

P. 1877 / 79

12 1979



informatyka

NALEPSZE ŻYCZENIA ŚWIATECZNE
I HONOROCZNE
SKŁADA ZESPÓŁ REDAKCYJNY



Programy dwukierunkowej konwersji ODRA-RIAD w systemie OS/JS <i>Andrzej Żurawski</i>	1
Zastosowania informatyki w organizacjach gospodarczych w świetle badań ankietowych <i>Marian Kuraś</i>	3
Informatyka w projektowaniu budowlanym (na tle konferencji INFOPRO 79 i PaC 79) <i>Maciej Robakiewicz</i>	8
Programowanie w PROLOGU: informator kolejowy <i>Stanisław Szpakowicz, Marek Świdziński</i>	12
Układy bardzo dużej integracji w technice komputerowej <i>Tadeusz Sinkiewicz</i>	16
SPIS TREŚCI ROCZNIKA 1979	20
Z KRAJU	
ELORG seminarium SM EMC Polska 1979 <i>Adam B. Empacher</i>	23
Posiedzenie grupy roboczej IFIP w Warszawie (W.K.)	23
ZE ZJEDNOCZENIA INFORMATYKI	
Zastosowanie monitorów ekranowych w systemie gospodarki materiałowej <i>Jan Cichoń, Janusz Czekaj, Ryszard Jankowski</i>	24
ZE SWIATA	
Informatyka w Tennessee Valley Authority <i>Jan W. Owsiński</i>	27
ICL 7709 (T.J.)	29
IKD 1980 (W.K.)	29
EuroIFIP 79 <i>Stanisław Jaskólski</i>	36
CENTRUM ETOB	
Współpraca polsko-rumuńska <i>Wincenty Łada</i>	32
III samochodowy rajd integracji (wład)	33
OSRODKI INFORMATYKI PREZENTUJĄ	
ROBOTRON 4201 do zarządzania produkcją urządzeń elektrycznych <i>Lech Łasica, Kazimierz Zieliński</i>	35
PROBLEMATYKA BAZY DANYCH	
Kompresja baz danych za pomocą kodów zmiennej długości <i>Andrzej Siemiński</i>	36
NASZE RECENZJE	
Repetytorium języka PLAN <i>Adam B. Empacher</i>	40

KOLEGIUM REDAKCYJNE

Redaktor naczelny: prof. dr hab. Leon ŁUKASZEWICZ
 dr 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 Wincenty ŁADA,
 mgr Stanisław MROZIK, dr inż. Tomasz PAWLAK.

Sekretarz redakcji: Anna GLUTH-NOWOWIEJSKA.

Red. techn.: Ewa KAMINSKA

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



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

ANDRZEJ ŻURAWSKI

Katowickie Przedsiębiorstwo Informatyki
Przemysłu Budowlanego ETOB

Programy dwukierunkowej konwersji ODRA-RIAD w systemie OS/JS

W Katowickim Przedsiębiorstwie Informatyki Przemysłu Budowlanego ETOB w 1978 roku jako pierwszy w sieci tej organizacji został zainstalowany komputer R-32. Nakłada to na przedsiębiorstwo szczególne obowiązki zarówno w zakresie opracowania nowych systemów informatycznych, jak i adaptacji już istniejących i eksploatowanych systemów na komputerach MINSK 32 oraz ODRA serii 1300, w które wyposażony był dotąd resort budownictwa.

W niniejszym artykule zostanie przedstawiony sposób rozwiązania problemu przenoszenia istniejącego oprogramowania na poziomie programów źródłowych z komputerem ODRA 1300 m R-32 oraz wzajemnej wymienności między tymi komputerami systemów EPD na poziomie zbiorów.

Z przeprowadzonej analizy wynikało, że dotychczasowe rozwiązania tego problemu w innych ośrodkach krajowych zostały opracowane z wykorzystaniem systemu operacyjnego DOS. Ponieważ komputery Jednolitego Systemu w sieci ETOB wyposażone są tylko w system operacyjny OS/JS, podjęto opracowanie programów konwersji zbiorów ODRA-RIAD eksploatowanych z wykorzystaniem tego systemu. W wyniku omawianych działań wykończono dwa takie programy: IEBCNVOR (ODRA→RIAD) i IEBCNVRO (RIAD→ODRA), zapewniające pełną (obukierunkową) konwersję.

CHARAKTERYSTYKA PROGRAMÓW KONWERSJI

Jak widać z samych nazw programów zostały one zaprojektowane jako rozszerzenie zestawu programów pomocniczych systemu OS/JS. Spełniają one wszystkie standardy oprogramowania podstawowego i charakteryzują się jednolitym sposobem wykorzystywania.

Podstawową cechą omawianych programów jest ich uniwersalność, która pozwala na dwustronne przenoszenie dowolnych zbiorów. Osiągnięto to poprzez zastosowanie szerokiego zestawu parametrów w zdaniach sterujących programów, które pozwalają między innymi na:

• dwukierunkowe kopiowanie prostych zbiorów lub prostych podzbiorów zbioru złożonego na zbiory sekwencyjne lub człony zbiorów bibliotecznych

- redagowanie danych wyjściowych poprzez konwersję typów danych z jednej postaci na inną
- pomijanie wybranych rekordów lub innych niż wybrane w procesie kopiowania z redagowaniem
- redagowanie poszczególnych grup rekordów w odrębny sposób.

W przypadku, gdy wyżej wymienione szerokie możliwości programów nie są dla użytkownika wystarczające, może on dołączać własne procedury, które mogą otrzymywać sterowanie po każdym przeczytanym rekordzie lub przed każdym zapisanym rekordem. Dzięki przekazywaniu procedurom zgodnie ze standardami systemu OS/JS w rejestrze 1 listy parametrów w postaci:

- adres rekordu (wejściowego lub wyjściowego w zależności od typu procedury)
- adres DCB zbioru wyjściowego
- adres DCB zbioru komunikatów

użytkownik może między innymi:

- dokonać modyfikacji w rekordzie wejściowym lub wyjściowym
- wykonać samodzielnie konwersję całego lub części rekordu i zapisać go w zbiorze wyjściowym

Mgr Andrzej ŻURAWSKI ukończył w 1972 roku studia matematyczne (sekcja numeryczna) na Uniwersytecie Jagiellońskim w Krakowie. Od ukończenia studiów pracuje w Katowickim Przedsiębiorstwie Informatyki Przemysłu Budowlanego ETOB, gdzie obecnie jest kierownikiem sekcji technologiczno-programowej. W 1976 roku ukończył trysemestralne studia podyplomowe z organizacji przetwarzania danych na Akademii Ekonomicznej w Katowicach.



- zapisać dowolną liczbę rekordów w zbiorze wyjściowym na podstawie informacji z rekordu wejściowego
- wyprowadzać komunikaty do systemowego zbioru komunikatów.

Jednocześnie może on sterować pracą programu głównie poprzez użycie kodów powrotu, nakazując:

- kontynuację pracy po modyfikacji lub bez modyfikacji rekordu wejściowego lub wyjściowego
- przerwanie konwersji i zamknięcie zbiorów
- pominięcie w dalszym przetwarzaniu określonego rekordu wejściowego lub wyjściowego
- kontynuację pracy z zablokowaniem dalszych wejść do danej procedury.

Istotnymi zaletami omawianych programów konwersji są również:

- stosunkowo niewielka zajętość pamięci operacyjnej
- duża szybkość konwersji, która dla konwersji średnio skomplikowanej nie odbiega istotnie od szybkości kopiowania.

Zajętość PAO wynosi przeciętnie 16 KB. Zależy ona od maksymalnych długości bloków i rekordów oraz od liczby różnych typów konwersji pól występujących w rekordzie. Wspomnianą niewielką zajętość PAO osiągnięto poprzez wszechstronne użycie makroinstrukcji systemowej GET-MAIN dla żądania tylko takiej wielkości PAO, jaka wynika z analizy zestawu zdań sterujących.

Również na podstawie analizy zdań sterujących generowane są ciągi rozkazów konwersji dla poszczególnych typów rekordów, co pozwala na osiągnięcie wspomnianej dużej szybkości działania.

STEROWANIE PRACĄ PROGRAMÓW

Celem lepszego zobrazowania możliwości i sposobu pracy programów przedstawiono poniżej sposób sterowania programem IEBCNVOR (ODRA→RIAD). Zasady stosowane w programie IEBCNVRO (RIAD→ODRA) są bardzo podobne, a niewielkie różnice wynikają jedynie z odwrotnych funkcji programu.

Program IEBCNVOR jest sterowany zdaniami języka opisu zadań (JOB, EXEC oraz DD o nazwach SYS-PRINT, SYSUT1, SYSUT2 i SYSIN) oraz zdaniami sterującymi program (CONV i RECORD). Zasady kodowania zdań sterujących są identyczne jak w programach pomocniczych systemu OS/JS. Zdanie funkcji CONV, które występuje jako pierwsze zdanie sterujące, określa atrybuty zbioru wejściowego, maksymalną liczbę zdań RECORD oraz ewentualne nazwy procedur użytkownika. Jego postać jest następująca:

CONV DSNAME → (nazwa zbioru we)
[,MEMBER = (nazwa podzbioru)]

[,VOLSER = { numer seryjny wolumenu
[NO oznacza pominięcie kontroli] }]

[,VOLNO = { numer kolejny krążka
[1] }]

[,GENNO = { numer generacji zbioru
[O oznacza pominięcie kontroli] }]

[,LRECLMAX =
= { maksymalna długość rekordu wejściowego
[512] }]

[,BLKMAX = { maksymalna długość bloku wejściowego
[4096] }]

[,DEN = { 0-gęstość zapisu TM wejściowej 200 bpi
[2-gęstość zapisu TM wejściowej 800 bpi] }]

[,MAXRCD = { maksymalna liczba zadań RECORD
[1] }]

[,INREC = nazwa procedury wejścia użytkownika]

[,OUTREC → nazwa procedury wyjścia użytkownika]

Zdanie RECORD służy do opisu poszczególnych typów rekordów i sposobu ich konwersji. Postać zdania RECORD jest następująca:

RECORD PIC → ((typ pola wejściowego początek pola wejściowego, długość pola wejściowego, typ pola wyjściowego, początek pola wyjściowego, długość pola wyjściowego),...).

Parametr powyższy podaje sposób konwersji poszczególnych pól w rekordzie. Dopuszczalne typy są następujące:

- dla pól wejściowych ODRA
- CH — pole znakowe
- ZD — numeryczne pole znakowe bez znaku
- SD — numeryczne pole znakowe ze znakiem (pierwszy bajt pola jest znakiem)
- CD — numeryczne pole znakowe ze znakiem (znak zwarty w pierwszym bajcie liczby)
- BI — numeryczne pole binarne
- dla pól wyjściowych RIAD
- CH — pole znakowe
- ZD — pole dziesiętne rozpakowane bez znaku
- SD — pole dziesiętne rozpakowane ze znakiem
- PD — pole dziesiętne spakowane
- BI — numeryczne pole binarne.

Należy zwrócić uwagę, że dzięki modułowej budowie programu mogą być w sposób prosty dodane konwersje pominiętych typów danych. Jednocześnie problem ten może być rozwiązany poprzez wcześniej opisane stosowanie procedury użytkownika.

[,IDENT = (początek pola, długość pola, wartość pola)].

Parametr ten opisuje identyfikator grupy rekordów, która jest redagowana w jednakowy sposób. Jeżeli nie podano parametru, to zdanie RECORD odnosi się do wszystkich rekordów z wyjątkiem rekordów identyfikowanych innymi zdaniami RECORD

[,CONV = { NO oznacza pominięcie rekordów danej grupy
[YES] }]

[,FILL = { znak, którym wypełniane są nieustawione
[pola wyjściowe X'OO'] }]

[,LRECL = długość rekordu wyjściowego danej grupy].

Jeżeli tego parametru nie podano, przyjęta zostaje długość z bloku DCB zbioru wyjściowego

[,INREC = nazwa procedury wejścia użytkownika]

[,OUTREC = nazwa procedury wyjścia użytkownika].

Parametry INREC i OUTREC zdania RECORD są mocniejsze od analogicznych parametrów podanych w zdaniu CONV i obowiązują dla danej grupy rekordów.

Jak widać z powyższego opisu programów konwersji umożliwiają one przenoszenie dowolnych zbiorów pomiędzy komputerami ODRA i RIAD. Wykonanie konwersji nie powinno nastęczać żadnych problemów użytkownikom wykorzystującym system OS/JS.

Sprostowanie

Serdecznie przepraszamy Autorów i Czytelników za błędy w artykule doc. dr. hab. inż. Andrzeja Gościńskiego i doc. dr. hab. inż. Edwarda Nawrockiego pt. „Komputerowe sterowanie procesami dyskretnymi”, opublikowanym w tegorocznym numerze 8/9 INFORMATYKI.

Zniekształcono nazwisko Autora (Nawrocki zamiast Nawarecki) zamieniono zdjęcia Autorów przy Ich życiorysach, a rys. 3 (s. 9) znalazł się w artykule „Maszyny bazy danych” jako rys. 1 (s. 56).

Błędy nie powstały z winy redakcji.

Zastosowania informatyki w organizacjach gospodarczych w świetle badań ankietowych

W wybranych przedsiębiorstwach przemysłowych na terenie całego kraju przeprowadzono ostatnio badania nad zastosowaniami informatyki. Ankieta, adresowana do członków kierownictwa, działów organizacji i zakładowych ośrodków informatyki, miała na celu uzyskanie informacji na temat zastosowań informatyki, a szczególnie:

- warunków, w jakich są realizowane
- zakresu i charakterystyk tych zastosowań
- oceny z różnych punktów widzenia
- potrzeb i oczekiwań użytkowników
- efektywności eksploatowanych systemów
- metod stosowanych w realizacji systemów informatycznych.

Złożoność problematyki badań spowodowała, że zakres pytań kierowanych do poszczególnych grup respondentów był bardzo szeroki. Jednocześnie, ze względu na to, że jedyną dostępną formą przeprowadzenia badań okazała się ankieta pocztowa, eliminująca bezpośredni kontakt z respondentem, pytania w przeważającej większości zostały skategoryzowane w oparciu o wcześniej przeprowadzone badania pilotażowe. Ponadto respondentom pozostawiono możliwość udzielenia dodatkowych wyjaśnień i komentarzy, z czego wielu ankietowanych korzystało.

Uzyskane wypowiedzi przyniosły wiele informacji, które może nie upoważniają do wyciągania ogólnych wniosków, ale pozwalają określić tendencje w rozwoju zastosowań informatyki w przedsiębiorstwach przemysłowych.

CHARAKTERYSTYKA BADANYCH PRZEDSIĘBIORSTW

Po odrzuceniu ankiet niekompletnych do analizy przyjęto 55 ankiet, co odpowiada 58% wybranych do badań przedsiębiorstw. Wśród tych przedsiębiorstw najliczniej reprezentowane są gałęzie, w których technika komputerowa jest najpowszechniej stosowana: przemysł chemiczny (41,8%), maszynowy (32,7%), ciężki (9,1%). Badane przedsiębiorstwa są zróżnicowane pod względem wielkości (małe — 29,1%, średnie — 34,5%, duże — 20,0%, wielkie — 16,4%)¹⁾, co pozwoliło zaobserwować związane z tym prawidłowości.

Wszystkie badane przedsiębiorstwa mają zakładowe ośrodki informatyki. Ich zróżnicowanie jest jednak bardzo poważne: od małych trzyosobowych komórek organizujących wdrożenia i eksploatację systemu informatycznego w oparciu o dzierżawiony sprzęt do wielkich ośrodków zakładowych wyposażonych w duże komputery, samodzielnie realizujących systemy informatyczne.

Różne są też struktury organizacyjne. Zwierzchnikiem ośrodka jest najczęściej dyrektor ekonomiczny (52,7%) i główny księgowy (30,9%). Takie przyporządkowanie ośrodka przez 65,5% respondentów uważane jest za czynnik ułatwiający, przez 9,0% — utrudniający, a przez 20,0% — za nie mający znaczenia dla realizacji prac. Jedynie w czterech przedsiębiorstwach połączono ośrodek z działem organizacji, mimo że 67,3% organizatorów i 52,7% informatyków

¹⁾ Każde z badanych przedsiębiorstw zaliczone zostało do jednej z czterech kategorii na podstawie: wartości produkcji, wielkości zatrudnienia i wartości środków trwałych

uważa, że istnieją wspólne obszary w działalności obu komórek i że 56,4% informatyków i 40,0% organizatorów stwierdza konieczność współpracy.

Przetwarzanie danych w badanych przedsiębiorstwach opierało się dotychczas głównie na technice ręcznej (w 15 przedsiębiorstwach — 27,3%), technice mechanicznej (w 18 przedsiębiorstwach — 32,7%), a w 17 przedsiębiorstwach (30,9%) stosowano w równej mierze obydwie techniki.

Prace nad komputeryzacją przetwarzania trwały od ponad 8 lat w 18,2% badanych przedsiębiorstwach, 7—8 lat w 20,0%, 5—6 lat w 30,9%, 3—4 lata w 21,8% a krócej niż dwa lata w 9,1% przedsiębiorstwach.

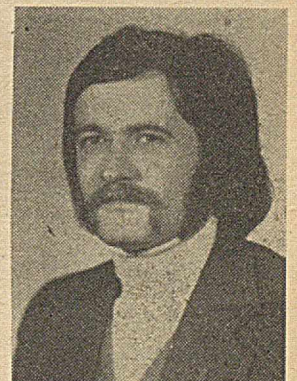
Systemy informatyczne eksploatowane są w oparciu o różnorodny sprzęt, przy czym najczęściej wykorzystywanymi komputerami są ODRY 1300⁻ (54,5%) i komputery importowane z krajów socjalistycznych (40,0%). Wielu użytkowników eksploatuje swoje systemy w oparciu o więcej niż jeden typ komputerów.

43,6% badanych przedsiębiorstw dysponuje własnym sprzętem komputerowym, który nie zawsze zaspokaja wszystkie potrzeby. Przedsiębiorstwa korzystają ponadto z usług ośrodków branżowych (34,5%), resortowych (7,3%), terenowych (5,5%), ZETO (29,1%) oraz innych (14,5%).

Większość przedsiębiorstw zorganizowała szkolenie na temat zastosowań informatyki dla kadry kierowniczej (72,7%), pracowników inżynieryjno-technicznych (60,0%) i dla pracowników administracyjno-biurowych (70,9%). Zakres tego szkolenia obejmował w większości przypadków informacje na temat funkcjonowania maszyny cyfrowej, programowania i projektowania systemów informatycznych o typowej strukturze.

Ankiety wypełniło 212 dyrektorów i innych członków kierownictwa badanych przedsiębiorstw (tabela 1).

Dyrektorzy biorący udział w ankiecie w większości (71,2%) przeszli przeszkolenie z zakresu zastosowań informatyki i twierdzą, iż w wyniku tego szkolenia uświadamiają sobie korzyści, jakie przynosi korzystanie z komputerów (69,8%).



Mgr Marian KURAS w 1970 r. ukończył studia na kierunku „Ekonomika przemysłu” w Wyższej Szkole Ekonomicznej w Krakowie. W tym samym roku rozpoczął pracę w przemyśle chemicznym. W 1972 r. ukończył Podyplomowe Studium Projektowania Systemów EPD w WSE w Krakowie. Od 1973 r. jest pracownikiem naukowo-dydaktycznym w Zakładzie Informatyki Akademii Ekonomicznej w Krakowie.

Tabela 1. Struktura członków kierownictwa uczestniczących w badaniach

Stanowisko respondenta	Liczba	Proc.	Gałęzie przemysłu				Wielkość przedsiębiorstwa			
			ciężki	maszyno- wy	chemiczny	inne	małe	średnie	duże	wielkie
Dyrektor naczelny	20	9,4	3	4	8	5	4	10	5	1
Zastępca ds. technicznych	26	12,3	4	14	7	1	7	10	3	6
Zastępca ds. ekonomicznych	53	25,0	5	16	18	14	20	16	9	8
Inni zastępcy techniczni	29	13,7	3	15	9	2	6	8	3	12
Inni zastępcy ekonomiczni	23	10,8	2	11	3	7	2	7	8	6
Główny księgowy	37	17,5	4	10	18	5	10	15	7	5
Główni specjaliści i pozostali członkowie kierownictwa	24	11,3	3	6	12	3	6	9	3	6
Razem	212	100,0	24	76	75	37	55	75	38	44
Procent	100,0		11,3	35,8	35,4	17,5	25,9	35,4	17,9	20,8

Respondenci wysoko oceniają przygotowanie zawodowe informatyków (61,8% — dobre, 17,5% — bardzo dobre), znajdują wspólny język z informatykami (88,2%), uważają, że informatycy znają specyfikę przedsiębiorstwa (80,2%) i uwzględnili tę specyfikę w realizowanym systemie informatycznym (78,8%). Równie zdecydowana większość dyrektorów twierdzi, że pełnomocnictwa, jakich udzielili informatykom, nie są nadmierne (67,0%) oraz że informatycy konsultowali z nimi rozwiązania (70,8%), w efekcie czego mogą uważać, iż mieli wpływ na przyjęcie określonych rozwiązań (63,2%).

ZDIEDZINY ZASTOSOWAŃ INFORMATYKI W BADANYCH PRZEDSIĘBIORSTWACH

Omówienie systemów informatycznych realizowanych w badanych przedsiębiorstwach należy rozpocząć od omówienia strategii i koncepcji komputeryzacji.

Automatyzacja przetwarzania realizowana jest najczęściej w ramach programów opracowanych dla branż lub resortów (64%) i indywidualnie przez przedsiębiorstwa (56%), ponadto w ramach usług świadczonych przez ośrodki ZETO (27%) lub adaptacji gotowych projektów (18%).

Nie wyklucza to jednak możliwości daleko idących modyfikacji systemów branżowych, jak również współpracy w opracowywaniu systemów z ośrodkami ZETO, bądź zakupu gotowych systemów i ich adaptacji.

