiących operandy. Np. po zdefiniowaniu relacji NABYTEK (SYG, AUTOR, TYTUŁ), zawierającej nabytki do biblioteki, aktualizację relacji ZBIOR można otrzymać wykonując działanie:

ZBIOR = ZBIOR + NABYTEK

Iloczyn produkuje relację wynikową, zawierającą wszystkie  $n$ -tki, które należą jednocześnie do obu relacji stanowiących operandy tego działania. Tak więc wyrażenie (NABYTEK % AUTOR, TYTUŁ)  $\cdot$  (ZBIOR % AUTOR, TYTUŁ) określa książki, które zostały nabyte, a ich egzemplarze są już w bibliotece. (Warto zauważyć, że wyrażenie NABYTEK, ZBIOR daje zawsze relację pustą, bo zakłada się, że sygnatura jest unikalna dla każdego egzemplarza książki w bibliotece).

W przypadku operacji różnicy ( $-$ ) relacje stanowiące jej operandy i nazywane odpowiednio odjemną i odjemnikiem nie muszą być identyczne. Relacja wynikowa operacji odejmowania zawiera te  $n$ -tki odjemnej, dla których nie istnieje w odjemniku  $n$ -tki posiadające dla wspólnych dla odjemnej i odjemnika atrybutów tych samych wartości co  $n$ -tki należące do relacji wynikowej. Np. wyrażenie ZBIOR — WYP

określa różnicę na atrybucie SYG, a relacja wynikowa będzie zawierać  $n$ -tki dotyczące książek nie będących aktualnie w wypożyczeniu.

Operacja połączenia ( $*$ ) również wykorzystuje wspólne atrybuty obu operandów. W trakcie jej działania tworzy się wszystkie możliwe pary  $n$ -tek o tej własności, że pierwsza pochodzi z pierwszego operandu, druga z drugiego a wspólne atrybuty mają w obu  $n$ -tkach te same wartości. Z każdej takiej pary tworzy się nową  $n$ -tkę zawierającą po jednej wartości każdego atrybutu obu  $n$ -tek z pary. Jeżeli relacje będące operandami nie mają wspólnych atrybutów, tworzy się wszystkie możliwe pary  $n$ -tek obu relacji.

Np. wyrażenie

ZBIOR \* WYP

stanowi połączenie na wspólnym atrybucie SYG i tworzy relację 5 rzędu, zawierającą informacje o wypożyczonych książkach (tzn. sygnaturę, autora i tytuł książki, nazwisko czytelnika i datę wypożyczenia).

Oprócz omówionych wyżej 6 podstawowych operacji, ISBL — jak i wszystkie tego typu języki — zawiera operacje wprowadzenia i wyprowadzenia relacji do pamięci operacyjnej maszyny cyfrowej. Ponadto często operacja projekcji jest rozszerzona o operację grupowania, która polega na tworzeniu w czasie projekcji dodatkowej kolumny, zawierającej informacje o tych  $n$ -tkach, które w wyniku projekcji zostaną połączone w jedną  $n$ -tkę. Np. wyrażenie WYP % CZYT (LICZ — KSIAZEK = SUM(SYG))

definiuje relację o atrybutach CZYT i LICZ — KSIAZEK. Ten drugi atrybut określa liczbę książek wypożyczonych przez czytelnika (sumowana jest liczba wystąpień  $n$ -tek w WYP z tą samą wartością atrybutu CZYT, czyli nazwiskiem czytelnika).

Często do algebry relacyjnej wprowadza się możliwość definiowania przez użytkownika procedur w języku programowania wyższego rzędu — w przypadku ISBL jest to PL/I.

PRTV znalazł wiele zastosowań, z których najważniejsze to systemy:

- planowania rozwoju miasta dla Londynu [12]
- informacyjny dla Światowej Organizacji Zdrowia oraz
- informacyjny dla badań w dziedzinie hodowli zbóż [9].

We wszystkich tych zastosowaniach utworzone bazy danych zawierały wysoki procent danych numerycznych, a odpowiedzi stanowiły na ogół duże grupy danych o postaci tabelarycznej. Ponadto bazy te nie wymagały ciągłej aktualizacji danych. W tego typu zastosowaniach użycie PRTV zakończyło się pełnym sukcesem.