Zdecydowana większość przedsiębiorstw deklaruje realizację kompleksowego systemu informatycznego, którego koncepcja została wypracowana dla całej branży/resortu (30,9%) lub dla przedsiębiorstwa (29,1%). Pozostałe 40% przedsiębiorstw projektuje i wdraża niezależne systemy pozostawiając problem ich scalenia na przyszłość (34,5%) lub nie przewidując ich scalenia (5,5%).

Tabela 2. Zakres zastosowań Informatyki w przetwarzaniu danych w badanych przedsiębiorstwach [proc.]

Funkcja	Zgłoszone, potrzeby dyrektora	System informatyczny			Bez odpowiedzi
		eksploatowane	częściowo wdrożony	realizacja planowana	
Ewidencja i rozliczanie w zakresie:					
— gospodarki materiałowej		70,9	14,5	9,1	5,4
— gospodarki środkami trwałymi	72,7	50,9	14,5	29,1	5,4
— siły roboczej		40,0	18,2	18,2	23,6
— produkcji		27,3	14,5	27,3	30,9
— wyrobów gotowych		50,9	10,9	20,0	18,2
— finansów		29,1	5,4	32,7	32,7
Planowanie	45,5	14,5	18,2	10,9	56,4
Rachunek kosztów	29,1	36,4	7,3	20,0	36,4
Sterowanie zapasami	76,4	21,8	7,3	27,3	43,6
Informowanie kierownictwa	9,1	3,6	1,8	0	94,5

W przetwarzaniu danych w badanych organizacjach gospodarczych dominują systemy ewidencji i rozliczeń (tabela 2).

Wymienione w tabeli sfery zastosowań w zasadzie odpowiadają potrzebom zgłaszanym przez dyrektorów. Analiza danych zawartych w powyższej tabeli skłania jednak do zastanowienia, czy rzeczywiście wypowiadający się członkowie kierownictwa uznają za najważniejsze zadania automatyzacji ewidencji i rozliczeń, czy nie widzą innych możliwości wykorzystania techniki komputerowej jako narzędzia doskonalenia procesu zarządzania.

W powyższym zestawieniu zwraca uwagę wysoki odsetek dyrektorów żądających wprowadzenia automatycznego systemu sterowania zapasami. Zainteresowanie to, jak można przypuszczać, spowodowane było aktualnością tematu w okresie prowadzenia badań. Podobnie zawyżone wydają się wypowiedzi informatyków, gdyż jedynie w 4 przypadkach można mówić o wdrażanych systemach sterowania zapasami. W pozostałych przypadkach deklaracje wynikają ze zbyt optymistycznej oceny możliwości systemu ewidencji materiałów.

Nieco za wysoki wydaje się również odsetek przedsiębiorstw przyznających się do eksploatacji automatycznego systemu rachunku kosztów, gdyż połowa tych rozwiązań to pakiety programów opracowujących ograniczony zakres zestawień w oparciu o dane wprowadzane z tradycyjnych nośników informacji.

Podobnie wygląda sytuacja w systemach planowania, których realizacji domagają się tak często dyrektorzy. Większość z wymienionych przez informatyków systemów to pakiety „przeliczania planów”, nie zasługujących na określenie automatycznych systemów planowania.

Struktura systemów informatycznych (respondenci opracowali opis struktury systemu i ogólny plan jego realizacji) realizowanych w badanych organizacjach gospodarczych jest niemalże identyczna, a co najwyżej wykazuje pewne odchylenia od istniejącego wzorca. Fakt, że struktura systemu informatycznego jest niezależna od koncepcji komputeryzacji powinien dawać wiele do myślenia.

Bardzo interesujące są wypowiedzi informatyków dające wyobrażenie o przebiegu realizacji systemów informatycznych. Regułą jest, że dla każdej jednostki przetwarzania opracowuje się odrębnie założenia i czas, jaki upływa od decyzji o rozpoczęciu prac do przyjęcia założeń, waha się od 1 do 24 miesięcy (średni czas — 4 miesiące). Czas, jaki upływa od decyzji o realizacji do wdrożenia, wynosi od 3 do 36 miesięcy (czas średni — 10 miesięcy). Próbną eksploatacja trwa średnio 3 miesiące, ale w niektórych z badanych przedsiębiorstw przedłuża się do 6 miesięcy. Do rzadkości należy przekazanie wdrożonego systemu informatycznego zainteresowanym służbom. W ponad 75% przypadków eksploatację systemu organizuje koordynator systemu z zakładowego ośrodka informatyki.

OCZEKIWANIA I OCENY EKSPLOATOWANYCH SYSTEMÓW INFORMATYCZNYCH

Badania przyniosły informacje o celach, jakim w opinii dyrektorów w badanych przedsiębiorstwach miało służyć wprowadzenie techniki automatycznej do przetwarzania danych: odciążenie personelu od prac obliczeniowych

(67,9%), zaspokojenie braku informacji (54,2%), stworzenie możliwości doskonalenia organizacji (25,0%), dążenie do obniżenia kosztu informacji (17,0%), względy prestiżowe, moda (3,3%), obsługa informacyjna procesu decyzyjnego (1,9%).

Tak określone cele wskazują na ekstensywny charakter zastosowań i uzasadniają strukturę systemów informatycznych oraz kolejność wdrożeń. Dyrektorzy w większości są zdania, że cele te zostały osiągnięte (60,9%), choć duża grupa wyraża odmienne zdanie (23,6%). 60% informatyków twierdzi, że cele zostały przez dyrekcję sprecyzowane w sposób jasny i konkretny, przeciwną opinię wyraża aż 30,9% respondentów.

Nieco odmiennie przedstawiono własne oceny możliwości uzyskania określonych efektów w wyniku wprowadzenia automatycznego przetwarzania danych (tabela 3).

Tabela 3. Oczekiwane i uzyskane efekty wdrożenia systemu informatycznego w ocenie dyrektorów [proc.]

Efekty	Oczekiwane			Uzyskane		
	Tak	Nie	Brak opinii	Tak	Nie	Brak opinii
Przyspieszenie informacji	80,6	2,4	17,0	62,7	15,6	21,7
Zwiększenie zakresu	78,8	2,4	18,8	65,1	12,3	22,6
Zmniejszenie zatrudnienia	44,3	26,9	28,8	31,1	34,0	34,9
Obniżenie kosztu	30,7	29,2	40,1	23,6	28,3	48,1
Doskonalenie organizacji	14,6	0	85,4	11,3	3,3	84,9

Oczekiwania dyrektorów dotyczyły głównie przyspieszenia informatyzacji i zwiększenia jej zakresu, w dużo mniejszym stopniu — obniżenia jej kosztu. Na ogół uważa się, że oczekiwania zostały spełnione, choć nie brak i ocen przeciwnych.

Respondenci określili najważniejsze obszary, które ich zdaniem wymagają automatyzacji. Celowe pozostawienie pełnej swobody co do formy określenia preferencji spowodowało, że uzyskano niejednolite wypowiedzi. W związku z tym analiza potrzeb w zakresie automatyzacji przetwarzania w opinii dyrektorów została przeprowadzona w trzech przekrojach:

1) dziedziny zastosowań

obsługa informacyjna zarządu przedsiębiorstwa	— 85%
obsługa informacyjna procesu produkcyjnego	— 44%
sterowanie procesów technologicznych	— 13%
obliczenia inżynierskie	— 10%
metody matematyczne w zarządzaniu	— 8%
brak opinii	— 12%

2) funkcje

ewidencja	— 73%
planowanie	— 46%
rachunek kosztów	— 29%
informowanie kierownictwa	— 9%
sterowanie zapasami	— 67%
brak opinii	— 12%

3) informacje oczekiwane przez użytkowników

informacje ewidencyjno-rozliczeniowe	— 69%
informacje planistyczno-kontrolne	— 42%
informacja dla sprawozdawczości	— 30%
informacje do analiz	— 26%
informacje operatywne	— 21%
informacje normatywne	— 20%
brak opinii	— 12%

Potrzeby zgłoszone przez dyrektorów dotyczą tak automatyzacji przetwarzania danych gospodarczych, jak i sterowania procesami technologicznymi oraz obliczeń inżynierskich. Te ostatnie sygnalizowane są nie tylko przez techników, często też przez dyrektorów ekonomicznych i głównych księgowych.

Zdecydowane pierwszeństwo przyznają respondenci potrzebom w zakresie ewidencji i rozliczania. Systemy ewidencyjno-rozliczeniowe realizowane w badanych przedsiębiorstwach w dużej mierze determinują wyobrażenia o możliwościach i obszarach zastosowań techniki komputerowej do przetwarzania danych gospodarczych. Nie można również zapominać o znaczeniu, jakie w kształtowaniu wiedzy informatycznej użytkowników ma proces szkolenia, który z reguły obejmuje obok omówienia funkcjonowania maszyny cyfrowej i programowania komputerów prezentację gotowej, uniwersalnej struktury systemu informatycznego.

Analiza odpowiedzi zawierających oceny zastosowań informatyki wykazuje, że wielu dyrektorów, wstrzymuje się od wyrażenia opinii, dość często wyrażane są opinie krytyczne. Generalnie jednak zdecydowana większość tak dyrektorów jak i informatyków uważa, że przyjęta koncepcja komputeryzacji przetwarzania jest słuszna. Zdanie takie wyraża 84% dyrektorów (4% ma odmienną opinię, a 12% nie wyraża swojej opinii) i 84% informatyków (5% koncepcję uważa za niewłaściwą, a 11% nie ma wyrobionej opinii). Znacznie mniejsza grupa dyrektorów (67%) jest zdania, iż struktura systemu informatycznego i kolejność wdrażania jego modułów odpowiada potrzebom przedsiębiorstwa (20% uważa, że brak jest takiej odpowiedniości). Bardziej konsekwentni są informatycy — w 82% odpowiadają pozytywnie na to pytanie (11% ma odmienną zdanie). W odpowiedzi na pytanie sprawdzające tylko 58% respondentów nie uważa za słuszną potrzebę zmiany koncepcji. 15% jest zdania, że należy zmieniać koncepcję i że jest to możliwe, a 16% nie widzi realnej możliwości zmiany koncepcji, mimo iż jest to konieczne.

Zasugerowane twierdzenie, że system informatyczny obsługuje wybrane komórki przedsiębiorstwa, zaakceptowało 74% ankietowanych dyrektorów, a 46% uważa, że system realizuje swe funkcje dla przedsiębiorstwa jako całości, przy czym 33% respondentów nie widzi sprzeczności pomiędzy obydwoimi twierdzeniami.

Dyrektorzy w większości (81%) twierdzą, że odczuli zmianę w wyniku wdrożenia systemu informatycznego i zmiany te uważają za korzystne (89%).

Bardzo istotną informacją umożliwiającą ocenę efektów wdrożenia systemu informatycznego byłoby średni czas oczekiwania na informacje żądane przez dyrektorów. Niestety aż 62% dyrektorów nie udzieliło odpowiedzi na pytanie. Z wypowiedzi pozostałych respondentów można dowiedzieć się, że 14% oczekuje na informacje krócej niż godzinę, 7% — 2 do 4 godzin, 17% — 5 do 8, 21% — 9 do 24, 18% — 25 do 48, 9% — 49 do 120, a 14% ponad 120 godzin, 38% respondentów, którzy określili czas, czeka więc na informacje krócej niż 8 godzin, 39% — 9 do 48 godzin, a 23% — ponad 48 godzin. Dane te drastycznie charakteryzują sytuację decydentów w warunkach systemu przetwarzania danych nastawionego na realizację rozbudowanych zadań ewidencyjnych i sprawozdawczych. Najdłużej na informacje oczekują dyrektorzy ekonomiczni i główni księgowi, a kierownicy pionów technicznych najczęściej nie podają odpowiedzi.

W opinii 45% respondentów po wdrożeniu systemu czas oczekiwania na informacje uległ skróceniu, 4% twierdzi, że wydłużył się, 8% uważa, że nie uległ zmianie, a 43% nie ma wyrobionej opinii.

Duży odsetek (61%) respondentów utrzymuje, że wyniki opracowane przez komputer wykorzystują do podejmowania decyzji, co jest o tyle dziwne, że systemy informatyczne w badanych przedsiębiorstwach są w większości systemami ewidencyjno-rozliczeniowymi o miesięcznym cyklu przetwarzania.

Tabela 4. Ocena przydatności informacji do podejmowania decyzji [proc.]

Treść pytania	Tak	Nie	Brak opinii
Przydatność informacji do podejmowania decyzji	87,9	5,2	26,9
— aktualność	51,9	14,1	34,0
— kompletność	37,7	19,4	42,9
— wszechstronność	11,8	37,3	50,9
— dokładność	53,3	8,0	38,7

Aktualność informacji stwierdza 65% dyrektorów naczelnych, 58% ekonomicznych i 60% głównych księgowych. Kompletność potwierdza 45% dyrektorów naczelnych i 54% głównych księgowych. Bardziej krytyczna jest ocena informacji ze względu na ich wszechstronność. Brak tej wszechstronności wytykają dyrektorzy naczelni (50%), ekonomiczni (49%), główni księgowi (46%) i główni specjaliści (57%). Większość respondentów we wszystkich grupach uważa informacje za odpowiednio dokładne.

Wdrożone systemy informatyczne nie zaspokajają wszystkich potrzeb, co w obszarze objętym przetwarzaniem automatycznym stwierdzają dyrektorzy (67% — 70% dyrektorów naczelnych, 69% technicznych, 66% ekonomicznych i 89% głównych księgowych), informatycy (75%) i organizatorzy (58%). Ci ostatni, oceniając efekty wdrożenia systemu informatycznego, tylko w 51% uznali, że wdrożenie systemu stworzyło możliwości doskonalenia organizacji, a odmienną opinię wyraziło 33%. Jednocześnie 73% specjalistów z dziedziny organizacji wyraziło nadzieję, że przyszłe zastosowania stworzą takie możliwości.

Należy dodać, że nie we wszystkich przedsiębiorstwach wdrażaniu systemu informatycznego towarzyszą zmiany organizacyjne. Zmiany takie przeprowadzono w 60% badanych organizacji (w 9% uznano je za bezcelowe), a w 33% zmian nie wprowadzono, mimo iż 13% respondentów uważa je za konieczne.

Organizatorzy i informatycy przedstawili swoje oceny opinii bezpośrednich użytkowników o eksploatowanych systemach informatycznych. W ocenie informatyków opinie negatywne występują bardzo rzadko i dotyczą tylko marginesu użytkowników, podobnie jak i brak opinii, podczas gdy opinie raczej pozytywne (ok. 50%) i zdecydowanie pozytywne (ok. 30%) są najpowszechniejsze. W ocenach organizatorów ok. 7% użytkowników ma negatywną opinię o eksploatowanym systemie, ok. 35% — raczej pozytywną, a liczba użytkowników, których opinia jest zdecydowanie pozytywna, szacowana jest na ok. 25%. Ocen nie podało ponad 25% respondentów.

EFEKTYWNOŚĆ WDROŻONYCH SYSTEMÓW INFORMATYCZNYCH

Charakteryzując efekty wdrożenia systemu informatycznego, organizatorzy w blisko połowie ankiet uchylili się od udzielenia odpowiedzi (tabela 5), co może świadczyć o braku rozeznania w zmianach zachodzących w organizacji.

Tabela 5. Ocena efektów dokonana przez specjalistów ds. organizacji [proc.]

Rodzaj efektu	Zwiększenie	Zmniejszenie	Bez zmian	Brak opinii
Centralizacja decyzji	14,5	9,5	16,4	60,0
Poziom zatrudnienia	18,2	30,9	10,9	40,0
Liczba zestawień wyników	49,1	7,3	0	43,6
Zakres informacji	54,6	1,8	1,8	41,8
Dostępność informacji	56,4	0	1,8	41,8
Koszt informacji	49,1	1,8	3,6	45,5

Opierając się na pozostałych opiniach uzyskuje się idealny obraz, który jak uczy praktyka, nie jest osiągalny nawet tam, gdzie organizatorzy biorą aktywny udział w realizacji systemu informatycznego. Przyczyną tak poważnych rozbieżności ocen z rzeczywistością tkwią najprawdopodobniej w braku zainteresowania organizatorów w wykorzystaniu informatyki jako narzędzia doskonalenia organizacji. Potwierdzeniem tego jest bardzo wysoki odsetek respondentów wstrzymujących się od wyrażania opinii.

Z wypowiedzi informatyków wynika, że w 56% badanych przedsiębiorstwach w ogóle nie przeprowadzono analizy efektywności. W pozostałych przedsiębiorstwach analizę przeprowadzały najczęściej zakładowe ośrodki informatyki, zakładowe komórki analiz, postępu techniczno-organizacyjno-ekonomicznego i władze zwierzchnie.

Zakres analizy tylko w 4 przedsiębiorstwach obejmował ocenę operacyjną w odniesieniu do potrzeb systemu zarządzania oraz ocenę opłacalności. We wszystkich pozostałych

przypadkach (o ile udzielono informacji o zakresie analizy) analiza efektywności ograniczała się do porównania kosztów przetwarzania przy wykorzystaniu dotychczas stosowanej techniki i kosztów przetwarzania automatycznego.

Nieprzeprowadzenie analizy bodaj w tym minimalnym zakresie zmusza do postawienia pytania: na jakiej podstawie użytkownik decydował o przystąpieniu do realizacji systemu, o przyjęciu projektu i akceptacji rozwiązania po przeprowadzeniu próby eksploatacyjnej? Symptomatyczne, że wśród przedsiębiorstw, w których nie prowadzono analizy efektywności, niemal w komplecie znalazły się przedsiębiorstwa realizujące zamierzenie w ramach programu branżowego (resortowego), a więc takie, w których system informatyczny musi być wdrożony.

Należy tutaj przypomnieć, że we wszystkich fazach realizacji systemu informatycznego analiza efektywności powinna być stosowana jako narzędzie samokontroli przez zespół projektujący i jako narzędzie kontroli przez nadzorujących prace z ramienia użytkownika. Najbardziej miarodajne powinny być jednak wypowiedzi ankietowanych dyrektorów, którzy z racji zajmowanych stanowisk są — a przynajmniej powinni być — zainteresowani efektywnością wdrażanych systemów informatycznych.

Oceniając efekty wdrożenia systemów, dyrektorzy najczęściej wymieniają przyspieszenie informacji i zwiększenie ich zakresu (tabela 3). O wiele mniejsza część respondentów wymienia wśród uzyskanych efektów zmniejszenie zatrudnienia czy obniżenie kosztów tworzenia informacji. Możliwości doskonalenia organizacji wymieniają nieliczni.

Odpowiedzi na pytania dotyczące analizy efektywności wskazują na niewielkie zainteresowanie opłacalnością poniesionych nakładów. Można stąd przypuszczać, że poważna część ocen efektów nie była podbudowana dogłębnym rozpoznaniem opierała się na przypuszczeniach i życzeniach (tabela 6).

Tabela 6. Analiza efektywności zastosowań w wypowiedziach dyrektorów

Faza realizacji systemu	Żądanie analizy [proc.]			Wynik analizy [proc.]		
	Tak	Nie	Brak odpowiedzi	Pozytywny	Negatywny	Bez opinii
Prace wstępne	20,2	29,2	41,6	25,9	4,3	69,8
Próbna eksploatacja	19,3	33,5	47,2	16,5	4,7	78,8
Eksploatacja użytkownika	30,7	25,0	44,3	28,8	5,2	66,0

Bardzo mały odsetek dyrektorów żądających przeprowadzenia analizy efektywności nie może być wytłumaczony twierdzeniem, że nie wszyscy muszą interesować się efektywnością, gdyż nie wszyscy odpowiadają za komputeryzację. Szczegółowa analiza ankiet dowodzi, że skutecznością nowej techniki interesują się różni przedstawiciele kierownictwa w niektórych przedsiębiorstwach, a ponad dwudziestu żaden z dyrektorów nie odpowiedział na pierwsze z postawionych pytań pozytywnie.

STOSOWANE METODY PROJEKTOWANIA SYSTEMÓW INFORMATYCZNYCH

Ankieta miała również dostarczyć informacji na temat stosowanych metod realizacji systemów informatycznych: rodzajów oraz ocen ich przydatności i efektywności. Ponadto konfrontacja informacji o metodach projektowania z innymi charakterystykami systemów miała dać podstawy do oceny ich skuteczności w realizacji obiektowych systemów informatycznych. Uzyskane wypowiedzi nie przyniosły tylu informacji, ile oczekiwano, gdyż na pytanie o stosowaną metodę projektowania blisko połowa respondentów (42%) nie udzieliła odpowiedzi. Wśród wymienionych metod znalazły się własne metody diagnostyczne (20%), metody określone jako prognostyczne (18%), metody nieokreślone (13%), metoda OBRI (7%).

Margines respondentów nie udzielających odpowiedzi na temat podstawowego warsztatu pracy specjalistów odpowiadających za realizację kosztownego zamierzenia wydaje się zbyt szeroki. Wśród pozostałych ankiet znalazły się

również wypowiedzi niekompletne, na podstawie których nie można było inaczej sklasyfikować metody niż jako nieokreśloną.

W pozostałych wypowiedziach dominują metody własne wypracowane w ośrodkach, które od lat projektują i wdrażają systemy ewidencyjne oraz grupa metod określonych jako prognostyczne. Wśród tych ostatnich przeważają zdecydowanie metody zakupione w firmach komputerowych z krajów zachodniej Europy i USA.

Wśród użytkowników i wielu informatyków pokutuje niesłuszny pogląd, że systemy informatyczne można realizować w oparciu o małe ośrodki zakładowe, zatrudniające pracowników bez kwalifikacji projektantów. Zadaniem takich ośrodków jest tylko organizacja wdrożeń i późniejszej eksploatacji systemów projektowanych przez ośrodki branżowe lub ZETO. Właśnie z tej grupy przedsiębiorstw pochodziły ankiety niekompletnie opracowane przez informatyków i ankiety zawierające najbardziej krytyczne opinie kierownictwa.

Tabela 7. Ocena stosowanych metod projektowania [proc.]