Ze względu na prostotę ideową relacyjny model baz danych jest szczególnie atrakcyjny przy realizacji minikomputerowych baz danych. Jedną z najpełniejszych tego typu implementacji jest INGRES [13]. Do formułowania zapytań w systemie tym wykorzystuje się QUEL, język typu rachunku relacyjnego. W języku tym używa się zmiennych, których wartościami są  $n$ -tki z relacji określonych dla każdej zmiennej przez instrukcję RANGE.

Postać tej instrukcji jest następująca: RANGE OF lista zmiennych IS nazwa relacji. Właściwe działania na bazie danych wykonywane są przy pomocy instrukcji o następującej budowie:

Instrukcja Nazwa (lista podstawień atrybutów)

WHERE Warunek

Możliwe jest użycie następujących instrukcji: RETRIVE INTO, APPEND TO, REPLACE OF, DELETE. Nazwa jest albo nazwą relacji (dla instrukcji RETRIVE INTO i APPEND TO) albo nazwą zmiennej (dla pozostałych instrukcji). Lista podstawień atrybutów zawiera po lewej stronie nazwę atrybutu wynikowej relacji instrukcji, a po prawej wyrażenie arytmetyczne lub napisowe, złożone ze stałych i zmiennych kwalifikowanych nazwami atrybutów. Podobnie warunek jest wyrażeniem boolowskim, w którym występuje porównanie między wartościami atrybutów pewnej relacji a wyrażeniami arytmetycznymi lub napisowymi.

Instrukcja RETRIVE INTO tworzy nową relację, której atrybuty otrzymują wartości zgodne z wartościami z listy podstawień atrybutów, wyliczanych tylko dla takich  $n$ -tek, których atrybuty spełniają warunki określone po WHERE. Np. po wykonaniu instrukcji

```
RANGE OF WIS WYP
RANGE OF ZIS ZBIOR
RETRIVE INTO SYG-100 (TYTUL = Z. TYTUL, AUTOR = Z. AUTOR)
```

WERE Z. SYG > 100

```
RETRIVE INTO CZYT-100 (CZYT = W. CZYT)
WHERE (W. DATA=010279|W. DATA=010379)ε W. SYG < 100
```

```
RANGE ON NIS NABYTEK
RETRIVE INTO DUPLIKAT (PISARZ = N. AUTOR, TYTUL = N. TYTUL)
```

WHERE N. AUTOR = Z. AUTOR ε N. TYTUL = Z. TYTUL

relacja SYG-100 będzie zawierać dane o książkach, o sygnaturze większej niż 100, CZYT-100 — o czytelnikach, którzy wypożyczyli 1 marca 1979 lub 1 lutego 1979 roku książki o sygnaturze mniejszej niż 100, a DUPLIKAT — o nowo nabytych książkach, których egzemplarze biblioteka już posiada.

Instrukcja APPEND TO pozwala na dołączenie do istniejącej relacji nowych  $n$ -tek. Np. aktualizacji relacji ZBIOR można dokonać przy pomocy instrukcji

```
RANGE OF N IS NABYTEK
APPEND TO ZBIOR (SYG = N. SYG, AUTOR = N. AUTOR, TYTUL = N. TYTUL)
```

Instrukcja DELETE pozwala na skreślenie z relacji pewnych  $n$ -tek. Np. aby w relacji ZBIOR uzyskać dane tylko książek nie wypożyczonych można napisać

```
RANGE OF W IS WYP
RANGE OF Z IS ZBIOR
DELETE Z WHERE Z. SYG = W. SYG.
```

Instrukcja REPLACE OF pozwala na aktualizację wartości atrybutów relacji.

Poza tym QUEL zawiera instrukcje tworzenia i niszczenia relacji i całej bazy danych, kopiowania i wyprowadzania relacji i definiowania metod dostępu do relacji.