Cecha metody	Tak	Nie	Brak opinii
Kompletność	32,7	12,7	54,6
Spójność	30,9	5,5	63,6
Efektywność	45,5	3,6	50,9
Uniwersalność	32,7	10,9	56,4
Dostosowanie do potrzeb	65,5	1,8	32,7
Łatwość przyswojenia	43,6	9,1	47,3
Skomplikowanie w stosowaniu	5,5	29,1	65,4

Bardzo wysoki odsetek respondentów wstrzymał się od oceny stosowanej metody projektowania (tabela 7), co uniemożliwiło wnioskowanie o przydatności określonych metod projektowania w konkretnych warunkach.

Oceny były często podawane również przez tych respondentów, którzy nie udzielili informacji o samej metodzie. Znamienne, że podkreślali oni takie cechy metody, jak dostosowanie do potrzeb, efektywność i łatwość przyswojenia.

56% respondentów wyraziło opinię, że metoda projektowania nie powinna zawęzać się do problematyki zbierania, przetwarzania i emisji informacji, 11% stanowili zwolennicy takiego ograniczenia, a 33% nie miało wyrobionej opinii w tej kwestii.

Informatycy przedstawili swoje wymagania, jakie powinna spełniać najlepsza w stosunku do ich potrzeb metoda projektowania: uwzględnienie potrzeb zarządzania (22%), skuteczność (20%), podejście systemowe (13%), uniwersalność (11%), elastyczność (9%), niska pracochłonność — wspomaganie komputerowe (7%). Opinii nie wyraziło 44% respondentów.

Próba analizy zależności między stosowanymi metodami projektowania a takimi charakterystykami zastosowań, jak strategia i koncepcja komputeryzacji, struktura systemu, oceny itd. przyniosła wynik negatywny. Brak współzależności świadczy o niedostosowaniu metod do obranej strategii i koncepcji realizacji systemu informatycznego, o poważnej dowolności w stosowaniu metod i częstym niewykorzystywaniu ich możliwości.

UWAGI KOŃCOWE

Wykorzystanie techniki automatycznej do przetwarzania danych gospodarczych w badanej grupie przedsiębiorstw prawie nie wykracza poza systemy ewidencyjne. Rozwiązania te realizowane są wycinkowo i w niewielu przypadkach są podporządkowane ogólnej koncepcji. Brak takiej koncepcji, stwierdzony w zdecydowanej większości przypadków, powoduje, że zastosowania okazują się mało efektywne i nie stwarzają rzeczywistych możliwości doskonalenia systemu zarządzania.

Ekstensywny charakter zastosowań jest w dużej mierze skutkiem braku przygotowania użytkowników. Rodzaj potrzeb informacyjnych badanej grupy dyrektorów tłumaczy

niski stopień zaawansowania wielu systemów informatycznych, które nazbyt często są rozwiązaniami doraźnymi, nie obejmującymi problematyki przetwarzania całościowo i nie uwzględniającymi przyszłych potrzeb. Użytkownicy niewiادی możliwości techniki komputerowej i pełni rezerwy wobec informatyków zlecają realizację systemów wycinkowych, pozornie tanich, gwarantujących doraźne efekty w wyniku szybkiego wdrożenia.

Przyczyn niedostatecznej wiedzy informatycznej użytkowników należy doszukiwać się w niewłaściwie przeprowadzonym szkoleniu kadry kierowniczej przedsiębiorstw. Zebrane materiały i obserwacja praktyki wykazuje, że zwykle szkolenie takie zamiast prezentować możliwości efektywnego wykorzystania komputera (najlepiej na przykładzie eksploatowanych rozwiązań pilotowych) obejmuje informacje o tym, czym zajmują się informatycy.

Rozwój ekstensywnych zastosowań musi prowadzić do konfliktów między informatykami a użytkownikami systemu, spowodowanych z jednej strony wzrastającym poziomem wiedzy informacyjnej użytkowników i ich wymagań a ograniczonymi możliwościami rozbudowy i doskonalenia w trakcie eksploatacji wycinkowych systemów informatycznych — z drugiej.

Teza ta nie została w pełni potwierdzona w przeprowadzonych badaniach, lecz w przedsiębiorstwach, w których wdrożono kilka systemów wycinkowych, członkowie kierownictwa są krytycznie nastawieni do systemów „projektowanych dla rachunkowości”, a nie na potrzeby zarządzających. Krytykowany jest przede wszystkim brak elastyczności i niemożliwość wprowadzania zmian i modyfikacji. W wyniku takiego rozwoju zastosowań nie można oczekiwać doskonalenia systemu zarządzania, co powinno być celem komputeryzacji przetwarzania danych.

Aktualnie funkcjonujący system zarządzania gospodarką narodową nie stawia wysokich wymagań systemom informacyjnym przedsiębiorstw, w których przetwarzanie podporządkowane jest ewidencji i sprawozdawczości. Należy jednak pamiętać, że zmiany w systemie zarządzania gospodarką, dające większą samodzielność wielkim organizacjom gospodarczym i przedsiębiorstwom, spowodują konieczność doskonalenia systemów informacyjnych w zakresie obsługi procesów decyzyjnych wewnątrz organizacji. Zmiany te mogą okazać się bardzo kosztowne dla tych użytkowników systemów, którzy poprzestają na rozwiązaniach doraźnych, zaspokajających co najwyżej dzisiejsze potrzeby. Wbrew utartym opiniom nie ma bowiem bezpośredniego przejścia od systemów ewidencyjno-rozliczeniowych do systemów informacyjno-decyzyjnych. Dlatego bardzo kosztownym błędem może okazać się dla wielu przedsiębiorstw koncepcja polegająca na projektowaniu i wdrażaniu „tanich” systemów wycinkowych bez jakiegokolwiek planu i myśli przewodniej.

Realizacja wycinkowych systemów informatycznych sprzyja istnieniu wyspecjalizowanej służby informatycznej, która w zależności od przynależności organizacyjnej skłonna jest preferować potrzeby określonych komórek organizacyjnych. Systemy projektowane i wdrażane powinny być tymczasem podporządkowane ogólnej koncepcji rozwoju organizacji jako jej część integralną.

Koncepcja — jako baza komputeryzacji — powinna powstać w wyniku współpracy specjalistów z dziedziny organizacji i informatyków przy aktywnym udziale kierownictwa. Opracowaniu i realizacji takiej ogólnej koncepcji sprzyja integracja służby organizacji z zakładowymi ośrodkami informatyki, a doświadczenia przedsiębiorstw, które dokonały połączenia, w pełni potwierdzają słuszność takiego rozwiązania organizacyjnego. Realizacja tak zaplanowanych zamierzeń wiąże się oczywiście z koniecznością zastosowania zaawansowanych metod projektowania.

Już obecnie pilną koniecznością wydaje się przeprowadzenie analizy efektywności wielu eksploatowanych systemów i podjęcie na tej podstawie decyzji o sposobie dalszej realizacji komputeryzacji przetwarzania. Wyniki uzyskane z przeprowadzonych badań z wielu względów nie upoważniają do formułowania ogólnych wniosków o zaawansowaniu zastosowań informatyki ani o ich przydatności do zarządzania tak przedsiębiorstwami, jak i całą gospodarką narodową. Na ich podstawie można jednak zaobserwować pewne negatywne zjawiska, którym należy przeciwdziałać w imię wspólnych interesów tak użytkowników, jak i informatyków.

Informatyka w projektowaniu budowlanym (na tle konferencji INFOPRO 79 i PArc 79)

W 1977 roku odbyła się I Konferencja „Informatyka w projektowaniu budownictwa INFOPRO 77” w Kudowie (patrz INFORMATYKA nr 8/77). Konferencja INFOPRO 79 odbyła się w dniach 27—29 maja br. również w Kudowie. Organizatorami, jak i poprzednio, byli: Komisja Koordynacji Ogólnobrazowej Projektowania, Centralny Ośrodek Badawczo-Projektowy Budownictwa Przemysłowego BISTYP, Komisja Informatyki Zarządu Głównego Związku Inżynierów i Techników Budownictwa i Oddział Wrocławski PZITB.

Trzydniowa konferencja, w której wzięło udział 250 osób z całego kraju, składała się z 10 sesji tematycznych. Konferencji towarzyszyły wystawa, pokaz filmów, sprzedaż wydawnictw itp. Imperaza została bardzo pozytywnie oceniona przez uczestników.

Dwuletni okres dzielący obydwie konferencje, a także wygłaszane referaty i przeprowadzane dyskusje pozwalają na ocenę dokonanego postępu i określenie pożądanych kierunków dalszego rozwoju.

SPRZĘT KOMPUTEROWY

Na poprzedniej konferencji przedmiotem szczególnej troski był bardzo słaby postęp w dziedzinie wyposażenia komputerowego biur projektów i brak wyraźnych perspektyw w tej dziedzinie. Aczkolwiek trudno jeszcze mówić o rozwiązaniu tych problemów, to jednak z zadowoleniem należy powitać dostawy krajowych minikomputerów MERA 400, które od 1978 roku są instalowane w biurach projektów. Jest to ten rodzaj sprzętu komputerowego, który dla wielu biur projektów wydaje się najbardziej odpowiedni. Dlatego też — pomimo różnych początkowych trudności nabycia, wdrażania i użytkowania — możliwość instalowania MERY 400 należy uznać za fakt bardzo pozytywny.

W ostatnich dwóch latach niektóre biura projektów zostały wyposażone w duże komputery krajowe, np. Bydgoskie BPBBP zakupiło RIADA 32, a BIPROMASZ — ODRĘ 1305. Duża moc obliczeniowa komputerów w oparciu o silne jednostki projektowe tworzy w tych biurach ważne ośrodki obliczeniowe o zasięgu regionalnym lub branżowym.

Rozwój wyposażenia komputerowego dla projektowania następuje wyłącznie w oparciu o produkcję krajową, przy czym produkcja ta na razie nie obejmuje urządzeń wejścia i wyjścia graficznego.

Istnieją pewne możliwości zakupu sprzętu komputerowego drogą importu z krajów socjalistycznych. Dotyczy to zwłaszcza autokreślarek DIGIGRAF produkcji CSRS, których niewielką liczbę zainstalowano w Polsce już w br. a także minikomputerów Jednolitego Systemu serii SM. Autokreślarki DIGIGRAF odpowiadają swymi parametrami potrzebom naszych biur projektów i mogą być dostosowane do współpracy z systemem MERA 400, co pozwala mieć nadzieję na rozwój grafiki komputerowej w krajowych biurach projektów.

PROGRAMOWANIE I STOSOWANIE

W dziedzinie oprogramowania najważniejszym osiągnięciem ostatniego dwulecia jest jego uporządkowanie. Nastąpiło całkowite ujednolicenie języków programowania (FORTRAN i BASIC), a także określone zostały wymagania w dziedzinie programowania, dotyczące budowy modułowej i wielkości modułów, faz opracowywania programów, dokumentacji itp.

Zamierzone tworzenie biblioteki programów do powszechnego użytku jest konsekwentnie realizowane, przy czym programy te kompletowane są w pierwszej kolejności w wersji ODRY 1305, a następnie adaptowane na RIADA 32. Od początku br. podjęto również opracowanie wersji tych programów na MERĘ 400.

Najbardziej popularne spośród programów stosowanych na ODRZE 1204 są obecnie opracowywane w FORTRANIE na ODRĘ 1305. Przy okazji wprowadzane są z reguły udoskonalenia algorytmów oraz formy danych wejściowych i wyników. Powstaje więc nowa, druga generacja tych programów.

W dziedzinie konstrukcji budowlanych najpopularniejszymi programami są w dalszym ciągu: pakiet PROBUS (Gliwickie BPBP) i program WB-21 (Wrocławskie BPBBR), które obecnie powstają w wersji II generacji na ODRĘ 1305, a także pakiet BRDA (Bydgoskie BPBBP) i program PROCHEM („Prochem”). Liczne zastosowania ma także system STRAINS (IDIM-PW), użytkowany w wielu ośrodkach dysponujących ODRĄ 1305. Ukończony w 1978 r. system AOS (WPIPB ETOB), obejmujący wersję AOS-KOS dla wysokich konstrukcji prętowych i AOS-WILGA dla płyt i tarcz, wchodzi obecnie do eksploatacji.

Nowym systemem w tej dziedzinie jest również TAPP (zapoczątkowany w ETOB, a obecnie realizowany przez BISTYP), dotyczący konstrukcji prętowych płaskich i przestrzennych, płytowych i tarczowych, z wynikami prezentowanymi w postaci graficznej. Dalszy rozwój tego systemu przewiduje również wejście graficzne.

W ostatnich 2 latach powstało też kilka programów obejmujących wymiarowanie konstrukcji żelbetonowych według zasad nowej normy tych konstrukcji.

Programy dotyczące statyki i wymiarowania konstrukcji budowlanych w zasadzie całkowicie zaspokajają istniejące potrzeby. Są to programy najliczniej stosowane, natomiast niewystarczający jest ciągły zasób programów dotyczących dynamiki konstrukcji budowlanych.

Programy dla instalacji przemysłowych i sanitarnych powstają głównie w dwóch ośrodkach, tj. BISTYP i Bydgoskim BPBBP.

Program ANKO należy nadal do najpopularniejszych i najczęściej stosowanych i ma wersje na 5 typów komputerów, przy czym autor programu zapowiada nowy, udoskonalony program dla instalacji c.o. Program VEN-BIS (BISTYP) dla instalacji wentylacyjnych ma coraz szerszy krąg użytkowników i powstają jego wersje na następne typy komputerów. Również opracowywany obecnie, a częściowo już użytkowany, AQUAPOL dla sieci wodociągowych zapowiada się jako program dobrze trafiony w potrzeby projektantów. Pakiet programów instalacyjnych „BYDGOSZCZ” (Bydgoskie BPBBP) jest także bardzo szeroko stosowany, stale doskonalony i rozbudowywany. W ostatnich latach powstały również jego wersje na RIADA 32 i ODRĘ 1305.

Zmiany nastąpiły w programach do obliczania zanieczyszczenia atmosfery. Bardzo liczne i różnorodne programy stały się nieaktualne w związku z wprowadzeniem nowej metodyki tych obliczeń. Na ich miejsce powstało kilka nowych programów opartych na wspólnych jednolitych podstawach metodycznych.

Projektowanie instalacji elektrycznych jest dziedziną, w której liczba i jakość programów ciągle jest niezadowalająca, stąd też stosowanie metod komputerowych jest niewielkie.

W ostatnim dwuleciu powstały podstawy dwóch systemów projektowania instalacji elektrycznych. Są to systemy opracowane w AGH (współpraca z BISTYPEM) oraz w Politechnice Wrocławskiej (współpraca z BIPROMASZEM). Można mieć nadzieję, że systemy te — wraz z kilkoma mniejszymi programami — zapoczątkują wyraźny postęp w tej dziedzinie.

Projektowanie rozplanowań i koordynacji funkcji, zwłaszcza w projektowaniu architektonicznym, ma ciągle niezbyt liczne opracowania i programy.

Największym przedsięwzięciem jest system ASY (Politechnika Białostocka) przeznaczony dla projektowania rozplanowania budynków mieszkalnych. System ten jest rozszerzany.

Dla koordynacji funkcji obiektów użyteczności publicznej opracowano w 1978 r. oryginalną metodę i program obliczeniowy (Politechnika Szczecińska).

Można ocenić, że zastosowania do projektowania funkcji i rozplanowań są dopiero w początkowym stadium rozwoju.

Ostatnie dwa lata potwierdziły w całej pełni szczególnie wysoką efektywność systemów automatyzacji projektowania, obejmujących obliczenia i kreślenie automatyczne, a przeznaczonych dla projektowania obiektów w ramach systemów konstrukcyjno-montażowych budownictwa. Systemy: ASTROFF (dla hal „FF” i P-70) i PASTOR (dla hal R) w BISTYPIE są stale wielokrotnie i bardzo chętnie stosowane przez wielu użytkowników.

Dlatego też równoległe z nowym systemem konstrukcyjnym dla budynków przemysłowych JSB powstaje związany z nim system automatycznego projektowania.

Podobnie system „MOSTOSTAL” (dla hal typu MOSTOSTAL — OTP) ma bardzo liczne zastosowanie i znacznie zmniejsza nakłady na projektowanie oraz skracca cykl projektowania.

Pojawił się też nowy kierunek kompleksowej automatyzacji w postaci systemu KORAB (COBPBO), stanowiący powiązanie pomiędzy projektowaniem budynków mieszkalnych i planowaniem produkcji w fabryce domów wytwarzającej elementy dla projektowanych budynków. Jest to szczególnie ważny kierunek rozwojowy automatyzacji projektowania.

W ostatnim dwuleciu wiele prac poświęcono zastosowaniom komputerowych urządzeń graficznych. Wiele z wymienionych i nie wymienionych programów i systemów rozszerzono o wejścia graficzne. Praktyczne zastosowanie automatycznego kreślenia w skali całego projektowania w Polsce jest na razie niewielkie, ponieważ niewiele biur projektów ma dostęp do autokreślarek. Na podstawie dotychczasowych prac można stwierdzić niewątpliwą ogromną przydatność autokreślarek w biurach projektów. Urządzenia te w przyszłości muszą stać się podstawowym — obok komputerów — urządzeniem automatyzacji projektowania.

Jeszcze węższy zasięg mają dotychczasowe prace z urządzeniami wejścia graficznego, tj. monitory ekranowe do prac graficznych (grafoskopy) i urządzenia do cyfrowego kodowania rysunków (digitizery). W BISTYPIE wykonano szereg eksperymentalnych programów z wykorzystaniem tych urządzeń.

Wbrew początkowym opiniom, korzystanie z grafoskopu jako urządzenia wejścia okazało się mało przydatne. Natomiast jest on bardzo cennym urządzeniem do bieżącej kontroli prac graficznych. Tak np. przed wykreśleniem rysunku na autokreślance wygodnie jest obejrzeć go na ekranie, co umożliwi szybką kontrolę i modyfikację rysunku.

Bardzo cennym urządzeniem okazał się czytnik rysunków (digitizer). Najprostszym jego zastosowaniem jest wprowadzenie do pamięci komputera danych graficznych (w postaci zbioru współrzędnych) jako podstawy do programu obliczeniowego, co w wielu przypadkach jest znacznym uproszczeniem w porównaniu do wprowadzania danych liczbowych.

Innym prostym zastosowaniem jest zarejestrowanie np. rzutu terenu po to, aby dane wyjściowe z programu obliczeniowego mogły być wykreślone na tym rzucie. Takie rozwiązanie zastosowano dla podkładów geodezyjnych, na których wykreśla się izolinie opadu pyłów.

Właściwe zastosowanie digitizera to projektowanie interaktywne. Z zarejestrowanych rysunków fragmentów i elementów można za pomocą digitizera i autokreślarki tworzyć wielowariantowe rozwiązania projektowe. Doświadczalny system tego rodzaju tworzony jest w BISTYPIE

dla interaktywnego projektowania obiektów w nowo powstającym systemie konstrukcyjnym MG (małogabarytowym).

W programach typu „przetwarzanie danych” dla celów projektowania nie nastąpił większy postęp. Stosowane są tu w pewnym zakresie programy dla kosztorysowania, a także dla zarządzania biurami projektów. Nie rozwinęły się na większą skalę zautomatyzowane systemy informacji naukowej i technicznej. Wyjątkiem jest system informacji o pracach badawczych w budownictwie INBAD (BISTYP), który jako resortowy system informacyjny jest praktycznie eksploatowany.

Omawiając zastosowania informatyki należy oczywiście pamiętać o korzystaniu z mniejszych urządzeń liczących, a w szczególności o stosowaniu minikomputerów WANG 2200 i COMPUCORP 424, które w biurach projektów odgrywają bardzo ważną rolę. W ostatnich latach nastąpił dalszy znaczny wzrost liczby i jakości programów opracowanych na te urządzenia i to w zakresie wielu specjalności projektowania.

ORGANIZACJA ZASTOSOWAŃ I ROZWOJU

Rozwój zastosowań informatyki w projektowaniu i współpraca między zainteresowanymi jednostkami organizacyjnie opiera się na Specjalistycznym Ośrodku Koordynacji Projektowania ds. ETO, będący organem KKO (Koordinacji Ogólnobranżowej Projektowania w Budownictwie). Ośrodek ten prowadzony przez BISTYP, opiekuje się ponadto klubami użytkowników minikomputerów MERA 400 i COMPUCORP oraz współpracuje z klubem użytkowników WANG 2200. Organizacja taka umożliwia integrację całego środowiska informatyków w biurach projektów.

W ostatnich latach zrealizowano szereg prac porządkujących działania w zakresie zastosowań i ułatwiających dalszy ich rozwój.

● Przy BISTYPIE zorganizowano bibliotekę programów i systemów, obejmującą całą dziedzinę projektowania. Biblioteka ta jest centralnym ogniwem informacyjnym w sprawach programów na potrzeby projektowania, a w pewnym zakresie również dostawcą dokumentacji tego typu programów.

● Wydano kompletne wykazy aktualnie dostępnych programów i systemów automatyzacji projektowania, a mianowicie:

— wykaz programów dla komputerów (opracowanie BISTYPU, II wydanie 1978 r.)

— wykaz programów dla minikomputera WANG 2200 (opracowanie METROPROJEKTU)

— wykaz programów na kalkulator programowany COMPUCORP 424 (opracowanie klubu użytkowników).

● Opracowano (MIASTOPROJEKT-Łódź 2) propozycje dla wyższych uczelni w sprawie programów nauczania na studiach poddyplomowych w zakresie automatyzacji projektowania. Propozycje te zostały przekazane uczelniom, które prowadzą takie studia oraz do szeregu innych uczelni z prośbą o rozważenie możliwości ich uruchomienia.

● Opracowano (BISTYP) programy nauczania dla kilku rodzajów kursów szkoleniowych informatyki w biurach projektów. Programy te zostały przekazane do terenowych ośrodków koordynacji projektowania jako materiał pomocniczy przy organizowaniu szkoleń. Opracowanie to powinno ułatwić układanie programów oraz przyczynić się do ujednolicenia i podniesienia efektywności szkoleń.