INGRES jest dostępny na wielu instalacjach PDP 11/40, ale ze względu na swoją wielkość i interakcyjny charakter pracy ma zastosowania głównie dydaktyczne. Jest on używany czasami do zaspokajania niekomercyjnych potrzeb pojedynczych ludzi (np. prowadzenie ewidencji studentów i wykładów, na które uczęszczają, na potrzeby wykładowców czy opiekunów studentów).

## WIELOMODELOWE BAZY DANYCH

Aby użytkownik mógł traktować bazę danych innego typu jako relacyjną tworzone są relacyjne interfejsy do hierarchicznych [10] lub sieciowych [1] baz danych. Daje to tę korzyść, że zapytanie w algebrze relacyjnej jest proste i krótkie i może je przygotować niespecjalista.

Przykładowym rozwiązaniem jest projekt ASTRID [1], w którym na podstawie zapytań w postaci wyrażeń algebry relacyjnej przeszukuje się codasylową bazę danych. Najistotniejszą trudnością jest tu przedstawienie Codasylowej bazy w postaci zbioru relacji, co wykazuje Administrator Bazy Danych. Natomiast zapytania w algebrze relacyjnej są przygotowywane i weryfikowane pod względem składniowym w trybie konwersacyjnym, a następnie uruchamiane na symulatorze bazy danych, by zweryfikować semantyczną poprawność zapytania. Ponieważ można wyświetlać wszystkie wyniki pośrednie, wyszukiwanie ewentualnych błędów jest łatwe. Znaczenie kontroli semantycznej wynika ze szczególnych możliwości algebry relacyjnej. Algebra relacyjna pozwala łatwo formułować zapytania o wysokiej złożoności. Prowadzą one do bardzo długich wyszukiwań i ewentualny błąd semantyczny w zapytaniu byłby bardzo kosztowny.

Żepełnie unikalne rozwiązanie realizowane jest w ramach projektu PRECI — *Prototype RELational Codasyl Interface* [8].

Zadaniem projektu jest stworzenie bazy danych typu relacyjnego z możliwością dostępu do danych poprzez codasylowe instrukcje. System będzie zapewniał dużą niezależność i mobilność danych ze względu na użycie 3-pozomej architektury: globalnego schematu (SCHEMA), schematu pamięci (STORAGE SCHEMA) i schematu użytkownika (SUBSCHEMA), które mogą być codasylowe. Językiem bazowym jest PL/I z DML w postaci pewnego podzbioru codasylowego DML oraz w postaci algebry relacyjnej [14]. Dla ułatwienia aktualizacji danych przy zmianie klucza klucze relacji i tzw. obce klucze (tzn. klucze jednej relacji występujące jako atrybuty drugiej) są zastępowane surogatami. Surogaty w sposób unikalny identyfikują  $n$ -tkę w całej bazie danych (a więc również jej miejsca w pamięci) i są tworzone z kluczy przy pomocy hash-kodowania. Jak wynika z analizy przeprowadzonej w [8] takie rozwiązanie umożliwia bezpośrednie reprezentowanie wszystkich ważniejszych struktur codasylowych, a pośrednie również grup powtarzalnych. System ma na razie charakter eksperymentalny i służy raczej do dalszego rozwijania modelu relacyjnego niż bezpośrednich zastosowań praktycznych.

Wieloletnie doświadczenia ze stosowania komputerowych baz danych wykazały, że główną barierą przy rozwoju zastosowań jest komunikacja z użytkownikiem. Typowym przykładem może być rozwój zastosowań baz danych na Uniwersytecie w Aberdeen, gdzie dostępny jest firmowy system IDS-II, o ideologii codasylowej, na maszynie HONEYWELL 66. Na niewielu wydziałach, na których naukowcy nauczyli się dodatkowo programowania, istnieje wiele ciekawych zastosowań. Np. wydział medycyny wykorzystuje kilka baz danych, m.in. anesteziologiczną — o efektach różnych intensywnych zabiegów, medyczo-socjologiczną — o historii zdrowia wielu pokoleń ludzi w Aberdeen, ginekologiczną — o wartości progностycznej badań kobiet w ciąży, urologiczną — o kolejce osób do sztucznej nerki, czy terapeutyczną — o pacjentach przyjmowanych do szpitala (głównie historia choroby). Podobnie wydział rolniczy wykorzystuje bazę danych o pokoleniach owiec dla optymalnego doboru par, gleboznawczą dla badań zmian jakości gruntów w Szkocji itd. Z drugiej strony wiele wydziałów w ogóle nie prowadzi żadnych baz danych nie mogąc pokonać bariery dostępu do gromadzonych danych.



Wydaje się, że dla użytkowników baz danych, którzy chcą te bazy tylko przeszukiwać, model relacyjny jest bardzo przydatny. Np. porównanie programów działających na głębokości bazy danych pokazało, że wykorzystanie systemu ASTRID dawało skrócenie programu użytkownika kilkadziesiąt razy w porównaniu z programem dla systemu IDS-II. Jednak dla prowadzenia bazy danych oraz efektywnej jej organizacji model relacyjny jest zbyt ogólny. Można więc stwierdzić, że będzie się nasilał trend do tworzenia interfejsów relacyjnych do baz danych innego typu. Interfejsy takie będą szczególnie przydatne wtedy, gdy na danych zgromadzonych w bazie danych trzeba wykonywać obliczenia, badania statystyczne, czy tworzyć zestawienia zbiorcze. Podobne wymagania są typowe dla systemów wspomagających zarządzanie, szczególnie na wyższych jego szczeblach. Stąd można wnioskować, że model relacyjny będzie intensywnie wykorzystywany przy tworzeniu „części użytkowej” konwersacyjnych systemów planowania i zarządzania oraz systemów analizy dużej liczby obserwacji i doświadczeń w badaniach naukowych. Zdaniem autora kierunek ten będzie dominował w najbliższych latach w zakresie zastosowań relacyjnego modelu baz danych.

## LITERATURA:

- [1] Bell R., Gray P. M. D.: Astrid user guide. Report AUCS/78.03, Computing Science Department, University of Aberdeen, September 1978
- [2] Boyce R. F., Chamberlin D. D., King W. F., Hammer M. M.: Specyfing Queries as Relational Expressions. „Data Base Management” pod redakcją J. W. Klimble, K. L. Koffeman, North Holland PC, 1974
- [3] Codde E. F.: A Relational Model of Data for Large Shared Data Banks. Communication of the ACM, vol. 13, nr 6, 1970
- [4] Codde E. F.: Further Normalization of the Data Base Relational Model. „Data Base Systems” pod red. R. Rustin, Prentice Hall Inc, 1972
- [5] Codde E. F.: Relational Completeness of Data Base Sublanguages. „Data Base Systems” pod red. R. Rustin, Prentice Hall Inc, 1972
- [6] Codde E. F.: Seven Steps to Rendez-vous with the Casual User. „Data Base Management” pod red. J. W. Klimble, K. L. Koffeman, North Holland PC, 1974
- [7] Codde E. F.: Recent investigation in relational data base systems. „Information Processing 74” pod red. J. F. Rosenfeld, North Holland PC, 1974
- [8] Deen S. M.: Codasyl interface to a relational system. Report AUCS/78.02, Computing Science Department, University of Aberdeen 1978
- [9] Huyshe S. R., Storey R. A., Aldred B. K.: Information Systems for Agricultural Crop Research. Internal Report, IBM UK Scientific Centre, Peterlee, November 1978
- [10] Martin G. U., Sanders A. J.: A System for Relational Access to IMS database. Internal Report, IBM UK Scientific Centre, Peterlee September 1977
- [11] Meldman M. J., Mcleod D. J., Pelicore R. J., Squire M.: RISS: A Relational Data Base Management System for Minicomputers. Van Nostrand Reinhold Co, New York, 1978
- [12] Smedley B. S.: An Urban Management System and Geographic Data Processing in Urban Planning. Report UKCS 0079, IBM UK Scientific Centre, Peterlee, April 1976
- [13] Stonebraker M., Wong E., Kreps P.: The Design and Implementation of INGRES. ACM Transactions on Data Base Systems, vol. 1, nr 3, 1976
- [14] Szymański B.: The Grammar of PRECT Relational Algebra. Report AUCS/78.04, Computing Science Department, University of Aberdeen, November 1978
- [15] Todd S. J. P.: The Peterlee Relational Test Vehicle — a system overview. IBM System Journal, vol 15, nr 4, 1976
- [16] Todd S. J. P.: Relational Database Research at the IBM United Kingdom Scientific Centre, Peterlee, A survey 1970—1977. Report UKSC-0093, IBM UK Scientific Centre, Peterlee, December 1977
- [17] Vetter M.: Principles of Data Base Systems: „International Computing Symposium 1977” pod red. E. Morlet, D. Ribbens, North Holland PC, 1977