● Rozpoczęto wydawanie materiałów pomocniczych na potrzeby wdrażania informatyki w projektowaniu. Wydawnictwa te w ramach serii „Metody i techniki projektowania” (BISTYP) będą obejmować wytyczne metodyczne i opisy programów. Warto przy okazji zaznaczyć, że w ostatnim dwuleciu ukazało się kilka książek dotyczących automatyzacji projektowania, głównie w wydawnictwie „Arkady”.

● Do powstałego w 1978 r., a obowiązującego od br. Cennika Prac Projektowych wprowadzono odrębny rozdział dotyczący prac rozwojowych i stosowania informatyki w projektowaniu. Ta pierwsza — zapewne niedoskonała — próba cennikowego ujęcia tych zagadnień powinna ułatwić pracę zespołów obliczeniowych w biurach projektów.

● Opracowano (BISTYP) propozycje dotyczące wrowadzenia atestacji programów. Proponowane atestowanie programów ma na celu zwiększenie zaufania do prawidłowości stosowanych programów, a także wyeliminowanie z obiegu programów nieefektywnych lub błędnych.

KIERUNKI ROZWOJU

Dalszy rozwój informatyki będzie opierał się na coraz liczniejszych instalacjach minikomputerów, głównie MERA 400. Wyposażenie w minikomputery powinno objąć wszystkie większe i średnie biura projektów. Każdą jednostkę dysponującą minikomputerem 1—2 lata po jego zainstalowaniu należałoby wyposażyć w autokreślarkę, aby powszechne stało się stosowanie wyjść graficznych.

Większe biura projektów, a zwłaszcza jednostki wiodące w koordynacji terenowej i branżowej, powinny być wyposażone stopniowo w duże komputery, tworząc regionalne i branżowe ośrodki wiodące. Ośrodki te powinny stanowić bazę obliczeniową dla największych obliczeń wszystkich biur projektów w regionie lub branży. Komputery tego typu należy eksploatować jako systemy wielodostępne — w pierwszym okresie przez instalowanie terminali w pracowniach własnego biura, a następnie rozszerzając je przez przyłączanie jako terminali minikomputerów z innych biur projektów.

W miarę powstawania możliwości sprzętowych powinny być instalowane urządzenia wejścia graficznego i tworzone systemy interaktywnego projektowania.

Obserwując od wielu lat powstawanie nowych programów dla celów projektowania, a następnie ich praktyczne użytkowanie, można stwierdzić, że istniały w tej dziedzinie dotychczas dwa etapy rozwojowe.

W pierwszym etapie programy powstawały przez prostą zamianę czynności ręcznych przez pracę komputera, tj. przy zastosowaniu tych samych metod i algorytmów, przy czym programy obejmowały na ogół krótkie odcinki pracy projektanckiej, a od strony informatycznej były dość prymitywne i niejedolite.

W drugim etapie, który w odniesieniu do większości zagadnień trwa jeszcze obecnie, nastąpił rozwój programów głównie pod względem informatycznym oraz przystosowania do potrzeb użytkowników. Podniósł się wyraźnie poziom programowania, programy stały się bardziej efektywne, a zamiast pojedynczych programów powstały ich pakiety i systemy. Ustaliła się forma dokumentacji programów, ujednolicono języki programowania i typy stosowanych komputerów, a część programów uzyskała wyjście graficzne.

Od strony użytkowej rozwój ten polega na obejmowaniu programami zagadnień większych i w sposób bardziej kompleksowy. W niektórych przypadkach przybiera on formę kompletnej automatyzacji projektowania wybranych konstrukcji czy obiektów.

Nastąpiło również udoskonalenie sposobów przygotowania danych oraz formy wyników, powodując znaczne zmniejszenie pracochłonności i przyspieszenie projektowania.

Drugi — niemniej ważny — cel stosowania informatyki w projektowaniu, jakim jest podniesienie efektywności rozwiązań projektowych, osiągnięto w bardzo niewielkim stopniu. Programy opierają się głównie na uproszczonych metodach i algorytmach, a zwłaszcza na wzorach i zaleceniach normatywnych, które są najczęściej przybliżeniami wprowadzonymi ze względu na trudności w ręcznym liczeniu. Opieranie obliczeń komputerowych na tych algorytmach nie może więc przynieść żadnych nowych efektów ekonomicznych w projektowanych obiektach.

Tak więc nowy, trzeci etap rozwoju programów powinien polegać na wprowadzaniu nowych założeń i metod obliczeniowych, które przez eliminację zbędnych przybliżeń i uproszczeń powinny doprowadzić do podniesienia efektywności projektowanych obiektów.

W każdej specjalności projektowania trzeba będzie sięgnąć do źródeł teoretycznych, naukowych w celu opracowania nowych algorytmów. Programy powinny objąć różne skomplikowane przypadki rozwiązań projektowych występujące w praktyce, a automatyczna optymalizacja parametrów rozwiązań powinna być powszechną zasadą. Dla tego też dalszy rozwój programów będzie wymagał szerokiej współpracy z placówkami naukowymi i wyższymi uczelniami. Niezbędne będzie również pełne wdrożenie zasady atestowania programów. Obszary stosowania informatyki w biurach projektów powinny być dalej rozszerzane, a korzystanie z obliczeń komputerowych powinno być coraz bardziej powszechne. W miarę powiększania się

możliwości sprzętowych coraz powszechniej powinno być stosowane wejście i wyjście graficzne. Trzeba będzie dążyć do znacznego rozszerzania zastosowań w zakresie automatycznego projektowania funkcji i rozplanowań, a także w dziedzinie informacji technicznej, ze stosowaniem techniki mikrofilmowej włącznie.

KONFRONTACJA Z PRACAMI ZAGRANICZNYMI

Krajowe prace w dziedzinie komputerowego wspomaganie projektowania można dodatkowo ocenić, porównując je z pracami prezentowanymi na międzynarodowej imprezie o bardzo podobnej tematyce.

W dniach 7—10 maja br. odbyła się w Berlinie Zachodnim konferencja międzynarodowa „PARC-79”, poświęcona zastosowaniom komputerów w architekturze, budownictwie i planowaniu miast. Konferencja planowana na około 300 osób zgromadziła ponad 500 uczestników z 30 krajów, co świadczy o dużym zainteresowaniu jej tematyką, jak również o potrzebie informacji i wymiany doświadczeń w tej dziedzinie.

Organizatorami konferencji było kilkanaście instytucji z różnych krajów, wśród których znalazły się stowarzyszenia techniczne i naukowe, a także ministerstwa odpowiedzialne za rozwój budownictwa i urbanistyki.

Ze strony Polski do grona organizatorów należał Polski Związek Inżynierów i Techników Budownictwa, reprezentowany przez Główną Komisję Informatyki Zarządu Głównego.

W czasie 4-dniowej konferencji wygłoszono ponad 70 referatów, zgrupowanych w następujące sesje:

- Aktualny stan rozwoju
- Zastosowania praktyczne
- Planowanie miast
- Badania i przyszły rozwój
- Zagadnienia wybrane.

Nurt badawczy reprezentowali najsilniej prelegenci z W. Brytanii, wśród których byli znani w tej dziedzinie uczeni i specjaliści, jak np. prof. T. W. Maver (Uniwersytet Strathclyde w Glasgow), prof. H. Frazer (Uniwersytet w Belfaście), prof. Purcell (Royal College of Art), prof. A. Bijl (Uniwersytet w Edynburgu), E. Hoskins (ARC-Cambridge), a także duża grupa prelegentów z RFN, a m.in. prof. K. Zuse (znany konstruktor komputerów), prof. H. Schwarz (Politechnika w Darmstadt) oraz prelegenci z USA (prof. N. Negroponte — MIT), Australii (prof. J. S. Gero z Uniwersytetu w Sydney), Francji, Holandii, Szwecji, Włoch, Polski i innych krajów.

Nurt zastosowań praktycznych najliczniej był reprezentowany przez prelegentów niemieckich, chociaż i w tej grupie referentów występowali prelegenci z kilkunastu krajów.

Niektóre ośrodki badawcze i zastosowań praktycznych występowały z większymi grupowymi referatami i pokazami. Tak np. prezentowano system ABACUS (Uniwersytet w Glasgow), system BOS (Applied Research of Cambridge), system RUCAPS i inne.

W referatach tych omawiane były liczne przykłady różnych systemów komputerowych, a także szereg prac stanowiących teoretyczną podbudowę automatyzacji projektowania i przetwarzania informacji w budownictwie. Trudno byłoby omówić treść poszczególnych referatów, można jednak scharakteryzować ogólnie istniejące kierunki rozwojowe i aktualny stan zastosowań.

Przede wszystkim zwraca uwagę powszechne korzystanie z grafiki komputerowej, tj. stosowanie wejść i wyjść graficznych oraz urządzeń do interaktywnego projektowania graficznego. Digitizery, autokreślarki i grafokopy stanowią podstawowe, niezbędne wyposażenie przy większości zastosowań komputerowych w architekturze i budownictwie. W zastosowaniach tych niezbędna jest nie tylko graficzna prezentacja wyników obliczeń, ale i niezależne od obliczeń komputerowe wspomaganie prac graficznych, zwłaszcza w oparciu o technikę „menu”.

Propagowane i oferowane są systemy minikomputerowe i terminale dla mniejszych przedsiębiorstw, których liczba w krajach Europy Zachodniej jest bardzo duża. Systemy te dotyczą zarówno obliczeń technicznych dla budownictwa, jak również szeregu obliczeń związanych z ekonomiką

budownictwa, tj. przygotowania ofert, opracowania kosztorysów, faktur, harmonogramów itp. Często są to systemy łączące rozwiązania techniczne i organizacyjno-ekonomiczne.

Należy stwierdzić, że interaktywne systemy graficzne dla projektowania architektonicznego, aczkolwiek licznie prezentowane są w większości przypadków nadal systemami dla bardzo wąskich zastosowań albo też systemami eksperymentalnymi lub badawczymi.

Poza referatami i dyskusjami nad ich treścią odbyły się także specjalne sesje dyskusyjne, zapoczątkowane wypowiedziami wprowadzającymi w temat.

Jedną z takich dyskusji, zatytułowana „Wprowadzenie komputerów do praktyki i kształcenia”, skoncentrowała się nad sposobem przygotowania kadr dla projektowania wspomaganego. Rozważano kto, kogo, jak i w jakim zakresie powinien kształcić, aby rozszerzyć zastosowanie komputerów. Problem jest trudny i pilny — konieczne jest przygotowanie kadry posiadającej zarówno wiedzę architektoniczną lub budowlaną, jak i znajomość teorii systemów, programowania i wielu różnych dyscyplin.

Inna dyskusja poświęcona była wymianie poglądów na temat zastosowań komputerów w budownictwie w Europie Zachodniej. Zwracano uwagę, że większe środki państwowe należy przeznaczyć na prace badawcze. Omawiano trudności wynikające ze słabej koordynacji prac różnych ośrodków i uczelni, w związku z czym prace są często niepotrzebnie powtarzane.

Oddzielna, dodatkowa grupa referatów, miała charakter krótkiego szkolenia w zakresie podstaw stosowania komputerów w budownictwie i przeznaczona była dla tych uczestników, którzy dotychczas praktycznie nie korzystali z komputerów w swojej pracy zawodowej. Referaty te cieszyły się dużym zainteresowaniem, co można przypisać bardzo starannemu przygotowaniu wykładów od strony ich ilustracji graficznej na wyświetlanych przezroczach i filmach.

Równolegle z konferencją odbyła się wystawa „Komputer na budowie” („Computer am Bau”), na której około 30 firm prezentowało sprzęt i systemy informatyczne dla

budownictwa. Były to głównie systemy przeznaczone dla mniejszych przedsiębiorstw architektonicznych i budowlanych, oparte na małych systemach komputerowych.

Udział w konferencji uczestników z Polski był bardzo żywy i wyraźnie widoczny na tle obrad. Przybyło 15 osób, z których 6 wygłosiło następujące referaty:

doc. dr hab. W. Gasparki (PAN): „Tendencje rozwojowe w badaniach nad projektowaniem”

prof. dr hab. Z. Leśniak (Politechnika Białostocka): „System komputerowy dla projektowania architektonicznego”
dr inż. A. Kociołek (Centralny Ośrodek Badawczo-Projektowy Budownictwa Ogólnego): „Projektowanie wspomaganie komputerowo w uprzemysłowionym budownictwie mieszkaniowym”

doc. dr hab. M. Ostrowska (Politechnika Szczecińska): „Zagadnienia systemowego projektowania architektonicznego”
mgr inż. A. Sambura (Gliwickie Biuro Projektów Budownictwa Przemysłowego): „Modelowanie matematyczne struktur hierarchicznych w projektowaniu”

dr inż. M. Robakiewicz (COBPBP BISTYP): „Zastosowanie komputerów w projektowaniu obiektów w ramach systemów konstrukcyjno-budowlanych w Polsce”.

Należy podkreślić, że referaty doc. Gasparskiego i dr. A. Kociołka zostały potraktowane jako główne referaty dwóch sesji.

Konferencję oceniono bardzo pozytywnie i w związku z dużym zainteresowaniem ma być powtarzana jako cykliczna impreza międzynarodowa co dwa lata.

Jak na tle konferencji berlińskiej można ocenić stan zastosowań komputerów w polskim projektowaniu budowlanym? Wydaje się, że poziom nasz nie ustępuje krajom zachodnim. Praktyczne zastosowania są może nawet bardziej rozwinięte, gdyż nasze biura projektów są organizacjami większymi i ściślej ze sobą współpracują, stąd też łatwiejsza jest koordynacja, współpraca i ogólne ukierunkowanie zastosowań. Również zastosowania grafiki komputerowej od strony badawczej i systemów pilotowych nie ustępują zagranicznym. W dziedzinie tej nie mamy jednak tak powszechnego stosowania ze względu na bardzo nieliczny sprzęt do prac graficznych. Stosunkowo mniej rozwinięte w Polsce są natomiast systemy łączące rozwiązania techniczne i organizacyjno-ekonomiczne.

Centrum Projektowania i Zastosowań Informatyki ZETO w Warszawie zakupi w trybie pilnym

procesor emc ODRA 1325

w stanie używalności lub wymagający remontu.

INFORMACJE PROSIMY KIEROWAĆ POD ADRESEM:

Centrum Projektowania i Zastosowań Informatyki ZETO, Dział Inwestycji

ul. 1-go Sierpnia 21, 02-134 Warszawa,

tel. 46-14-70

Programowanie w PROLOGU: informator kolejowy

PRZY OKIENKU INFORMACJI

- Czym mogę służyć?
- O której godzinie jest wieczorny ekspres do Olsztyna z Warszawy Zachodniej?
- Nie ma takiego pociągu. Czym jeszcze mogę służyć?
- Chcielibyśmy prosić o poinformowanie nas, kiedy odjeżdżają z Dworca Wschodniego pociągi do Augustowa?
- O jaki rodzaj pociągu Państwu chodzi?
- Chcielibyśmy jechać osobowym.
- O jakie porze chcieliby Państwo odjechać?
- Tak gdzieś druga-trzecia.
- Proszę inaczej sformułować odpowiedź.
- Interesuje nas pora popołudniowa.
- Osobowy do Augustowa o 14.55 z Dworca Wschodniego. Osobowy do Augustowa o 17.23 z Dworca Wschodniego. Czym jeszcze mogę służyć?
- Dziękujemy, to wszystko.
- Proszę o nas nie zapominać.
-
- Czym mogę służyć?
- Czy mogłabym się dowiedzieć, kiedy i na jaką stację przyjeżdża wrocławski?
- Nie rozumiem, proszę zapytać jeszcze raz.
- Na jaki dworzec i o której przyjeżdża pociąg pośpieszny z Wrocławia?
- O jaką porę chodzi?
- Obojętne.
- Pośpieszny z Wrocławia o 9.45 na Dworzec Centralny. Pośpieszny z Wrocławia o 10.01 na Dworzec Wschodni. Pośpieszny z Wrocławia o 23.53 na Dworzec Centralny. Pośpieszny z Wrocławia o 0.09 na Dworzec Wschodni. Czym jeszcze mogę służyć?
- Dziękuję, to wszystko.
- Proszę o nas nie zapominać.

Podobne dialogi (choć może nie zawsze tak uprzejmie) można usłyszeć na każdym warszawskim dworcu przy okienku informacji. Ale przytoczona rozmowa nie odbyła się naprawdę na żadnej kolejowej: przeprowadzono ją z komputerem, z programem konwersującym o rozkładzie jazdy. Mamy nadzieję, że Czytelnik wybaczy nam mistyfikację. Zapisaliśmy te dialogi tak, jak się to czyni w powieściach, tymczasem faktycznie dialogi z komputerem pisane są wielkimi literami bez znaków diakrytycznych, choć ze standardową interpunkcją polską (konwersacja toczy się na piśmie). Program prowadzący dialog został napisany w języku PROLOG (prezentowanym w numerze 4/79 INFORMATYKI [4]) i uruchomiony na maszynie ODRA 1305 w Warszawie. To tłumaczy „warszawocentryzm” powyższej konwersacji.

DLACZEGO NAPISALIŚMY TEN PROGRAM?

Napisanie programu było dla nas eksperymentem, zarówno informatycznym, jak ligwistycznym. Interesowały nas przede wszystkim możliwości i ograniczenia PROLOGU, jego przydatność do przetwarzania tekstów polskich. Po drugie, pragnęliśmy sprawdzić praktyczną użyteczność pewnej koncepcji opisu składniowego polszczyzny, jak również możliwości wzbogacenia tego opisu o składnik semantyczny. Po trzecie wreszcie — praca nad programem była dla nas źródłem nie ustającej satysfakcji, jak bywa udziałem, powiedzmy, miłośnika łamigłówek, któremu udaje się raz po raz dopasować nowy element układanki (nie chcemy jednak twierdzić, że pisaliśmy program wyłącznie z nastawieniem ludycznym...).

O tym, że PROLOG dobrze się nadaje do przetwarzania tekstów języka naturalnego, byliśmy w zasadzie przekonani [6]. Postanowiliśmy natomiast zbadać, czy nadaje się on również do organizowania większego programu konwersacyjnego, a więc programu, który nie tylko interpretuje wypowiedzi, ale i reaguje na nie, nie tylko analizuje pytania, ale syntetyzuje odpowiedzi.

Ze względu na ograniczenia pamięciowe implementacji PROLOGU na ODRZE przyjęliśmy, że program będzie miał charakter eksperymentalny. Zgodnie z oczekiwaniami okazało się możliwe przeprowadzenie szeregu interesujących doświadczeń, także ligwistycznych. Przede wszystkim jednak efekty naszej pracy utwierdziły w nas przekonanie, że sporządzenie nawet złożonego programu w PROLOGU wymaga stosunkowo niewielkich nakładów. Przygotowanie omawianego programu kosztowało bowiem niespełna 100 „osobogodzin” i około 50 godzin czasu zegarowego komputera (pod nadzorem systemu EXECUTIVE). Warto podkreślić zwłaszcza drugą liczbę: ODRA 1305 jest jednak dość powolna, PROLOG na ODRZE jest więc stosunkowo czasochłonny. Jesteśmy pewni, że każdy, kto próbował zatrudnić komputer do przetwarzania tekstów języka naturalnego, rozumie optymistyczną wymowę tych liczb!

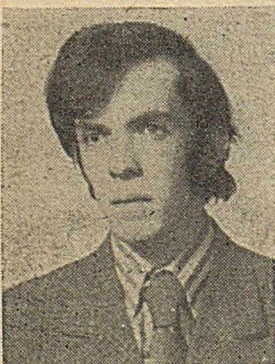
Korzyści ligwistyczne są zapewne jeszcze większe. Udało się mianowicie wykazać, że koncepcja opisu składniowego polszczyzny opracowana w ośrodku warszawskim i białostockim [5] nadaje się do zastosowań informatycznych, czego nie można powiedzieć ani o opisach składniowych z tradycyjnych gramatyk, ani nawet o znanych teoriach ostatnich dziesięcioleci [2]. Co więcej, udało się sporządzić program z semantyką, który nie tylko przypisuje analizowanemu wypowiedziom strukturę składniową, ale również interpretuje je znaczeniowo, bada ich niesprzeczność i pełność semantyczną. Nie koniec jednak na tym. Program zachowuje się językowo jak „żywy” użytkownik języka, to znaczy wyciąga wnioski z analizowanych wypowiedzi, reaguje na nie, odpowiada sensownie (i grzecznie, jak mało który „prawdziwy” informator...) na zadawane mu pytania. Tak więc semantyka nasza, choć ograniczona, jest semantyką, która działa: przypisanie wypowiedzi odpowiedniej interpretacji nie stanowi celu ostatecznego — to jedynie środek do celu, jakim jest dialog człowieka z komputerem.

O JĘZYKU DIALOGÓW DWORCOWYCH

Jak wiadomo, istnieje już sporo komputerowych systemów informacyjnych. Wymieńmy dla przykładu system informacyjny lotniczej działający do kilku lat w Stanach Zjednoczonych. Nasz program różni się od takich systemów — co stanowi o jego sile i słabości zarazem — tym, że kontakt klienta z komputerem ma przez cały czas charakter językowy. Kontakt ów polega na wymianie zdań, nie zaś

Dr. Stanisława SZPAKOWICZA prezentowaliśmy w tegorocznym numerze 4 INFORMATYKI, przy artykule „Wprowadzenie do PROLOGU”.

Dr Marek ŚWIDZIŃSKI jest adiunktem w Instytucie Języka Polskiego Uniwersytetu Warszawskiego. Studia polonistyczno-językoznawcze ukończył w roku 1971, specjalizując się w problematyce analizy semantycznej. W pracy doktorskiej, którą obronił w 1977 roku, zajmował się interpretacją semantyczną pewnej klasy zdań złożonych. Obecnie pracuje nad sformalizowanym opisem składniowym współczesnej polszczyzny pisanej.



np. na tym, że klient wypełnia określone rubryki formularza, komputer zaś drukuje odpowiedź wypełniając inne. Co więcej, nasz program wymaga od rozmówcy sformułowania pytania w poprawnej polszczyźnie, a więc oczekuje zdań gramatycznie nienaganych oraz — co równie ważne, pełnych, a także poprawnych sementycznie. Wymaganie to jest podyktowane względami teoretyczno-lingwistycznymi, nie praktycznymi. Gdyby chcieć naprawdę wprowadzić konwersację z komputerem na dworcu, należałoby dopuścić wszystkie wypowiedzi klienta na równych prawach, postulując, by program każdą z nich próbował interpretować, tzn. by w każdej doszukiwał się sensu. Nam jednak szło o rozwiązanie zadania ściśle lingwistycznego, mianowicie o zaimplementowanie pewnej kompletnej składni i semantyki (jakkolwiek jest to pełność najoczywistej względna). Nie trzeba dodawać, że zdanie poprawne gramatycznie to dla nas tyle, co dane przez reguły gramatyki, którą zastosowaliśmy. A zatem istnieje mnóstwo zdań poprawnych z punktu widzenia przeciętnego użytkownika języka, których nasz program nie potrafi zanalizować.

Naszą ambicją było objęcie jak największej liczby typów konstrukcji składniowych w ramach wspomnianych ograniczeń. Przewidzieliśmy więc wiele typów wypowiedzi, od najprostszych pytań w rodzaju „Kiedy jest osobowy do Elku?” po kwestie rozbudowane, ze zwrotami grzecznościowymi i pytaniem wyrażonym „całym zdaniem”, w rodzaju „Czy mogłybyśmy prosić o poinformowanie nas, o której godzinie odjeżdżają wieczorem z Warszawy Wschodniej pociągi pośpieszne do Białegostoku?”. Dopuszczalne są wypowiedzi o różnym szyku. A analizowane są wyrażenia synonimiczne („dworzec” — „stacja”, „odjeżdża” — „odchodzi” — „jest”, „pociąg ekspresowy” — „ekspresowy” — „ekspres” itp.). Program potrafi też analizować pytania wielokrotne („Kiedy i jakie pociągi...?”, „Skąd i o której godzinie...?”), a także pytania, w których nie zostały wyrażone życzenia klienta co do pory odjazdu lub przyjazdu, dworca czy rodzaju pociągu.

Program nie strukturyzuje wypowiedzi w terminach tradycyjnych części zdania. Szczegóły naszej hierarchizacji składniowej omawiamy w następnym punkcie. Wypowiedziom przypisuje się pewną reprezentację znaczeniową. Mówiąc intuicyjnie, program ustala, czego pytający chce się dowiedzieć, czy zadane pytanie jest niesprzeczne (np. „O której odchodzi osobowy ekspres do Ząbek?”, „Na jaki dworzec przyjeżdżają pociągi z Krakowa na Wschodni?”), czy ma ono sens pragmatyczny (np. „Kiedy odjeżdża pociąg nocny z Centralnego?” — nie wiadomo, dokąd), czy są w nim wyrażone wszystkie opcje klienta, a jeśli nie, to które nie są sformułowane, czy istnieją pociągi spełniające zadane warunki itd. W naszym programie używamy pewnego zestawu parametrów semantycznych, które opiszemy dalej. Przypisuje się je wypowiedziom na podstawie reguł gramatycznych, a także słownika.

Poszczególne hasła słownika mają na ogół określone wartości parametru semantycznego. W słowniku tym, tak jak w słowniku dowolnego języka naturalnego, zdarzają się hasła wieloznaczne. Na przykład słowo „jaki” (składnik wyrażenia „jaki pociąg”) występuje w słowniku dwukrotnie, w znaczeniu „o której godzinie” i „jaki typ pociągu”. Są też słowa równoznaczne, np. „ekspresowy” i „ekspres”. Dzięki temu program może zinterpretować pewne pytania na dwa sposoby („Jakie są pociągi do Kuluszek?” — pytanie o godzinę lub o typ pociągu), a dwie składniowo różne wypowiedzi — jako synonimiczne („O której przyjeżdża pociąg ekspresowy z Opola?” i „Kiedy jest ekspres z Opola?”).

Kwestie informatora (prośby o dodatkowe dane, wydruki podtrzymujące konwersację itp.) są w naszym programie ustalone raz na zawsze. Również udzielane informacje mają stały format.

PRZETWARZANIE WYPOWIEDZI PODRÓŻNEGO

Wypowiedź skierowana do programu to albo pytanie, albo dodatkowe dane, które pytający przekazuje informatorowi na jego życzenia. Poprawną reakcją na wypowiedź jest udzielenie odpowiedzi na pytanie, prośba o dalsze szczegóły lub odrzucenie wypowiedzi ze względu na zawarte w niej błędy. Aby właściwie zareagować na podany przez interlokutora tekst, należy go „zrozumieć”, to znaczy dokonać jego analizy. Analiza wypowiedzi obejmuje rozpoznanie jej struktury składniowej — cech poszczególnych wyrazów

i powiązań między nimi — oraz ustalenie jej znaczenia, w czym dopomaga znajomość owej struktury. W naszym programie analiza składni i znaczenia wypowiedzi przebiega równocześnie: składnia pozwala wyróżnić zwroty lub wyrazy istotne dla znaczenia całości (np. nazwę dworca lub określenie typu pociągu), a wyrazy decydujące tylko o poprawności gramatycznej (np. niektóre zwroty grzecznościowe) spostrzec, nie badając wszakże ich znaczenia.

Zbiór konstrukcji składniowych używanych podczas dialogu z informatorem opisujemy za pomocą gramatyki formalnej [1]; zapewnia to ścisłość niezbędną w pracy z komputerem. Składnię wypowiedzi wyraziliśmy w postaci gramatyki metamorficznej [3], którą stosujemy również do znajdowania znaczenia wypowiedzi. Gramatyki metamorficzne stanowią mocny formalizm dostępny bezpośrednio w języku PROLOG. Reguła gramatyczna definiuje konstrukcję jako ciąg tworzących ją konstrukcji niższego rzędu, w szczególności pojedynczych wyrazów; reguła może też określać warunki, w których się ją stosuje, oraz kontekst, w jakim dana konstrukcja ma wystąpić. Warunek jest to wywołanie procedury PROLOGOWEJ, która ustanawia odpowiednią relację między komponentami konstrukcji definiowanej przez daną regułę.

Symbole nieterminalne (oznaczające konstrukcje) i terminalne (oznaczające pojedyncze wyrazy) — w sensie składni PROLOGU — termami, których nazwy poprzedza się dla zachowania jednoznaczności znakami % i #. Symbole mogą więc mieć argumenty, które wykorzystuje się do wyrażania cech opisywanych konstrukcji. W naszym programie argumenty służą m.in. do ustalania znaczeń. Na przykład grupa opisująca pociąg może zawierać dane o typie pociągu i porze jego przybycia. Symbol nieterminalny odpowiadający tej grupie ma argument, który podczas rozpoznawania jej w jakiejś wypowiedzi otrzymuje wartość stosownego typu pociągu; istnieje też analogicznych argument dla pory dnia. Inne zastosowanie argumentów symboli nieterminalnych to opis zgodności charakterystyk gramatycznych poszczególnych komponentów konstrukcji składniowych. Możemy np. użyć sformułowania „jaki pociąg przyjeżdża...” i „jakie pociągi przyjeżdżają...”; w obu wersjach każdy z trzech wyrazów występuje w tej samej liczbie, tym samym rodzaju i przypadku. W regule gramatycznej fakt ten zapisujemy, używając trzykrotnie tego samego argumentu na oznaczenie np. liczby w symbolach odpowiadających kolejnym wyrazom.

Reguły gramatyki metamorficznej stanowią integralną część programu w PROLOGU. Zbiór reguł odnoszących się do tej samej konstrukcji składniowej tłumaczy się na procedurę służącą do analizowania owej konstrukcji. Analizę inicjuje się za pomocą procedury standardowej, której dostarcza się tekstu do rozpoznania i symbolu konstrukcji, jaką pragnie się w tym tekście rozpoznać. Symbol ten jest nie w pełni określony, a jego argumenty podlegają ustaleniu w toku analizy. Dzięki charakterystycznemu „dualizmowi” parametrów procedur w PROLOGU (wymienności ról parametrów „wejściowych” i „wyjściowych”) można też, poczynając od całkowicie określonego symbolu, dojść do tekstu; mamy wówczas do czynienia z syntezą konstrukcji.

Zasób konstrukcji występujących w języku dialogów dworcowych dobraliśmy na użytek doraźny. Nasza gramatyka nie jest gramatyką języka polskiego w ogóle, ale gramatyką jego specyficznego podzbioru, opartą na specyficznej hierarchizacji wypowiedzi. Dlatego to wprowadziliśmy na przykład szereg grup istotnych dla zrozumienia wypowiedzi, takich jak grupa „pociągowa”, „dworcowa” i „miastowa” (z punktu widzenia składni języka polskiego są one wszystkie grupami rzeczownikowymi); z drugiej strony nie zajmujemy się różnicowaniem czasowników, traktując je wyłącznie jako nośniki informacji o kierunku podróży („odjeżdża” — „przyjeżdża”) lub o chęci poinformowania się („zapytać”, „dowiedzieć się” itp.).

Nazwy symboli użyte przez nas są z tych samych przyczyn wysoce niekonwencjonalne, ponieważ kierowaliśmy się w ich wyborze tylko zaletami mnemonicznymi (KWERENDA, INTRODUKCJA, SRODEK, KONCOWKA, POMOC itp.). Korzystaliśmy tu z takiej samej swobody, jaką ma każdy, kto pisze w języku programowania wyższego poziomu. Tworzenie gramatyki metamorficznej jest zresztą pod pewnym względem niczym innym niż programowaniem. Budowanie zaś gramatyki (i analizatora!) „na zadany temat” jest — o czym pragnęliśmy się przekonać — zupełnie nietrudne dzięki zastosowaniu PROLOGU.

Równie doraźne są rozwiązania semantyczne. Informacje wydobywane z wypowiedzi to cel i kierunek podróży, typ pociągu, nazwa dworca i pora lub godzina odjazdu bądź przyjazdu. Program jest „ukierunkowany na Warszawę”: zakłada się, że chodzi o pociągi odjeżdżające z Warszawy lub przyjeżdżające do Warszawy. Osiągnęliśmy w ten sposób minimalizację liczby danych o każdym pociągu, co jest nie bez znaczenia w eksperymentalnym programie, który zajmuje się nie tylko semantyką. Pozbycie się powyższego założenia nie powinno być zresztą trudne.

Do reprezentowania znaczeń służą w naszej gramatyce parametry semantyczne — umowne stałe pozwalające stwierdzić zgodność znaczenia wyrazu z oczekiwaniami lub tożsamość znaczeniową kilku wyrazów (naturalnie tylko z punktu widzenia naszego programu). Tak np. stała T jest wspólnym parametrem semantycznym takich wyrazów określających typ pociągu, jak „osobowy”, „pośpieszny”, „ekspresowy”, „ekspres”. Nazwy stałych są wybrane zupełnie arbitralnie.

Przebieg przetwarzania pokażemy na przykładzie wypowiedzi „Chciałbym wiedzieć, o której przyjeżdża pociąg wieczorny z Bydgoszczy”. Chcemy w niej rozpoznać kwerendę, złożoną z introdukcji (zwrotu grzecznościowego) i właściwego pytania. Introdukcja „Chciałbym wiedzieć” jest poprawna składniowo, ale nie zawiera żadnych istotnych danych poza informacją o płci rozmówcy; po przetworzeniu stosowanego fragmentu tekstu informacja ta jest podstawiana na zmienną — argument symbolu INTRODUKCJA(*PLEC). Przetworzenie pytania „o której przyjeżdża pociąg wieczorny z Bydgoszczy” — o czym za chwilę — powoduje podstawienie na argument symbolu

PYTANIE(*STRUKT) termu odzwierciedlającego znaczenie wypowiedzi.

Oto jedna z reguł definiujących kwerendę — ta, którą zastosowaliśmy (pomijamy dla uproszczenia sposób rozpatrywania w naszym programie znaków presztankowych):

$$\%KWERENDA(*STRUKT,*PLEC) == \\ \%INTRODUKCJA(*PLEC) \%PYTANIE/*(*STRUKT).$$

Wartości zmiennych *PLEC i *STRUKT przekazuje się — po zakończeniu analizy — modułowi sterującemu konwersacją.

$$\%PYTANIE(Q(*W,*M,*K,*D,*G,*P,*T) == \\ \%PYTACZ(*W) \%ZDANIE(*M,*K,*D,*G,*P,*T).$$

Pytaczem, czyli konstrukcją wprowadzającą wątek wypowiedzi (to, o co pytano, jest w przykładowej wypowiedzi wyrażenie „o której”. Jego rozpoznanie powoduje nadanie stosownej wartości zmiennej *W. Zdanie (prosimy nie mylić tej nazwy obiektu w naszej gramatyce z pojęciem stosowanym w językoznawstwie) wprowadza pozostałe elementy znaczenia, przy czym zmienna *G, odpowiadająca godzinie, ma w naszym przykładzie pozostać wolna — pytanie nie powinno zawierać odpowiedzi! Po rozpoznaniu zdania „przyjeżdża pociąg wieczorny z Bydgoszczy” otrzymają wartości tylko zmienne *M (nazwa miasta — Bydgoszcz), *K (kierunek podróży — do Warszawy) i *P (pora — wieczór), a ewentualnym ustaleniem *D i *T zajmie się moduł sterujący konwersacją. Term Q o siedmiu argumentach służy do zebrania w całość elementów znaczenia wypowiedzi. Przed rozpoznaniem komponentów pytania wiadomo tylko, że wartością *STRUKT jest term postaci Q(...) z siedmioma zmiennymi wolnymi; ich wartości są sukcesywnie wyliczane w trakcie analizy.

Dalsza analiza przebiega podobnie — drogą stopniowego uszczegóławiania struktury składniowej wypowiedzi na podstawie odpowiednich reguł, z zachowaniem określonego przez nie szyku wyrazów. W naszym przykładzie zdanie rozpada się na trzy grupy: „przyjeżdża”, „pociąg wieczorny” i „z Bydgoszczy”, których analiza, oprócz stwierdzenia poprawności budowy, dostarcza wartości zmiennych — odpowiednio — *K, *P i *M. Należy zaznaczyć, że mimo arbitralności przyjętego przez nas sposobu hierarchizacji składniowej postulujemy poprawność gramatyczną wypowiedzi. Hierarchizacja ta ma zresztą mocną motywację empiryczną.

Własności gramatyczne konstrukcji i znaczenia jej komponentów zależą pośrednio lub bezpośrednio od własności

pojedynczych wyrazów. Za pośrednictwem argumentów mogą one być przekazywane od wyrazów poprzez grupy do zdania, pytania i wreszcie kwerendy.

Wyrazy nie występują w regułach w sposób jawny, ale są reprezentowane przez symbole nieterminalne oznaczające rzeczownik, czasownik, przysłówek itp. o cechach danych przez argumenty. Cechy te to nazwa części mowy (np. RZECZ), forma (np. „GODZINIE”), postać standardowa (np. GODZINA), parametr semantyczny (np. G) i wartości kategorii fleksyjnych (np. miejscownik, liczba pojedyncza, rodzaj żeński).

Reguła opisująca np. rzeczowniki ma postać

$$\%RZECZ(*F,*KAT,*SEM,*IR) == \\ \#*F - S(*F,RZECZ,*KAT,*SEM,*IR).$$

Symbol terminalny #*F oznacza kolejne — jeszcze nie rozpatrywane — słowo tekstu; w naszym przykładzie po rozpoznaniu introdukcji, a przed rozpoznaniem pytania kolejnym słowem jest „o”. Warunek jest, jak powiedzieliśmy, wywołaniem procedury. Procedura S działa jak przeszukiwanie słownika. Jej wywołanie interpretujemy tak: „znajdź w słowniku lub sprawdź, że się w nim znajduje hasło wyznaczone przez argumenty S”.

Słownik jest bazą danych opisaną za pomocą klauzul unarnych, na przykład

$$+S("GODZINIE",RZECZ,MIEJ.POJ.ZEN,G,GODZINA).$$

Dwa pierwsze parametry w wywołaniu S są w naszym programie zawsze ustalone; wartości pozostałych podlegają ustaleniu lub sprawdzeniu wskutek sięgnięcia do słownika. Do przeglądania słownika zaprzęga się PROLOGOWY mechanizm wyboru klauzuli na podstawie postaci parametrów nagłówka.

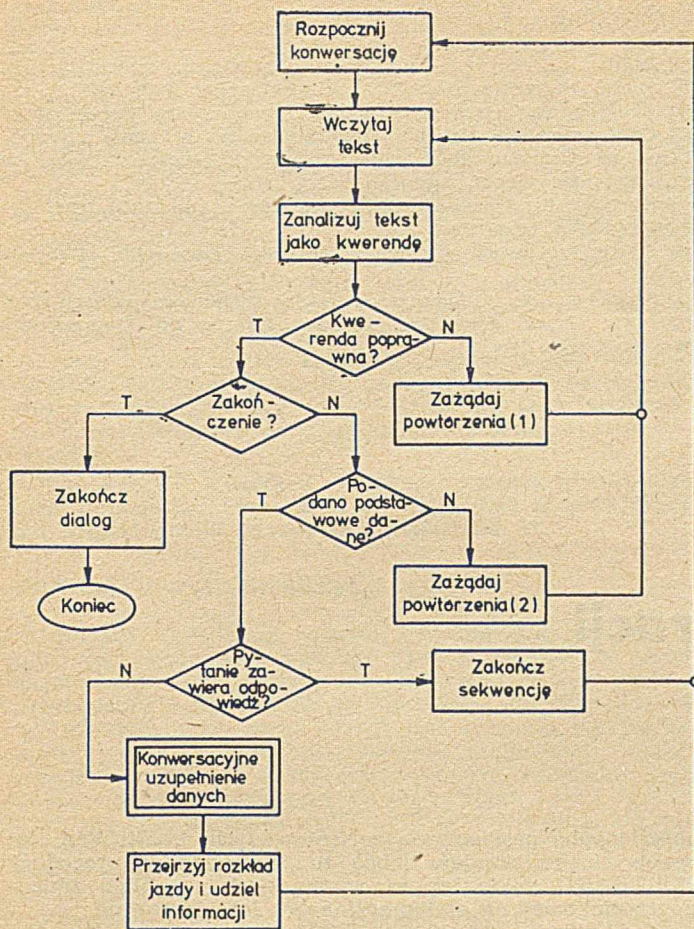
Reguły gramatyki metamorficznej zastosowanej w naszym programie można podzielić na trzy grupy. Pierwsza to ok. 90 reguł definiujących kwerendę. Druga zawiera ok. 30 reguł opisujących składnię wypowiedzi, które są odpowiedziami na pytania informatora o dodatkowe dane. Składnię tych pytań opisuje trzecia grupa reguł, stosowana do syntetyzowania tekstów.

STEROWANIE KONWERSACJA

Konstrukcja dialogu z informatorem opiera się na następujących założeniach:

1. Dialog składa się z pewnej liczby sekwencji <prośba o informację> — <konwersacyjne uzupełnienie brakujących danych> — <odpowiedź informatora> oraz z wypowiedzi kończącej rozmowę (po kwestii informatora „Czym jeszcze mogę służyć”).
2. Poszczególne sekwencje łączy tylko osoba rozmówcy (lub osoby rozmówców), a ściślej — jego płć, natomiast pozostałe dane są lokalne dla jednej sekwencji.
3. Informator udziela odpowiedzi po konwersacyjnym uzupełnieniu brakujących danych, aby np. w wypadku pytania o dworzec tylko dworzec pozostał nie ustalony (rozmówca może powiedzieć „obojętne”, ale musi coś odpowiedzieć na każde z pytań informatora).
4. Akceptuje się tylko wypowiedzi poprawne pod względem gramatycznym, tzn. dopuszczalne w naszej gramatyce, a jakkolwiek błąd powoduje odrzucenie wypowiedzi i prośbę o jej ponowne sformułowanie.
5. Wypowiedź z błędem gramatycznym, wypowiedź nie zawierająca nazwy miasta — punktu krańcowego podróży — i pytanie zawierające odpowiedź (np. „O której jest ekspres do Gdyni o 16.00?”) powodują zakończenie sekwencji bez udzielania odpowiedzi i przejście do następnej sekwencji.

Koncepcję obiegu sterowania w module zarządzającym konwersacją uwidaczniamy na rysunku. Jest on tyleż pogłówny, co nieścisły, w PROLOGU bowiem sterowanie opiera się na wywołaniach procedur i nawracaniu, a nie — jakby to wynikało ze schematu — na skokach i instrukcjach warunkowych. Poniżej przedstawiamy „najwyższe piętra” modułu zarządzającego, pomijając w przytaczanych klauzulach mniej istotne szczegóły, takie jak np. stałe teksty drukowane przez informatora.



Dialog inicjujemy wywołaniem procedury INFORMACJA:

+ INFORMACJA — INFO(*PLEC).

Wywołanie procedury INFO odpowiada jednej sekwencji; ostatnią czynnością procedury powinno być rekurencyjne wywołanie INFO, czyli przejść do kolejnej sekwencji. Pięć rozmówcy jest początkowo nie ustalona.

+ INFO(*PLEC) — WCZYTAJ(*PYTANIE)
— PRZETWORZ(*PYTANIE,*PLEC)

Sekwencja polega na wczytaniu i przetworzeniu wypowiedzi rozmówcy. Zauważmy, że obowiązek rekurencyjnego wywołania INFO spada na procedurę PRZETWORZ.

+ PRZETWORZ(*PYTANIE,*PLEC)
— SYNTAXE(KWERENDA(*STRUKT,*PLEC),NIL,
*PYTANIE)
—/ — REAKCJA(*STRUKT,*PLEC).
+ PRZETWORZ(*PYTANIE,*PLEC) — INFO(*PLEC).