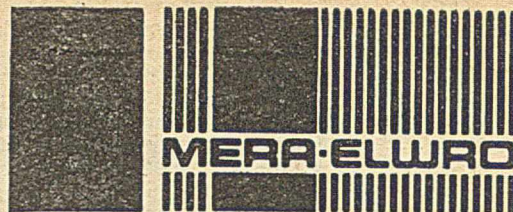
## Systemy minikomputerowe

Uniwersytetu Gdańskiego  
Instytut Cybernetyki Ekonomicznej i Informatyki  
(81-824 Sopot, ul. Armii Czerwonej 101 tel 51-00-61 w. 406)  
proponuje:  
komputerowy system obliczeń statystycznych

Autor: mgr inż. Andrzej Kwapisz  
Sprzęt: minikomputer MERA 305 w konfiguracji podstawowej, uzupełnionej jednostką pamięci dyskowej MERA 9425

Język programowania: FORTRAN IV — wersja 305  
Funkcje systemu: wykonywanie następujących obliczeń statystycznych

- zależność dwóch cech (test  $X^2$ )
- test istotności Z i T
- test jednorodności wiariancji — test Bartleeta
- ocena widma pojedynczego metoda Parzena
- aproksymacja rozkładu logarytmiczno-normalnego
- aproksymacja rozkładu normalnego
- aproksymacja krzywą Pearsona pierwszego typu
- analiza wariancji z wyborem grup jednorodnych
- weryfikacja hipotezy o różnicy pomiędzy dwiema średnimi
- obliczanie średniej, odchylenia standardowego, wariancji, współczynnika zmienności, asymetrii i koncentracji
- statystyka zgodności w metodzie ekspertyzy grupowej
- obliczanie współczynnika korelacji Pearsona
- obliczanie współczynnika korelacji nieliniowych
- obliczanie współczynnika korelacji dla tablicy
- porządkowanie ciągu liczb w kolejności od min. do max. i odwrotnie
- opracowanie wyników ankiet
- obliczanie całki pojedynczej i podwójnej
- analiza Monte-Carlo
- wyznaczanie równań regresji z wyborem funkcji najlepiej aproksymującej (liniowa, potęgowa, wykładnicza, logarytmiczna, hiperboliczna i kwadratowa)
- trend liniowy i krzywoliniowy
- regresja liniowa dla jednej, dwóch, trzech, czterech i pięciu zmiennych niezależnych z testami uzupełniającymi
- Większość z wymienionych programów działa na zbiorach dyskowych. Do założenia tych zbiorów służy program ZAP, który pozwala założyć zbiór zawierający kilka tysięcy liczb. Programy użytkowe systemu wybierają z takiego zbioru zestawy danych wg podanego z klawiatury gądz czytnika klucza (numeru zestawu). Metoda ta pozwala na wszechstronne przetworzenie zbioru danych bez potrzeby kilkakrotnego wczytywania taśmy z danymi.
- Poza tym istnieje możliwość:
  - wydrukowania zbioru danych w formie bardzo czytelnej dla użytkownika
  - dokonania korekty całych zestawów danych lub pojedynczych liczb w zbiorze
  - zapisu poprawionego zbioru na taśmie papierowej w kodzie ISO-7 (dla stworzenia nośnika archiwalnego), bądź w kodzie BCD np. do dalszego przetwarzania na komputerze ODRA.
- Dzięki umieszczeniu całego pakietu programów w bibliotece programów źródłowych na dysku istnieje bardzo łatwy dostęp do każdego z programów.
- Zastosowanie optymalnych algorytmów oraz pamięci dyskowej umożliwia szybkie opracowanie nawet bardzo dużych zbiorów danych.