Procedura standardowa SYNTAXE inicjuje analizę wypowiedzi. Jeżeli kończy się ona pomyślnie, co wiąże się z nadaniem wartości zmieniom *STRUKT i *PLEC, to reaguje się na rozpoznaną wypowiedź, odcinając przedtem (za pomocą/) drugi wariant PRZETWORZ. Jeżeli analiza się nie powiedzie, wraca się do początku sekwencji (nawrót przy wykonaniu SYNTAXE oznacza wykonanie drugiego wariantu procedury PRZETWORZ).

+ REAKCJA(ZAKONCZENIE,*PLEC) —/
+ REAKCJA(Q(*W,*M,*K,*D,*G,*P,*T),*PLEC)
— SPRAWDZ(*M,*K,*G,*P) —/
— OBROBKA(Q(*W,*M,*K,*D,*G,*P,*T),*PLEC).
+ REAKCJA(*STRUKT,*PLEC) — INFO(*PLEC).

Wypowiedź kończąca dialog nie wymaga reakcji. Inne wypowiedzi mają strukturę znaczeniową postaci Q(...). Gdy miasto *M i kierunek *K są dane, a godzina *G i pora dnia *P nie są sprzeczne, np. „pociąg nocny o 14.00” (bada to procedura SPRAWDZ), wówczas strukturę poddaje się dalszej „obróbce”; w przeciwnym wypadku wraca się do początku sekwencji.

Procedura OBROBKA powinna realizować dialog uzupełniający luki w kwerendzie, przeglądać rozkład jazdy, formułować odpowiedź i ponownie wywoływać INFO. Wartość *W — wątek wypowiedzi — decyduje o wyborze jednej z analogicznych procedur: OBRD, OBRG, OBRP, OBRT; zajmują się one uzupełnianiem luk w wypadku pytania — odpowiednio — o dworzec, godzinę, porę i typ. Każda z nich wywołuje potem procedurę ODPOWIEDZ, której wykonanie sprowadza się do wykonania procedur ZNAJDZ i INFO.

+ ZNAJDZ(*M,*K,*D,*G,*P,*T)
— R(*T,*K,*M,*G,*D) — DOPASUJ(*G,*P)
— PISZODP(*T,*K,*M,*G,*D) — IMPAS.

Procedura ZNAJDZ wyszukuje w rozkładzie jazdy i wypisuje pozycje zgodne ze swymi parametrami. Rozkład jazdy jest bazą danych opisaną za pomocą klauzul unarnych, np.

+ R(POSPIESZNY,Z,BYDGOSZCZY,21.49,CENTRALNY).
+ R(EKSPRES,DO,GDYNI,5.58,WSCHODNI).

Po znalezieniu pierwszej pasującej pozycji bada się zgodność godziny i pory (DOPASUJ — niezgodność powoduje nawrót), wypisuje się informację w czytelnej postaci (PISZODP), po czym wymusza nawrót (wywołanie nie istniejącej procedury IMPAS). Tak więc zawsze nawraca się do wywołania R, co powoduje dalsze przeszukiwanie rozkładu jazdy.

Wywołanie procedury ZNAJDZ kończy się nawrotem po wyczerpaniu klauzul procedury R. Aby więc wywołać INFO, procedura ODPOWIEDZ musi mieć — pomijając szereg szczegółów technicznych — postać

+ ODPOWIEDZ(Q(*W,*M,*K,*D,*G,*P,*T),*PLEC)
— ZNAJDZ(*M,*K,*D,*G,*P,*T).
+ ODPOWIEDZ(*STRUKT,*PLEC) — INFO(*PLEC).

Zauważmy raz jeszcze, że dane o pięciu rozmócy, ustalone podczas analizy, przekazuje się na początek kolejnej sekwencji. W czasie dalszych analiz będzie się te dane sprawdzać, odrzucając wypowiedzi „innej osoby”.

Przytoczone klauzule są w rzeczywistości nieco bardziej rozbudowane z uwagi na konieczność różnicowania wydruków informatora: służy do tego m.in. drugi — pominięty tutaj — argument INFO. Procedura ODPOWIEDZ jest napisana tak, aby podawać właściwą informację również wtedy, gdy żaden pociąg nie odpowiada wymaganiom podróźnika („Nie ma takiego pociągu.”).

PERSPEKTYWY

Spośród możliwych rozszerzeń i modyfikacji prezentowanego programu najbardziej oczywiste — a przy tym nie trudne do wprowadzenia — jest uprawdopodobnienie jego treści „kolejowych”. Chodzi o likwidację uprzywilejowania Warszawy oraz o przyjęcie bardziej realistycznego kształtu rozkładu jazdy, który mógłby zawierać marszruty pociągów zamiast (jak dotychczas) samych punktów krańcowych. Jeszcze łatwiejsze jest zróżnicowanie odpowiedzi informatora w zależności od wątku kwerendy i od pięci rozmócy.

Pożądanymi byłyby też zmiany prowadzące do zwiększenia elastyczności dialogu. Warto by np. próbować odpowiedzi bez zmuszenia do uzupełniania luk w kwerendzie; byłoby ono konieczne tylko wtedy, gdy liczba pociągów pasujących do zbyt ogólnikowego pytania jest nadmierna. Gdyby w trakcie dopytywania rozmówca kilkakrotnie udzielał niegramatycznej odpowiedzi, można by przyjąć, że odpowiedział „obojętnie”. Także przerwanie dialogu (lub przynajmniej sekwencji) byłoby uzasadnione w razie trudności z podtrzymaniem konwersacji.

Najistotniejsze badań rozszerzenia są najtrudniej osiągalne. Mamy na myśli nawiązywanie w rozmowie do poprzednich sekwencji i zwłaszcza rezygnację z wymogu ab-

solutnej poprawności gramatycznej wypowiedzi. To ostatnie można by zasymulować zwiększając repertuar dopuszczalnych wypowiedzi przez odpowiednią rozbudowę gramatyki.

Innym możliwym kierunkiem dalszych prac jest korzystanie z ogólnej koncepcji programu informatora w celu tworzenia analogicznych programów poświęconych dowolnym, byle dostatecznie wąskim dziedzinom życia. Byłyby to prace w dużym stopniu odtwórcze z informatycznego punktu widzenia, mogłyby natomiast mieć pewne znaczenie lingwistyczne. W tej mierze obiecujące wydaje się podjęcie eksperymentów z różnorodnymi reprezentacjami semantycznymi, dla których najlepszym i najpewniejszym sprawdzianem jest zastosowanie w uruchomionym i przetestowanym programie.

LITERATURA

- [1] Blikle A.: Wybrane zagadnienia liguistyki matematycznej. W tomie „Problemy przetwarzania informacji”, WNT 1974
- [2] Chomsky N.: Syntactic structures. Mouton, The Hague 1957
- [3] Colmerauer A.: Metamorphosis grammars. W tomie „Natural Language Communication with Computers”, Lecture Notes in Computer Science 63, Springer-Verlag 1978
- [4] Kłuzniak F., Szpakowicz S.: Wprowadzenie do PROLOGU. INFORMATYKA nr 4/1979
- [5] Saloni Z.: Cechy składniowe polskiego czasownika. Wrocław 1976
- [6] Szpakowicz S.: Automatyczna analiza składniowa polskich zdań pisanych. Praca doktorska, Instytut Informatyki Uniwersytetu Warszawskiego 1978

Układy bardzo dużej integracji w technice komputerowej

Układy scalone bardzo dużej integracji (very large scale integration — VLSI) już obecnie wywierają znaczny wpływ zarówno na architekturę urządzeń komputerowych, jak i na sposób ich wykorzystania. Według ocen ekspertów jest to początek zasadniczych zmian, jakie nastąpią w technice komputerowej przyszłości.

W wyniku wprowadzania VLSI w technice komputerowej można zaobserwować trzy podstawowe kierunki zmian:

- 1) zmniejszenie kosztów realizacji bloków funkcjonalnych oraz miniaturyzacja konstrukcji
- 2) zwiększenie liczby funkcji realizowanych w urządzeniu przy zachowaniu tego samego poziomu kosztów
- 3) coraz szersze stosowanie sprzętowych realizacji nowych funkcji oraz funkcji dotychczas realizowanych programowo.

ZMIANY W STRUKTURZE JEDNOSTKI CENTRALNEJ

Zastosowanie układów VLSI spowoduje, że struktury projektowanych jednostek centralnych będą zbliżały się coraz bardziej do struktur modularnych, których moduły będą stanowić sprzętowe realizacje określonych funkcji. Rozwiązania te z kolei będą w znacznym stopniu oddziaływać zarówno na metody projektowania, jak i użytkowania systemów komputerowych.

Jednostka centralna stanowi w strukturze modularnej blok funkcjonalny podporządkowany procesorowi sterującemu, sprzężonemu z pamięcią mikroprogramów. Zawartość pamięci mikroprogramów całkowicie określa maszynę pod względem listy rozkazów. Centralna jednostka może zatem pracować z różnymi listami rozkazów, zależnie od zawartości pamięci mikroprogramów, a więc może również emulować maszyny wcześniejszych generacji, co pozwala korzystać z opracowanego dla nich oprogramowania.

Obecnie w większości przypadków rozkazy wejściowe stanowią elementarne rozkazy maszynowe generowane przez assembler lub kompilator. Wszystkie wprowadzane do maszyny programy, niezależnie od ich charakteru i wykorzystywanego języka, są przekształcane w skończony zbiór rozkazów, które mogą być realizowane za pomocą odpowiednich mikroprogramów. Taka realizacja programów jest w wielu przypadkach nieefektywna, ponieważ zalety problemowo zorientowanych języków programowania są niweczone w procesie translacji na język maszynowy.

W ostatnich latach można zaobserwować tendencje zmierzające do uzupełniania mikroprogramów specyficznymi sekwencjami mikrorozkazów przyspieszającymi wykonanie

określonych, problemowo zorientowanych konstrukcji językowych. Przykładem mogą tu być niektóre procedury FORTRANU, takie jak konwersja liczb z postaci zmienno-przecinkowej na stałoprzecinkową lub odwrotnie.

Zwiększenie pamięci głównej w wielu przypadkach pozwoli zrezygnować ze zwiększania pamięci metodami programowymi, takimi jak pamięć wirtualna lub stronicowanie, wymagającymi odpowiednich programów zarządzających. Nie należy jednak sądzić, że problemy te nie będą występowały nadal, gdyż wymagania dotyczące zwiększania pojemności pamięci rosną co najmniej tak szybko jak ich dostępność.

MIKROPROCESORY W STRUKTURACH KOMPUTEROWYCH

Łatwo można wykazać, że zadana przepustowość przetwarzania może być uzyskana taniej przez połączenie kilku tanich mikroprocesorów niż przez budowanie dużego procesora monolitycznego. Za fałszywą należy jednak uznać argumentację, że za pomocą wielu współpracujących mikroprocesorów można uzyskać moc przetwarzania dużego systemu komputerowego, przykładowo takiego jak IBM 370/158. Poza niektórymi specyficznymi zastosowaniami (jak np. rozpoznawanie obrazów) nie ma praktycznie metody, aby program sekwencyjny (a większość programów jest z natury sekwencyjna) był wykonywany szybciej w sposób równoległy za pomocą małych procesorów.

Na uwagę zasługuje inna klasa struktur komputerowych, w których mogą być wykorzystane zalety mikroprocesorów, nawet w przypadku dużych komputerów. Do tej klasy należą struktury realizujące przetwarzanie potokowe (ang. *pipelining*). Koncepcja przetwarzania potokowego jest stosowana już od dość dawna głównie w dużych jednostkach arytmetycznych, i ogólnie mówiąc polega na rozbiściu złożonej operacji na szereg działań elementarnych, które mogą być wykonywane w odrębnych procesorach.

W skonstruowanej do tego celu strukturze przetwarzania potokowego sygnały są przekazywane kolejno z wyjść mikroprocesorów znajdujących się we wcześniejszych warstwach na wejścia mikroprocesorów umieszczonych w następnych warstwach struktury. Każdy z procesorów takiej struktury jest procesorem specjalizowanym sterowanym przez centralną jednostkę sterującą.

Koncepcja przetwarzania potokowego jest jedną z form ogólnej zasady rozproszenia funkcji centralnej jednostki przetwarzającej, której inną formą mogą być przykładowo procesory peryferyjne w komputerach serii CYBER.

Dystrybucja funkcji systemu jest bardzo efektywna w przypadku operacji wejścia/wyjścia. Funkcje są tu dobrze zdefiniowane, a interfejsy z jednostką centralną mogą być określone w sposób jednoznaczny. Poza tym w niektórych przypadkach funkcje wejścia/wyjścia mogą być zależne od zastosowań, a więc powinny być programowane przez użytkownika. Dotyczy to zarówno tradycyjnie stosowanych kanałów multipleksorowych i kanałów szybkich urządzeń wejścia/wyjścia, jak i jednostek sterujących stosowanych w telekomunikacji. Do typowych funkcji inteligencji rozproszonej, umieszczanych w mikroprocesorowych jednostkach sterujących urządzeniami wejścia/wyjścia, należy zaliczyć sprawdzanie i korektę błędów, formowanie kolejek i tablic danych, buforowanie, obsługę przerwań, procedury diagnostyczne i formatowanie danych.

Jedną z zalet takiej koncepcji jest mniejsza zależność działań centralnej jednostki przetwarzającej od operacji wejścia/wyjścia, a więc większe uniezależnienie przepustowości obliczeniowej od przesłań informacji na wejściu i wyjściu.

Inna korzyść wynika z określonej lokalizacji odpowiednich funkcji, wskutek czego zmniejsza się pracochłonność ich projektowania, realizacji i testowania.

OBNIŻKA KOSZTÓW EKSPLOATACJI

Istotnym czynnikiem wpływającym na kształtowanie się przemysłu komputerowego lat 80-tych są rosnące koszty eksploatacji, które obecnie dla dużych systemów rosną w przybliżeniu o 5% rocznie w odniesieniu do ceny dzierżawy systemu, przy czym wzrost jest znacznie szybszy dla małych systemów. Wzrost kosztów eksploatacji wynika przede wszystkim z rosnących zarobków personelu obsługującego oraz konieczności magazynowania części zamiennych nie przynoszących dochodu. Zrozumiałe stają się więc poszukiwania takich rozwiązań, w których malejącymi kosztami sprzętu można byłoby skompensować rosnące wynagrodzenie kwalifikowanego personelu.

Pierwszym istotnym składnikiem kosztu eksploatacji, który może być obniżony za pomocą dodatkowego sprzętu, jest koszt diagnozy uszkodzenia. Składnik ten już obecnie jest uwzględniany przez prawie wszystkich producentów dużych systemów i coraz bardziej jest doceniany przez producentów minikomputerów. Najczęściej stosowane przedsięwzięcia to specjalne rozwiązania układów wejścia/wyjścia oraz opracowywanie funkcji diagnostycznych, które mogą być wykorzystywane do testowania różnych układów komputera. Testowanie może być wykonywane na poziomie pakietów i już obecnie jest stosowane w niektórych urządzeniach peryferyjnych i minikomputerach, przy czym sygnały wejściowe uzyskiwane są za pomocą przełączników, a do sygnalizacji stanów wyjść wykorzystywane są diody świecące. Bardziej wyrafinowane metody stosowane są w dużych systemach, dla których minikomputer stanowi narzędzie testowania i lokalizacji uszkodzonego modułu lub elementu. W większości obecnie projektowanych dużych systemów komputerowych procesor diagnostyczny jest wbudowywany w system w celu wykorzystania go przez obsługę do szybkiej detekcji uszkodzenia.

Rozwiązania tego typu stosowane są stosunkowo niedawno, gdyż metody automatycznej diagnostyki charakteryzują się dużą złożonością, a to utrudnia ich realizację. Dostępność tanich technologii wytwarzania skomplikowanych układów VLSI już w niedalekiej przyszłości pozwoli realizować układowo skomplikowane funkcje diagnostyczne badające poprawność układów systemu i generujące sygnały identyfikacji stanu systemu.

Procesor diagnostyczny, wyposażony w modem, umożliwi prowadzenie zdalnego testowania i lokalizacji uszkodzeń. Operator korzystający z takiej aparatury może zdalnie wprowadzać do badanego systemu specjalne programy testujące i otrzymywać wyniki testów, które w wielu przypadkach pozwalają stwierdzić, że występujące błędy są błędami natury programowej, a nie sprzętowej.

W niedalekiej przyszłości należy oczekiwać, że sprzęt diagnostyczny będzie dokładnie lokalizował uszkodzenia wskazujące użytkownikowi sposób ich usunięcia, na przykład przez wymianę uszkodzonego pakietu, co przesunie ciężar czynności naprawczych na stronę użytkownika, zwalniając go w wielu przypadkach z obowiązku korzystania z obsługi serwisowej. Taki sposób obsługi na razie będzie stosowany w odniesieniu do małych komputerów, w końcu lat 80-tych również w odniesieniu do dużych systemów komputerowych.

Rozwój sprzętowych urządzeń diagnostycznych we wszystkich omówionych przypadkach będzie czynnikiem wpływającym na zmniejszenie kosztów obsługi zarówno w zakresie diagnostyki, jak i napraw.

Końcowy krok w kierunku zabezpieczenia systemu przed skutkami uszkodzeń będzie polegać na stosowaniu sprzętu redundancyjnego. Decyzja przyjęcia takiego rozwiązania musi być poprzedzona porównawczą analizą kosztu dodatkowego sprzętu z jednej strony i skutkami uszkodzenia systemu z drugiej strony. Malejący koszt dodatkowego sprzętu (realizowanego na układach VLSI) coraz częściej będzie uzasadniał celowość jego stosowania w zestawieniu ze skutkami wyłączenia systemu na czas remontu w przypadku jego uszkodzenia.

ZARZĄDZANIE BAZAMI DANYCH

Wiele wskazuje na to, że najbardziej dynamicznie rozwijającą się dziedziną zastosowań komputerów w latach 80-tych będzie zarządzanie bazami danych, co wywoła również potrzebę opracowania bardziej wyszukanych metod manipulowania danymi, jak i podniesienia technologicznych parametrów sprzętu komputerowego. Będą działały zatem czynniki stymulujące zarówno rozwój sprzętu, jak i oprogramowania.

Dla operacji zarządzania danymi wymagana jest duża moc obliczeniowa, przy czym operacje te zwykle są wykonywane w komputerze równoległe z innymi programami użytkowymi. Ze względu na to, że większość operacji zarządzania danymi wykonywana jest repetycyjnie i może być standaryzowana, pożądane jest, aby operacje te były wykonywane w wysoko specjalizowanych procesorach przystosowanych do efektywnego wykonywania tych zadań. Należy zatem stwierdzić, że celowe staje się zdefiniowanie i opracowanie specjalizowanego języka do zarządzania danymi. Otrzymujemy więc kolejny specjalizowany procesor językowy, który może być dołączony do komputera lat 80-tych o strukturze modularnej. Operacje zarządzania danymi wymagają także dużych pamięci o bezpośrednim dostępie. I w tym zakresie istnieją już prototypowe opracowania pozwalające uzyskiwać odpowiednie pamięci w technologii magnetycznych domen cylindrycznych, z których dane mogą być przekazywane bezpośrednio do pamięci głównej.

NOWE KONCEPCJE SYSTEMOWE

Przetwarzanie rozproszone bez wątplenia stanowi bezpośrednią konsekwencję pojawienia się układów dużej i bardzo dużej integracji. Jedynie zdecydowany spadek kosztu realizacji funkcji logicznych oraz pamięciowych pozwoli na zwiększenie i rozproszenie funkcji systemowych w poszczególnych blokach złożonych systemów.

W przeszłości projektanci systemów komputerowych koncentrowali się prawie wyłącznie na doskonaleniu i optymalizacji jednostek centralnych. Obecnie coraz więcej uwagi poświęca się efektywności systemu, co w rezultacie prowadzi do rozproszenia funkcji w blokach składowych systemu z uwzględnieniem czynnika lokalnego — chodzi o łatwość korzystania i dostępność informacji — oraz czynnika globalnego, tzn. wydajności i jakości.

Ta nowa orientacja będzie coraz silniej oddziaływać na projekty systemów. Poważne trudności wywołuje jednak brak możliwości zdefiniowania kryterium efektywności systemów rozproszonych. Po usunięciu z jednostki centralnej funkcji obliczeniowych i pamięciowych straciło znaczenie kryterium sprowadzające się do wyznaczania stosunku cena/wydajność. Ponadto trudne stało się porównywanie rozwiązań alternatywnych na etapach projektowania koncepcyjnego systemów rozproszonych. Trudne zadanie wyboru najbardziej odpowiedniego sprzętu sprowadza się obecnie do przeprowadzenia dokładnej analizy możliwych rozwiązań i zestawienia materialnych i niematerialnych zalet każdego z nich.

Projektant systemu komputerowego powinien również pamiętać, że nie istnieje system rozproszony ogólnego zastosowania. Każdy użytkownik ma swoje wymagania, a różnice występują nawet w tej samej gałęzi przemysłu. Producentom nie pozostaje zatem nic innego, jak oferowanie racjonalnych rozwiązań systemowych bazujących na doświadczeniu uzyskanym u wiodących użytkowników i zaspokajających aktualne potrzeby rynkowe.

Sytuacja ta stwarza zapotrzebowanie na analityków systemów nowego typu, to jest takich, którzy mniejszą wagę będą przywiązywali do wydajności procesora lub parametrów pamięci, a bardziej zaangażują się w problematykę interfejsów, powiązań sprzętu z oprogramowaniem i właściwą koordynację wielu fizycznie lub geograficznie rozproszonych funkcji.

Układy VLSI będą wywierały również znaczny wpływ na oprogramowanie. Projektowanie i praktyka oprogramowania będą zmieniać się co najmniej w trzech aspektach, których łączne oddziaływanie będzie prowadziło do zmniejszenia kosztu oprogramowania w przeliczeniu na jeden rozkaz.