## Symulator pracy jednostki centralnej Odra-1305 dla pamięci operacyjnych

W MERA-ELWRO skonstruowany został symulator pracy jednostki centralnej dla modułów pamięci operacyjnej sterowanych koordynatorem KPF-1. Symulator umożliwia autonomiczną pracę jednego z modułów przy równoczesnej niezależnej pracy procesora na pozostałej pamięci. Dysponuje on następującymi funkcjami:

- pisz „0” lub „1” ze sprawdzeniem poprawnej pracy rejestru informacji na pakietach FHX
- czytaj
- pisz „0” lub „1”, a potem czytaj ze sprawdzeniem
- trudny wzór
- test adresacji
- praca na stałym adresie
- wprowadzenie dowolnego adresu
- praca na jednym z wybranych bloków 16 K
- praca przy zablokowanej kontroli błędów
- praca z prędkością maksymalną lub asynchroniczną

Układ indykacji na 25 diodach elektro-luminescencyjnych umożliwia ciągłą obserwację informacji WE-WY, adresu oraz przerzutnika błędów.

W realizacji technicznej symulator wykonany został na trzech typowych pakietach 4-warstwowych, na których zamontowano jego strukturę logiczną, elementy manipulacyjne i indykacyjne. Do kompletu należy jeszcze zwora realizująca połączenia międzypakietowe. Całość współpracuje z pamięcią poprzez kanał drugiego procesora.

Decyzja o podjęciu produkcji zależy od potrzeb.

Zgłoszenia należy kierować do Biura Generalnych Dostaw MERA-ELWRO, ul. Ostrowskiego 32, 53-238 Wrocław.

## Nowi użytkownicy komputerów

ODRY 1305:

Akademia Ekonomiczna w Poznaniu  
Zakład Elektronicznej Techniki Obliczeniowej w Krakowie  
„Merinotex” w Toruniu  
„Agroma” w Szczypiornie k. Kalisza  
ETOB w Lublinie  
ETOB w Zielonej Górze  
„Len” w Żyrardowie  
ETOB w Bielsku-Białej  
Centralny Ośrodek Informatyki Drogownictwa w Warszawie  
Instytut Fizyki Jądrowej — Nowosybirsk (ZSRR)  
Zarząd Portu Lotniczego — Schönefeld (NRD)

Lutrija — Zagrzeb (Jugosławia)  
RSW „Prasa” — Warszawa  
Huta „Zygmunt” — Bytom  
„Chemidex” — Tomaszów Mazowiecki  
METEKON — Katowice  
PIT — Warszawa  
Zakłady Przemysłu Węglowego — Dąbrowa Górnicza  
Zakład Energetyczny — Gliwice  
OZO PKP — Łódź

ODRY 1325:

TESLA — Pardubice (CSRS)  
Zakłady Azotowe — Kędzierzyn  
UNITRA-RAWAR — Warszawa

— R-32:

Zakłady Mechaniczne — Ursus  
Fabryka Autobusów — Sanok  
ETOB — Kraków  
KONSTAL — Chorzów  
Predóm-Projekt — Wrocław  
POLMO — Lublin  
POLFA — Warszawa  
ROMET — Poznań  
Instytut Systemów Sterowania w Sosnowcu  
Politechnika Warszawska.

## Nowe kierunki szkolenia

Ośrodek Szkoleniowy ELWRO-SERWIS przyjmuje zgłoszenia na kursy na temat:

- systemu zarządzania siecią teletransmisji i przetwarzania translacji
- systemu automatycznego programowania na multipleksor.

Oferty należy kierować pod adresem: Ośrodek Szkoleniowy ELWRO-SERWIS, ul. Świerczewskiego 27, 50-044 Wrocław.

Jednocześnie informujemy, że kursy na temat programowania ODRY (PLAN, COBOL, FORTRAN), systemu operacyjnego GEORGE 3 oraz programowania komputerów JS (PL-1, RPG, FORTRAN, COBOL) prowadzi Ośrodek Szkoleniowy NOT. Także więc korespondencję dotyczącą kursów programowania należy kierować pod adresem: Ośrodek Szkoleniowy NOT, ul. Świerczewskiego 74, 50-020 Wrocław.

## Oprogramowanie komputerów JS

Dział Serwisu Oprogramowania ELWRO-SERWIS dysponuje obecnie następującym oprogramowaniem komputerów w systemie OS:

- biblioteka modułów matematycznych
- system obliczeń inżynierskich — SOWA
- Konwentor języka FORTRAN na PL-1.

## Wystawa Jednolitego Systemu w Moskwie

W dniu 15 czerwca br. zostanie otwarta w Moskwie wystawa poświęcona dziesięcioleciu porozumienia o Jednolitym Systemie Maszyn Cyfrowych.

Organizację ekspozycji polskiej koordynuje Centrum MERA-ELWRO, ale ekspozycja obejmie również urządzenia produkowane przez inne zakłady Zjednoczenia MERA, takie jak: „Elzab”, „Błonie”, „Meramat” KFAP. Na wystawie zaprezentowane zostaną m.in.: pamięć taśmowa EC 5001, pamięć taśmowa kasetowa EC 5091, drukarka wierszowa EC 7033.01, system monitorów ekranowych MERA 7900, procesor komunikacyjny EC 8371.01, stolik operatora EC 7076, zdalna stacja wsadowa EC 8514, terminal konwersyjny EC 8575, system do przygotowania danych na taśmie magnetycznej EC 9150, pamięci na dyskach elastycznych.

W oparciu o procesor komunikacyjny EC 8371.01 Polska zaprezentuje również na radzieckim komputerze EC 1045 następujące systemy:

- sterowania produkcją
- automatyzacji pomiarów eksperymentu biologiczno-fizycznego
- automatyzacji prac projektowo-konstrukcyjnych
- sterowania obrabiarkami i robotami przemysłowymi.

Zaprezentowane zostaną także stosowane w podsystemach teleprzetwarzania JS następujące systemy:

- dyspozycji resortowej
- informowania kierownictwa
- informowania władz terenowych
- sterowania jakością produkcji
- ewidencji kadrowej
- ewidencji materiałowej
- wyszukiwania informacji.

Na wystawie eksponowana będzie także tematyka kompleksowej obsługi, obejmująca serwis techniczny i programowy, szkolenie, projektowanie ośrodków obliczeniowych, działalność służb niezawodnościowych oraz zabezpieczenie w części zamienne.

Podczas „dni polskich” (rozpoczęcie przewidywane jest 27 czerwca) odbędą się okolicznościowe wykłady i zostaną wygłoszone referaty związane z rozwojem konstrukcji i obsługi komputerów JS w naszym kraju.

Wszystkie materiały opracował Jerzy Jankowski

# Łamy Informatyki otwarte dla wszystkich!

Zanim jednak nasi Autorzy sięgną po pióro, prosimy, by zechcieli zapoznać się z poniższymi informacjami.

Nadsyłane artykuły nie mogą być publikowane lub przeznaczone do opublikowania w innych czasopiśmiech.

W artykułach można omawiać, prezentować lub proponować wszystko, co dotyczy współczesnej informatyki, oraz wszystko, co wiąże się z jej kierunkami rozwoju — zarówno z pozycji informatyka, jak i użytkownika informatyki.

Materiał, oprócz tekstu zasadniczego, powinien zawierać — na oddzielnych stronach — kartę tytułową (strona 1), krótki życiorys zawodowy autora (strona 2) i jego zdjęcie, wykaz literatury, tabele, rysunki, podpisy pod rysunki, zdjęcia.

Na stronie 1 należy podać tytuł naukowy, imię i nazwisko, nazwę zakładu pracy, adres prywatny i telefon, tytuł artykułu oraz informację, jaką drogą przesłać honorarium po opublikowaniu artykułu: kasa Wydawnictwa, poczta, bank (w takim przypadku prosimy podać numer konta PKO).