Po pierwsze coraz więcej funkcji systemu komputerowego będzie realizować się sprzętowo z wykorzystaniem pamięci mikroprogramów. Dotyczy to przede wszystkim tych funkcji, które można określić jako standardowe ze względu na ich budowę wewnętrzną i strukturę interfejsu. Do tej grupy funkcji można zaliczyć na przykład takie funkcje użytkowe, jak konwersja kodów, przesyłanie danych, operacje wejścia/wyjścia. Analogiczne funkcje elementarne mogą być określone dla kompilatorów oraz dla systemów operacyjnych. W wyidealizowanym systemie metodami programowymi powinno się rozwiązywać tylko te funkcje, dla których jest to niezbędne. Funkcje o ustalonej lub rzadko zmienianej strukturze powinny być realizowane sprzętowo lub w postaci oprogramowania firmowego (programów umieszczanych przez producenta w pamięciach stałych).

Drugi aspekt oddziaływania układów VLSI na oprogramowanie dotyczy programów zarządzających (systemów operacyjnych), stanowiących pomost pomiędzy sprzętem i programami użytkowymi. Zwykle większość programów zarządzających należy do klasy funkcji modyfikowalnych firmowo (podlegających tzw. rewizjom lub uaktualnieniom). Jednak w rzeczywistości zasadniczą przyczyną uniemożliwiająca zrealizowanie systemu operacyjnego w postaci oprogramowania firmowego jest trudność dokładnego określenia własności systemu operacyjnego dla produktu końcowego, bowiem w skład oprogramowania firmowego mogą być włączone jedynie dobrze sprawdzone i stabilne programy zarządzające.

Oprogramowanie zarządzające musi zapewnić wymagane własności funkcjonalne i umiejętnie godzić szereg wyma-

gań konfliktowych, na przykład szybkość realizacji programu z dostępną wielkością pamięci. I w tym miejscu pomocną okazuje się technologia układów VLSI, dając projektantowi do dyspozycji tanie pamięci, z których może on budować duże redundancyjne systemy pamięciowe, zapewniające szybszą realizację programów.

Stosowanie pamięci o dużej pojemności będzie jednak miało najbardziej istotny wpływ na pracę programistów. Jak dotąd celem programisty była przede wszystkim optymalizacja wykorzystania pamięci. Dotyczyło to zarówno oprogramowania systemowego, jak i programów użytkowych. Powszechne obecnie dążenie do najbardziej optymalnego kodowania stanie się mniej uzasadnione, gdy większość systemów będzie miała postać zmieniających się konfiguracji sprzętowych nie pozwalających bazować na stabilnych definicjach i procedurach. Coraz szersze stosowanie układów VLSI spowoduje, że efektywność kodowania będzie rozpatrywana przede wszystkim w kategoriach czasu wykorzystywanego na pisanie, testowanie i wykonanie dokumentacji programów, natomiast znacznie mniejszą wagę przykładać się będzie do wielkości pamięci, którą te programy zajmują.

Mimo stałego i szybkiego rozwoju technologii zachowany jest stosunkowo duży dystans między możliwościami technologii i praktycznymi rozwiązaniami układowymi stosowanymi w seryjnym sprzęcie komputerowym. Jest to spowodowane przede wszystkim ostrożnością producentów we wprowadzaniu najnowszych technologii do praktycznych realizacji funkcji systemowych. W przypadku, gdy rozwiązania są zbyt nowatorskie, wiąże się z nimi ryzyko niezgodności z wymaganiami rynku lub możliwość wyparcia przez tańszą technologię. Z kolei rozwiązania zbyt konserwatywne nie są konkurencyjne pod względem ceny.

Wprowadzanie układów VLSI zmusi użytkowników systemów komputerowych do innego spojrzenia na analizę systemu, mianowicie będzie od nich wymagało dokładniejszego sprecyzowania całościowej koncepcji systemu, pozwalając równocześnie na znacznie mniejsze zaangażowanie w problematykę optymalnego wyboru poszczególnych urządzeń systemu, z jednostką centralną włącznie.

Opracował Tadeusz SINKIEWICZ na podstawie artykułu J. M. Gabeta „VLSI: The Impact Grows”, DATAMATION, czerwiec 1979

Czytajcie i prenumerujcie INFORMATYKĘ!

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 — 90 zł
- półroczna — 180 zł
- roczna — 360 zł

Przedpłaty przyjmowane są w następujących terminach:

- do 10 marca — na II kwartał
- 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

Prenumeratę ze zleceniem wysyłki za granicę przyjmuje Centrala Kolportażu Prasy i Wydawnictw, ul. Towarowa 28, 00-958 Warszawa, konto NBP XV Oddział w Warszawie nr 1153-201045-139-11 w terminach obowiązujących dla prenumeraty krajowej.

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

Spis treści rocznika 1979

ARTYKUŁY PROBLEMOWE

	nr	str.			nr	str.
BABA WIESŁAW — Usprawnienia projektowania algorytmów	8-9	16	KAMIŃSKI KRZYSZTOF — Brytyjskie Centrum badań symulacyjnych (oprac.)	7	19	
BEREZIŃSKI MIROSŁAW, HOŁUBIEC JERZY — Program rozwoju informatyki we Włoszech	2	11	KIERZKOWSKI ZBIGNIEW, KOPERSKI JACEK — Możliwości automatycznej korekcji błędów składniowych w programach	3	3	
BIERNAT JANUSZ — Systemy komputerowe ze strukturami nadmiarowymi	6	13	KLEPACZ WŁADYSŁAW — Sprzęt informatyczny na Wiosennych Targach Lipskich 79	8-9	32	
BIEGASA KRZYSZTOF — TEED — program dla kontekstowej edycji zbiorów	6	11	KLUŻNIAK FELIKS, SZPAKOWICZ STANISŁAW — Wprowadzenie do PROLOGU	4	10	
BILIŃSKI MAREK, PTAK EDWARD — Język wyszukiwawczy JWO	11	18	KOLENDOWSKI JERZY, WIT ROMUALD — Komputerowe rozwiązywanie różniczkowych cząstkowych	1	4	
BOCIAN LIDIA, JUCHNIKOWSKI GRZEGORZ — Metoda sieciowa GERT	11	9	KOWALSKA JADWIGA, NOWAKOWSKI ANTONI — System wspomagane nauczania SYPRON	5	17	
CHMIEL JÓZEF — Wykorzystanie metod programowania liniowego na komputerach ODRA 1300	10	13	KOWALSKI LECH, KUCHCIŃSKI KRZYSZTOF, SZEJKO STANISŁAW, WISZNIEWSKI BOGDAN — Język BASIC dla dwuprocesorowej konfiguracji MERA 300	2	4	
CHROBROWSKI MICHAŁ, JANECKI LESZEK, DOROZIK LEON — System koordynacji przewozów TRANSTER PKS	5	14	KOWALSKI KAZIMIERZ, ZGRZYWA ALEKSANDER — Kumulacja danych bibliograficznych w systemie SDI	8-9	30	
CICHY MAREK — Elementy gry w nauczaniu programowania	4	4	KRÓL JERZY, KURAS JACEK, LEMBAS JERZY — Język symulacyjny dla komputerów ODRA 1300	11	16	
CIEŚLIŃSKI MAREK, GERWIN KRZYSZTOF, PLEWAKO RYSZARD — Zabezpieczenie informacji w systemie rezerwacji miejsc na promach	8-9	25	KRÓLIKOWSKA BARBARA, KUČZERA PIOTR, OLEJNICZAK WOJCIECH — Metodologia projektowania według podstawowych struktur systemu informatycznego	5	7	
CZYREK ZBIGNIEW, GERWIN KRZYSZTOF — System obsługi bazy komputerowej z ruchomymi terminalami i radiową transmisją danych	1	10	KULIKOWSKI JULIUSZ — REDT — Ogólnodostępna sieć transmisji danych	6	19	
DABROWSKI HENRYK — Bank Danych Statystycznych ROZWOJ i kierunki budowy języka użytkownika	7	9	KURAS MARIAN — Zastosowania informatyki w organizacjach gospodarczych w świetle badań ankietowych	12	3	
DRAŻEK ZYGMUNT, SZYJEWSKI ZDZISŁAW — Wybrane problemy projektowania oprogramowania	5	12	LEWOC JÓZEF, MLEK MAREK, NAWOJSKI JANUSZ, ROZENT MIECZYŚLAW — Automatyczna odnowa pracy systemu rzeczywistego dla zestawu ODRA 1325/SMA	6	4	
DREWNIAK WIT — Niezawodność systemów komputerowych ODRA 1305 i R-32	7	5	LIZUREK MIROSŁAW — Oprogramowanie kaset urządzenia INTEL DIGIT-PI	11	13	
DUBIELEWICZ ARTUR, JAN MAGOTT — Sprzętowa realizacja systemów operacyjnych	11	1	ŁADOS ZBIGNIEW — Wykrywanie błędnych numerów	1	13	
DZIEDZICZAK IGNACY, KOLBUSZ EDWARD, WIERZBICKI TADEUSZ — Kierunki rozwoju metodologii projektowania informatycznych systemów zarządzania	5	4	Jak zmniejszyć zużycie papieru tabulogramowego	4	13	
FRANCZAK RYSZARD, ZIELIŃSKI KAZIMIERZ — STABEC-Z usprawnia zarządzanie systemem elektroenergetycznym	10	6	Skomputeryzowane dokumenty dla bankowych rozliczeń pieniężnych	8-9	4	
GACKOWSKI ZBIGNIEW — O systemach informatycznych inaczej	1	1	ŁĄCKI STANISŁAW — Uwarunkowania prawidłowej eksploatacji pamięci taśmowych	4	4	
GAGSCH SIEGFRIED — Zasady wspomaganego komputerowo syntezy zadań	7	1	MITAL ZENON — Niezawodność oprogramowania	11	7	
GILECKI RYSZARD, SZUBERT MAŁGORZATA — Praktyczne aspekty zastosowania systemu GEORGE 2	8-9	22	MROZIK STANISŁAW — Organizacja prac nad produktami programowymi ogólnego przeznaczenia	8-9	10	
GILECKI RYSZARD, SAWICKI WOJCIECH, TABACZYŃSKI WŁODZIMIERZ — Program formatujący teksty	11	4	MURASZKIEWICZ MIECZYŚLAW — Wyszukiwanie i sortowanie danych za pomocą układu komórkowego	3	9	
GAGSCH SIEGFRIED — Zasady wspomaganego komputerowo Jednolitego Systemu	6	1	Wyświetlająca jednostka sterująca	10	16	
GŁODEK ZENON, PERENC JÓZEF, SZEWCZYK AGNIESZKA — Symulacja w wyborze struktury zbiorów danych	5	10	OLEŃSKI JÓZEF — Granice i instrumenty centralnych systemów informatycznych	2	1	
GOGOLEK WŁODZIMIERZ — Konwerter analogowo-cyfrowy jako terminal komputera ODRA 1325	1	7	Część 1.	2	1	
GOLIŃSKI JAN — Aktualne problemy oprogramowania	8-9	1	Część 2.	3	1	
GOŚCIŃSKI ANDRZEJ, NAWARECKI EDWARD — Komputerowe sterowanie procesami dyskretnymi	8-9	6	OSTASIEWICZ WALENTY — Szkic z historii programowania	8-9	27	
GÓRNICKI TADEUSZ, WYRZYKOWSKI MICHAŁ — Uruchamianie systemów mikroprocesorowych przy użyciu MERY 303	7	11	PAWLIKOWSKI KRZYSZTOF — Pętlowe sieci łączności w splywowo-rozwojowych systemach informatycznych	2	14	
HAC ANNA — Porównanie języków SIMULA 67, PASCAL i FORTRAN na przykładzie symulacji systemu operacyjnego	2	9	ROBAKIEWICZ MACIEJ — Informatyka w projektowaniu budowlanym (na tle konferencji INFOPRO 79 i PaC 79)	12	8	
HOJA JERZY, SZCZYPTA ANTONI, TLAGA WALDEMAR, ZIELONKO ROMUALD — Zastosowanie MERY 305 do diagnostyki oraz kontroli pakietów analogowych i cyfrowych	1	16	SINKIEWICZ TADEUSZ — Jak wybrać mikroprocesor	10	2	
JASKÓLSKI STANISŁAW — Doskonalenie procesu przygotowania danych	10	9	Układy bardzo dużej integracji w technice komputerowej	12	16	
JĘDRASZKO JACEK — Zastosowanie urządzenia końcowego do aktualizacji zbioru danych	2	6	STRZAŁKOWSKI PIOTR — Między UNIDATA a HONEYWELLEM	3	14	
JÓZWIAK EWA — Mierzenie jakości i wydajności programowania	3	17	Część 1.	4	16	
	4	19	Część 2.	4	16	
			Modularyzacja	6	7	
			Część 1. Pojęcie modułu	7	14	
			Część 2. Języki modularne			
			SZCZUCKI JERZY — Komputeryzacja gospodarki materiałowej na przykładzie województwa wrocławskiego	3	12	
			SZPARKOWICZ STANISŁAW, ŚWIDZIŃSKI MAREK — Programowanie w PROLOGU: informator kolejowy	12	12	
			TRYK WITOLD — Informatyka w Hiszpanii (oprac.)	6	16	
			WALUSZEWSKI JAN — Czy oprogramowanie będzie prawie chronione?	6	14	
			WISZNIEWSKI BOGDAN — Język BASIC dla dwuprocesorowej konfiguracji MERA 300	2	4	
			ZAPOLSKI ZDZISŁAW — Czynniki wpływające na metodykę projektowania systemów	3	6	
			ZURAWSKI ANDRZEJ — Programy dwukierunkowej konwersji ODRA-RIAD w systemie OS/JS	12	1	

Projektowanie systemu informatycznego w oparciu o wspólną bazę danych — Wisław Dubczyński, Tadeusz Gryc Część 3.	1	39
Centralna baza danych handlu wewnętrznego w CSRS — Jan Smrčina	2	39
Bank danych statystycznych w technologii DBTG CODASYL — Marek Lasota	3	37
Realizacja funkcji zarządzania bazą danych jako podstawa ich klasyfikacji — Maria Maler-Kapcia	4	35
Koszty i efekty bazy danych — oprac. Maciej Leśny	5	37
Główne kierunki rozwoju zastosowań relacyjnego modelu bazy danych — Bolesław Szymański	6	39
Administrator zastosowań — Andrzej Brandt	7	36
Obiektowa struktura bazy danych — Witold Rekuć	8	38
Maszyny bazy danych — Jerzy Pasula	8—9	55
Wybrane problemy relacyjnej bazy danych — Ewa Józwiak		
Część 1. Projektowanie i własności języków relacyjnych	8—9	58
Część 2. Techniki implementacyjne	10	37
O klasyfikacji systemów zarządzania bazą danych — Leszek A. Maciaszek	10	39
Technologia baz danych w praktyce — Grzegorz Gruchman	11	34
Projekt struktury pamięci — Andrzej Brandt	11	36
Kompresja baz danych przy pomocy kodów zmiennej długości — Andrzej Siemiński	12	36

SZTUCZNA INTELIGENCJA

Realia i perspektywy — Marek Holyński	1	26
Metody rozpoznawania obrazów — Zbigniew M. Wójcik	2	17
Na trzy głosy		
Nie ma jednolitej definicji — Andrzej Dziurkowski	3	19
Mizerne naśladownictwo — Marek Gliński	3	20
Heurystyka pomoże — Agnieszka Szewczyk	3	20
Wykorzystanie analogii przy rozwiązywaniu problemów — Adam Biela	4	21

NASZE RECENZJE

Projektowanie a komputer — Stanisława Bonkowicz-Sittauer	1	37
Narodziny nowego działu statystyki gospodarczej — Adam B. Empacher	3	34
„Informacja i świat w którym żyjemy” — Jerzy Seidler	4	34
Glossariusz informatyczny: od <i>abontenta</i> do <i>żądania</i> — Adam B. Empacher	5	35
Teleinformatyka na półkach księgarskich — Juliusz Kulikowski	6	34
Trudna sztuka programowania — Stanisława Bonkowicz-Sittauer	7	34
Piąte wydanie na temat FORTRANU — Tadeusz Szuskiewicz	7	35
Efektywność systemów informatycznych zarządzania — Zygmunt Bieńko	8—9	48
Wtajemniczenie teorii(hiper)grafowe — Adam B. Empacher	8—9	49
Kompiuter (nieprawidłowo!) — A.B.E.	10	III str. okł.
Analityczna weryfikacja programów — Stanisława Bonkowicz-Sittauer	11	32
Informatyka w transporcie — Zbigniew Bieńko	11	33
Repetitorium języka PLAN — A. B. Empacher	12	40

NAUCZANIE I KSZTAŁCENIE

Nauczanie projektowania w ogrzewnictwie — Ryszard Śnieżyk	1	35
Studenckie koła informatyki — Waldemar Siwiński	2	32
Model nauczania informatyki — Mieczysław Maruszkiewicz, Zbigniew Nowicki	3	32
Informatyka na farmacji — Maciej Paprocki	4	32
Prosty system ewidencji studentów — Ludomir Kieszczyński, Zbigniew Ogonowski	7	30
Możliwości kształcenia informatyków we Francji — Bogna Lichodziejewska	8—9	50

KSPZ — komputerowy system planowania zajęć — J. Polowczyk, H. Runka, Z. Rzemyskowski, W. Sikora	8—9	51
--	-----	----

OŚRODKI INFORMATYKI PREZENTUJĄ

Programowanie modularne w praktyce — Jiří Zaveský	3	25
Zastosowanie pakietów programów SOD w CPN Poznań — Rafał Łykowski	6	31
ROBOTRON 4201 do zarządzania produkcją urządzeń elektrycznych — Lech Łasica, Kazimierz Zieliński	12	34

TRYBUNA CZYTELNIKA

Sprostowanie — Henryk Kopia	1	38
O krytyce sytuacji nie istniejących — Edward Nawrocki	2	38
W odpowiedzi na list Pana E. Nawrockiego — Stanisław Mrozik	2	38
O OPTY nie OPTYmistycznie — Aleksander Lesz	4	32
Nieco więcej o OPTY — Stefan Pleszczyński	4	32
Dwa listy w sprawie oszczędności papieru tabulogramowego	7	33

USPRAWNIENIA, NOWE KONCEPCJE, POMYSŁY

Nowe makrorozkazy GEORGE 2 dla ALGOLU i FORTRANU — Konrad Maruszczyk, Irena Zalewska	1	36
System PASCAL 360 — Michał Iglewski	2	36
System edycji programów w języku SAWIK — Lech Brennek, Bogdan Lebedziuk	8—9	46
Diagnostyka komputerowa drukarki wierszowej DW 325 — Jerzy Zakrzewski	10	36

ZAGADNIENIA PRAWNE

Zagadnienia prawne nie rozwiązane — Andrzej Klimek	2	34
Dwa słowa o wymianie zdań — Jan Waluszewski	2	35
Patentowanie oprogramowania komputerów	3	III str. okł.