Konstrukcja artykułu powinno być zwarta i przejrzysta; wstęp musi wprowadzić czytelnika w zagadnienie, w podsumowaniu należy sformułować wnioski; podział na rozdziały, podrozdziały i akapity powinien być logiczny i konsekwentny. Należy zwrócić szczególną uwagę na poprawność stylistyczną i terminologiczną, unikać skrótów, rzadko stosowanych wyrażen obcych i żargonu fachowego; starannie definiować nowe terminy. Należy również wystrzegać się nieczytelnych i zbyt rozbudowanych wzorów.

Tekst powinien być napisany na maszynie, jednostronnie, na papierze nieprzebitkowym formatu

A-4, z marginesem 5 cm (30 wierszy na 1 stronie, 60 znaków w 1 wierszu).

Wykaz literatury powinien zawierać: kolejny numer pozycji (w nawiasie kwadratowym), nazwisko i imię autora, tytuł publikacji (książki lub artykułu), ewentualnie tytuł i numer czasopisma (w przypadku artykułu), miejsce i rok wydania.

Tabele — każda na oddzielnej stronie — powinny być numerowane i opatrzone tytułem oraz ściśle związane z tekstem (odniesienie na marginesie).

Rysunki — każdy oddzielnie! (uwaga: nie wklejać rysunków w tekst!) — powinny być czytelne i również ściśle związane z tekstem (odniesienie na marginesie). Format rysunku nie może być mniejszy niż  $10 \times 10$  cm.

Podpisy pod rysunkami, napisane również na oddzielnej stronie, oprócz kolejnego numeru powinny zawierać tytuł rysunku i ewentualnie legendę dotyczącą poszczególnych elementów.

Łączna objętość materiału nie powinna przekraczać w przypadku

- artykułu problemowego — 12 stron
- reportażu — 8 stron
- recenzji, relacji z imprezy — 6 stron
- informacji — 4 stron maszynopisu

Tak przygotowany materiał prosimy dostarczyć w dwóch egzemplarzach pod adresem: redakcja INFORMATYKI, ul. Jasna 14/16, 00-041 Warszawa. Wszelkich dodatkowych informacji udzielamy pod telefonem 27-71-40.

Autor opublikowanego w INFORMATYCE artykułu otrzymuje bezpłatnie egzemplarz okazowy.

Materiałów nie zakwalifikowanych do druku redakcja nie zwraca.

*Wydawnictwo SIGMA – wydawca 75 czasopism technicznych dla inżynierów, techników i robotników wykwalifikowanych oraz dla czytelników interesujących się techniką pozazawodowo.*

*Wydawnictwo Czasopism i Książek Technicznych SIGMA jest przedsiębiorstwem Naczelnej Organizacji Technicznej; powstało w marcu 1979 r.; kontynuuje trzydziestoletni dorobek edytorski Wydawnictw Czasopism Technicznych NOT; w roku bieżącym przygotowuje się do podjęcia wydawania poradników technicznych, pomocy inżynierskich, wydawnictw szkoleniowych i popularyzatorskich.*

*Wydawnictwo SIGMA dzięki silnemu powiązaniu ze stowarzyszeniami naukowo-technicznymi sfinalizowanymi w NOT prezentuje na łamach wydawanych czasopism szeroki wybór artykułów, komunikatów i informacji z zakładów produkcyjnych, ośrodków badawczych i uczelni, a także opracowania techniczno-gospodarcze ukazujące opinie specjalistów na temat rozwoju poszczególnych gałęzi krajowego przemysłu, budownictwa, energetyki, telekomunikacji oraz na temat ochrony naturalnego środowiska człowieka.*

*Prenumeratę czasopism Wydawnictwa SIGMA (dawniej WCT NOT) przyjmują oddziały RSW „Prasa-Książka-Ruch” oraz urzędy pocztowe.*

*Wydawnictwo SIGMA – ul. Świętokrzyska 14a, 00-950 Warszawa, skrytka pocztowa 1004. Dyrektor – naczelny redaktor – tel. 26-91-52. Dział Handlowy – tel. 26-80-16.*