ZE ŚWIATA

Rozwój informatyki na Węgrzech — oprac. Witold Tryuk	1	32
Informatyka w Rumunii — oprac. Witold Tryuk	1	34
„Należy tworzyć swoje własne systemy...” — oprac. Z. Moroz	3	30
IFIP 80 — Władysław Klepacz	3	31
Drukarki komputerowe — Zbigniew Naotyński	4	30
Starcie ICL z IBM (T.J.)	4	30
Urządzenia fotoskładu drugiej generacji (T.J.)	4	31
Nowe modele SIEMENSA (T.J.)	4	31
Prasa brytyjska o komputerach ROBOTRON (T.J.)	4	31
„Sądę, że informacja będzie uznana za źródło wartości tego samego rzędu co kapitał i praca” — oprac. Piotr Strzałkowski	5	24
Tanie terminale		
ICL — FEEDBACK DATA LTD		
Pamięci laserowe o pojemności 10 G bitów — oprac. W. Klepacz	5	25
EuroIFIP — Władysław Klepacz	6	25
Największy na świecie — oprac. A. Nawrocki	6	29
Co słyhać w informatyce francuskiej? (P.S.)	7	26
Park komputerowy (P.S.)	7	26
Zatrudnienie (P.S.)	7	26
Centrum banków informacji na Riwierze (I.S.)	7	26
Nowe czeskosłowackie urządzenia (W.K.)	8—9	43
Odpowiedź SIEMENSA na system IBM (T.J.)	8—9	44
IBM System 38 (T.J.)	8—9	44
HONEYWELL w Jugosławi (W.K.)	8—9	44
CYBER dla Chin (I.S.)	8—9	44
Na wydziale Informatyki w Pittsburghu — Jarosław Deminēt	10	30
Komputerowa giełda surowców wtórnych	10	31
Nowy kombinat CSRS (W.K.)	10	31
Automatyzacja pracy dziennikarskiej (W.K.)	10	31
Konferencje IFIP na temat kształcenia — Stanisław Waligórski	11	28
800-megabajtowy dysk (Z.N.)	11	29

	nr	str.		nr	str.
Szybka drukarko-kopiarka (Z.N.)	11	29	Aby budować szybciej, lepiej i taniej — Krystyn Bernatowicz	2	24
Informatyka w Tennessee Valley Authority — Jan W. Owsinski	12	27	„Dziś i jutro informatyki farmaceutycznej” — Maciej Paprocki	2	25
ICL 7700 (T.J.)	12	29	Wedle stawu grobla — Krystyn Bernatowicz	3	26
IKD 1980 (W.K.)	12	29	Czy „branzowy” stanie się „centralnym”? — Krystyn Bernatowicz	3	27
EuroJFP'79 — Stanisław Jaskólski	12	30	Zastostowanie komputerów w przemyśle (I.S.)	3	29
Z WIZYTA W EUROPEJSKICH OŚRODKACH			Komitet Informatyki NOT: alternatywa czy działanie na rzecz integracji? — Krystyn Bernatowicz	4	24
Jak pracuje SATA i POLORBIS — Jarosław Deminet	4	26	Klub Użytkowników Minikomputera MERA 400 (W.K.)	6	23
Stacja Tjaereborg — Marek Cichy	5	26	Udane tyle że nie „targi” (K.B.)	6	24
Uczelniane ośrodki — Janina Mincel	6	26	Szkoła Naukowa „Sieci Komputerowe” (B.M.)	6	25
Marzenia szalonego studenta — Jarosław Deminet	6	27	Informatyka w procesach zarządzania produkcją (K.B.)	7	21
PICTURE SYSTEM — Piotr Chrzastowski	7	24	III Fabryczny Rocznik Organizacji i Informatyki FSM (E.K.)	7	21
Czas podsumować — Marek Lao	7	24	I Ogólnopolska Gielda Pomysłów Racjonalizatorskich dla komputerów ODRA 1305 — M. Stronicki, M. Sajkowski	8—9	36
ZE ZJEDNOCZENIA INFORMATYKI			Minikomputery wychodzą w morze (K.B.)	8—9	36
LP SERWIS — oprogramowanie do rozwiązywania zadań programowania liniowego — Arkadiusz Gens	1	28	IV Ogólnopolskie Seminarium Metodyczne Informatyki — L. Profelski	8—9	39
Jubileusz i tęsknota do samodzielności — Krystyn Bernatowicz	1	31	Informatyka w Miejskich Zakładach Komunikacyjnych m.st. Warszawy	10	19
Dorobek ZETO Warszawa w komputeryzacji prac inżynierskich — Grzegorz Ziętara	2	28	„Doskonalenie eksploatacji systemów komputerowych zwiększa efektywność informatyki” — Tomasz Pawlak	10	22
Typowy system nadawania symboli Kodu Towarowo-Materiałowego — Jerzy Orłowski	2	29	Czytnik kart dziurkowanych w systemie minikomputerowym MERA 300 — Zbigniew Bronszewski, Ryszard Rawski, Zbigniew Tarczyński	10	24
System generacji programów obsługi wejścia SOWIK — Aleksy Brezcko	2	30	INFORMATYKA NA WOLI		
Za i przeciw systemom powtarzalnym — Artur Hajnicz	3	21	W blasku korzyści, w cieniu niedostatków — Krystyn Bernatowicz	11	20
Krajowa Konferencja DIEBOLDA — Władysław Klepacz	3	24	O informatyce na studenckiej naradzie (P.K.)	11	22
ZETO Łódź i OBRI Warszawa oferują program do obliczeń numerycznych — Andrzej Grandys, Leszek Namysłowski	4	23	O konferencji w Żaganiu (K. B.)	11	23
Współpraca z VVB Maschinelles Rechnen w NRD — Jan Goliński	5	28	Kolejna instalacja komputera IBM (L.G.)	11	24
Wynalazcy z ZETO Lublin — Ryszard Próchniak	5	30	ELORG seminarium SM EMC Polska 1979	12	23
Refleksje na stokach Kasprowego (K.B.)	5	31	GIELDA		
ZETO Łódź zwycięża we współzawodnictwie (W.K.)	5	31	Programy uniwersalne		
Tenis po raz drugi (K.B.)	6	31	Systemy minikomputerowe		
Socjalistyczne współzawodnictwo w przedsiębiorstwach ZETO — Władysław Klepacz	7	22	Systemy powtarzalne		
Szkolenie informatyczne w Zjednoczeniu Informatyki — Wacław Pankiewicz	8—9	34	Systemy minikomputerowe oprac. Andrzej Klimek	6	39
Działalność Zjednoczenia Informatyki w latach 1976—1979 — Andrzej Aranowski	10	26	MERA-ELWRO		
Świątokrzyski Związek Informatyków — I. Kowalik	10	29	Nowa wersja egzekutora dla komputera ODRA 1325		
Zastosowanie monitorów ekranowych w systemie gospodarki narodowej — Jan Cichoń, Janusz Czekał, Ryszard Jankowski	12	24	Informator szkoleniowy na rok 1979		
CENTRUM ETOB			Umowa serwisowa z PREDOM-ORG	1	38
Po 25 latach (wład)	5	20	Program informacyjny BZXX		
ETOB na „BUDEXPO” (wład)	5	22	Programy diagnostyczne DTLU i TRAS		
O cenach za wdrożenia systemów użytkowych (wład)	6	20	Program LPLX		
Nowy ośrodek w Zielonej Górze (K.B.)	7	27	Oprogramowanie komputerów ODRA		
Transmisja dla fabryki domów (wład)	8—9	40	Wymienność pakietów dyskowych		
ETOB Lublin — bliżej samodzielności (K.B.)	8—9	41	Nowi użytkownicy komputerów	2	33
Odnaczenia państwowe w sieci ETOB (wład)	10	32	Oprogramowanie komputerów JS		
Nowy ośrodek ETOB w Bielsku-Białej (wład)	10	33	Urządzenia pomocnicze dla ośrodków obliczeniowych		
ETOB Wrocław — dziewiątym samodzielnym przedsiębiorstwem (wład)	10	34	Zmiany konstrukcyjne w MTS 304-2		
Pochwała dla MERY 300 — Wincenty Łada	11	26	Umowa serwisowa MERA-ELWRO—MERA-ELZAB		
Współpraca polsko-rumuńska (wład)	12	32	Kto może być przyjęty na kursy?		
III samochodowy rajd integracji (wład)	12	33	Działalność Rady Kompleksowej Obsługi JS EMC	3	36
PORTRETY ZAWODOWE			Symulator pracy jednostki centralnej ODRA 1305 dla pamięci operacyjnej		
Krzysztof Zielak	5	23	Nowi użytkownicy komputerów		
Ewa Frąckowiak	6	22	Nowe kierunki szkolenia		
Juliusz Nalewajski	7	28	Oprogramowanie komputerów JS		
Jan Stepaniec	8—9	45	Wystawa Jednolitego Systemu w Moskwie	6	40
Eugeniusz Kubica	11	27	oprac. Jerzy Jankowski		
Z KRAJU					
INFOGRYF' 79	1	20	Kadry dla nowoczesnej informatyki (rozmowa z wiceministrem Nauki, Szkolnictwa Wyższego i Techniki, doc. dr inż. Walerym Kujawskim) — Krystyn Bernatowicz	4	1
Symposium i festiwal — Andrzej Klimek	1	21	Specjalne nagrody państwowe dla wybitnych uczonych	10	1
XXX-lecie Instytutu Matematycznego PAN i informatyki w Polsce — Czesław Olech	2	20			
Hybrydowy system DORNIERA — Roman Koniczyn	2	23			

Kompleksowa centralna obsługa techniczna i niezawodność czechosłowackich komputerów EC 1021

Wdrażanie systemu informatycznego jest jak wiadomo operacją bardzo złożoną i czasochłonną. Jednym z problemów, który trzeba rozwiązać przy wprowadzaniu takiego systemu, jest sprawna obsługa techniczna komputera.

W Czeskiej Republice Socjalistycznej (CSR) organizację obsługi technicznej sprzętu komputerowego (NOTO) powierzono przedsiębiorstwu państwowemu „Kancelářské Stroje” (nazywane dalej skrótem firmowym KSNP). Przedsiębiorstwo to już w 1972 r. stworzyło warunki organizacyjne zapewniające kompleksową centralną obsługę techniczną (KCTP) komputerów Jednolitego Systemu produkcji krajowej typu EC 1021. Producentem tych komputerów jest przedsiębiorstwo państwowe ZPA Čakovice.

Koncepcja KCTP jest wynikiem wielu doświadczeń związanych z eksploatacją komputerów JS EMC na terenie całego kraju. Podstawowe cele tej koncepcji to:

- maksymalne ograniczenie zatrudnienia pracowników wysoko kwalifikowanych (inżynierów i techników) — ma to duże znaczenie w CSRS ze względu na wyczerpane rezerwy kadrowe
- zmniejszenie zapasów części zamiennych i sprzętu pomiarowo-kontrolnego, niezbędnych dla uzyskania wysokiej sprawności eksploatacyjnej sprzętu komputerowego
- wprowadzenie metod przemysłowych do serwisu informatycznego, zapewniających ilościowe zwiększenie oraz poprawę jakości obsługi technicznej.

Podstawowe zadanie KCTP polega na zapewnieniu użytkownikom komputera EC 1021 obsługi technicznej na podstawie zawartych umów. Wykonanie przez NOTO tego trudnego zadania wymaga stopniowego tworzenia służb, które w sposób bezpośredni lub pośredni gwarantowałyby realizację koncepcji KCTP. Należało zorganizować: kadrę techniczną, działającą bezpośrednio przy instalacjach komputerowych u poszczególnych użytkowników, zaopatrzenie oraz gospodarkę magazynową częściami zamiennymi, laboratoria serwisu oraz wypożyczalnię sprzętu pomiarowo-kontrolnego itp. Poza tym koncepcja KCTP wymagała również zawarcia umów regulujących stosunki między dostawcami (producentami sprzętu komputerowego) a odbiorcą (NOTO). Warto też dodać, że tak pojęta obsługa techniczna nie mogła się obyć bez sprawnego wewnętrznego systemu informacyjnego, usprawniającego organizację przedsięwzięcia. Duże znaczenie dla owej organizacji ma maszynowe przetwarzanie niektórych agend działalności KCTP.

Po sześciu latach stopniowego rozwoju kompleksowej centralnej obsługi technicznej można obecnie zbilansować i ocenić osiągnięte wyniki.

NOTO obejmuje aktualnie usługami KCTP 75 komputerów EC 1021, co stanowi ok. 70% łącznej liczby komputerów tego typu eksploatowanych na terenie Czeskiej Republiki Socjalistycznej.

Jeśli chodzi o ograniczenie zatrudnienia, uzyskano godne odnotowania efekty. Konserwacja techniczna komputera przy jednoczesnej eksploatacji, z uwzględnieniem wszystkich możliwych specjalności wynikających z konfiguracji komputera i przeciętnej planowanej absencji pracowników (urlopy, choroby), wymaga 4–5 pracowników. Natomiast w przypadku organizacji KCTP niezbędni są najwyżej dwaj pracownicy.

Zmniejszono również znacznie zapasy części zamiennych. Wartość pieniężna zapasów części zamiennych w przeliczeniu na 1 komputer wynosi obecnie w KCTP ok. 3,5% kosztu zakupu całej konfiguracji komputera, przy czym z zapasów tych korzysta się także w przypadku napraw gwarancyjnych, które nie są objęte obsługą kompleksową. Dla porównania można przytoczyć, że pierwszy system komputerowy EC 1021 wyeksportowany do Polski (jego obsługę techniczną zapewniał sam użytkownik) został wyposażony w komplet części zamiennych o wartości ok. 10% kosztu zakupu konfiguracji. Przyjęcie w CSRS podobnego wskaźnika dla wszystkich 75 komputerów obsługiwanych w systemie KCTP spowodowałoby wzrost wartości zapasów

części zamiennych o niebagatelnej kwocie około 64 930 tys. Kčs.

Uprzemysłowienie obsługi technicznej polega przede wszystkim na utworzeniu zaplecza technicznego KCTP w postaci sieci laboratoriów serwisu, gdzie szybko i na wysokim poziomie fachowym wykonuje się naprawy wymiennych modułów i niektórych podzespołów elektromechanicznych. Większe naprawy urządzeń (remont średni lub generalny) odbywają się bezpośrednio u producenta, a w tym czasie NOTO wypożycza użytkownikom związanym umową obsługi kompleksowej inne sprawne urządzenia zapewniające ciągłość działania ośrodka obliczeniowego. Utworzenie takiego zaplecza technicznego dla jednego, nawet największego systemu komputerowego przez użytkownika nie miałyby żadnych szans realizacji ze względu na rachunek ekonomiczny.

Obsługa KCTP, zagwarantowana przez dostawcę sprzętu (KSNP) ma też inne zalety. Jedną z nich jest znaczny wpływ na niezawodność eksploatacyjną komputera — i to w sposób bardziej efektywny, niż to może osiągnąć pojedynczy użytkownik dysponujący własną obsługą techniczną.

Dane o niezawodności eksploatacyjnej jakiegokolwiek wyrobu mają wartość użytkową jedynie wtedy, jeżeli uzyska się je bezpośrednio w eksploatacji tego wyrobu w warunkach rzeczywistych i w zastosowaniu, do którego jest on przeznaczony. W wyniku umów zawartych pomiędzy KSNP a użytkownikami komputer EC 1021 na obsługę kompleksową obowiązki dostawcy nie kończą się z chwilą przekazania komputera do eksploatacji i z upływem okresu gwarancyjnego, lecz praktycznie dopiero zaczynają się. W umowach obsługi kompleksowej ustalona jest również minimalna gwarantowana sprawność eksploatacyjna komputera. Zwiększa to nie tylko formalne ale i rzeczywiste zainteresowanie KSNP związane z zapewnieniem oraz stałym analizowaniem niezawodności eksploatacyjnej tych komputerów, a poprzez znajomość danych o tej niezawodności — z fachowym podejmowaniem odpowiednich przedsięwzięć organizacyjno-technicznych zarówno wewnątrz własnej organizacji, jak i w ramach stosunków odbiorca dostawca, ukierunkowanych na produkcję i właściwą bazę naukowo-badawczą. W ten sposób powstaje bardzo efektywne sprzężenie zwrotne z obszaru wykorzystania techniki obliczeniowej do sfery rozwoju i produkcji komputerów.

Dzięki całemu zespołowi przedsięwzięć udało się usunąć większość istotnych problemów, jakie ujawniały się w toku instalacji i eksploatacji komputerów EC 1021 do końca 1975 roku. Wynikiem ścisłej współpracy pomiędzy NOTO a zapleczem badawczo-rozwojowym i producentem jest wiele innowacji w poszczególnych urządzeniach systemu komputerowego EC 1021.

Niezawodność eksploatacyjna jest z reguły definiowana za pomocą następujących czterech podstawowych parametrów:

- średniego czasu międzyawaryjnego
- średniego czasu naprawy
- współczynnika wykorzystania technicznego
- współczynnika gotowości technicznej.

W obsłudze kompleksowej dane o niezawodności eksploatacyjnej zbierane są ze wszystkich instalacji EC 1021 i przetwarzane z reguły co miesiąc. Aktualną niezawodność eksploatacyjną komputerów EC 1021 określa współczynnik wykorzystania technicznego $K_{tet} = 84,6\%$ oraz współczynnik gotowości technicznej $K_p = 94,1\%$. Są to wartości, które w pełni zadowolają użytkowników.

Warto jeszcze dodać, że NOTO przewiduje objęcie kompleksową obsługą również komputera generacji 3,5 produkcji czechosłowackiej typu EC 102 — z myślą o tym rozpoczęto już odpowiednie przygotowania.

Jaroslav MILDORF

EO/1405/K/79



SEMINARIUM SM EMC POLSKA 1979

W przestronnej Sali Rycerskiej odrestaurowanego zamczyska w konińskim Uniejowie odbyło się w październiku czterodniowe seminarium promocyjne¹⁾ radzieckiego zjednoczenia „Elektronorgtechnika”, w krajach RWPG znanego pod skrótem telegraficznym ELORG²⁾.

W seminarium uniejowskim wzięło udział 87 informatyków polskich, reprezentujących 52 krajowe przedsiębiorstwa, uczelnie i inne instytucje zainteresowane szybkim importem możliwie nowoczesnego sprzętu telekomputerowego. Oferta ELORG-owska stwarza wyjątkowo pomyślne perspektywy, zwłaszcza z chwilą utworzenia w Warszawie składnicy konsygnacyjnej części zamiennych do minikomputerów radzieckich i uruchomienia wyspecjalizowanego serwisu instalacyjno-konserwatorskiego.

Referat wprowadzający w problematykę bieżącą i przyszłościową systemów minikomputerowych wygłosił dr Walentin Piotrowicz Sjomik, kierownik Wydziału Oprogramowywania Minikomputerów Serii SM w moskiewskim Instytucie Elektronicznego Maszyn Sterowniczych (INEUM).

W ramach problematyki dotyczącej maszyn SM-3 i SM-4 w szczególności omówiono:

- charakterystykę ogólną tych maszyn
 - ich architekturę i oprogramowanie
 - zasady tworzenia instalacji użytkowych („kompleksów”) i ich rozszerzania zgodnego z linią rozwojową serii SM
 - systemy operacyjne do pracy w czasie rzeczywistym.
- Sporo czasu poświęcono komputerom typu M-4030-1 (8 referatów), a w szczególności omówieniu:
- podstawowej charakterystyki jednostek centralnych
 - architektury systemów liczących i możliwości ich rozbudowy perspektywicznej
 - dyskowego systemu operacyjnego DOS-ASWT
 - sprzętu teleprzewarzanowego
 - systemu IRIS zarządzania bazami danych i

¹⁾ Kilka dni później podobne seminarium odbyło się w Polańczyku — dla ponad 80 uczestników z ponad 30 instytucji. Współorganizatorem był ETOB Rzeszów

²⁾ Wszeczchwiazkowe Zjednoczenie V/O ELEKTRONORGTECHNIKA utworzyło przy Przedstawicielstwie Handlowym ZSRR w Warszawie swoje Centrum Techniczne ELORG; od pewnego też czasu mikrobusy z napisem ELORG-SERVICE można spotkać na ulicach polskich miast, w których instalowane są radzieckie minikomputery

● pakietów programów użytkowych, zwracając przy tym uwagę na możliwość adaptacji oprogramowania „riadowskiego”.

Wymowną ilustracją omawianych zagadnień był kolorowy film o oferowanych minikomputerach. Największe wrażenie na uczestnikach seminarium wywarło jednak demonstrowanie „na żywo” zdalnego dostępu do przykładowej bazy danych, zawierającej informacje o uczestnikach seminarium. Pokaz ten wymagał udostępnienia specjalnej linii na trasie: Zamek — poczta w Uniejowie — Turek — Konin — Warszawa.

Trzecia, stosunkowo najmniej liczna grupa referatów dotyczyła innych minikomputerów serii SM, tj. maszyn typu SM-1 i SM-2, importowanych od pewnego czasu do naszego kraju.

Wreszcie czwarta grupa referatów dotyczyła takich zastosowań omawianego sprzętu minikomputerowego, jak np.:

- zautomatyzowane miejsce pracy („ZMP”)
 - CAMAC-owskie pomiarowe kompleksy obliczeniowe („PKO”)
 - architektura instalacji użytkowych.
- Przedstawiciele wszystkich instytucji uczestniczących w seminarium, otrzymali komplety materiałów źródłowych — na razie w języku rosyjskim i częściowo angielskim. W najbliższej przyszłości można oczekiwać przetłumaczenia tych materiałów na język polski.

Należy szczerze przyklasnąć idei seminariów promocyjnych dla potencjalnych użytkowników. Tym bardziej, że strona organizacyjna imprezy uniejowskiej była wręcz wzorowa — niewątpliwie dzięki wysiłkowi współorganizatora, jakim był Ośrodek Szkolenia Informatycznego ZETO-Lódź.

Można mieć nadzieję, że seminaria ELORG-owskie staną się zaczątkiem nowej formy „ruchu informatycznego” między Bugiem a Odrą. W szczególności można oczekiwać:

- utworzenia polskiego Klubu Użytkowników SM — na wzór takiego zrzeszenia istniejącego już w Czechosłowacji
- zorganizowania centralnego szkolenia przyszłych polskich użytkowników minikomputerów serii SM na kursach krajowych prowadzonych przez licencjonowanych wykładowców z czołowych ośrodków komputerowych
- okresowych konferencji metodycznych i problemowych w ZSRR, staży zawodowych dla wykładowców, spotkań instruktorów z dawnymi kursantami itp. imprez, których konkretny kształt określi przyszła praktyka.

Adam B. EMPACHER

Posiedzenie grupy roboczej IFIP w Warszawie

W dniach od 15 do 18 października br. odbyło się w Warszawie kolejne doroczne posiedzenie grupy roboczej WG 2.2 Międzynarodowej Federacji Przetwarzania Informacji (IFIP).

Pełna nazwa grupy WG 2.2 „Opis formalny koncepcji programowania” (ang. *Formal description of programming-concepts*) zwięźle wyjaśnia podstawowy zakres jej zainteresowań i bieżącej działalności.

W bardziej szczegółowym, statutowym sformułowaniu, celem głównym działalności tej grupy jest „...rozwój koncepcji programowania poprzez badania naukowe, eksperymentowanie i porównywanie różnych modeli formalnych tych koncepcji, przy czym modele te powinny uwzględniać:

- aktualny poziom rozwoju techniki komputerowej
- dostateczny stopień uogólnienia opisu całych systemów lub podsystemów użytkowych
- specyfikację problemu lub rozwiązania
- praktyczne wskazówki odnośnie do zakresu możliwości funkcjonalnych, bezbłędności, równoważności, implementacji i wydajności
- rozwój normalizacji i techniki specyfikacji
- potrzeby dydaktyki...”

Należy wyjaśnić, że grupa WG 2.2 działa już prawie 15 lat (od 1965 r.) i wchodzi w skład Komitetu Technologicznego nr 2 IFIP „Programowanie”, chronologicznie najstarszej agencji działalności podstawowej Federacji (utworzonej w 1962 r.). Grupa liczy obecnie 45 członków, którymi są wybitni naukowcy z 15 krajów. Polskę reprezentują profesorowie A. Blikle i A. Mazurkiewicz z Instytutu Podstaw Informatyki PAN. Aktualnym przewodniczącym grupy jest prof. E. J. Neuhold, kierownik katedry oprogramowania w Instytucie Informatyki Uniwersytetu Stuttgartskiego (RFN), który również przewodniczył obecnemu spotkaniu. W spotkaniu warszawskim uczestniczył także sekretarz WG 2.2 prof. E. K. Blum z Wydziału Matematyki Uniwersytetu Południowej Kalifornii (USA), który oprócz pełnienia swych obowiązków formalnych był autorem jednego z wygłoszonych referatów.

Miejscem posiedzeń spotkania warszawskiego była Sala Lustrzana Pałacu Staszica, a ich głównymi tematami były teoretyczne problemy współczesnego rozwoju języków programowania oraz bezbłędności programów. W ciągu czterech dni obrad wygłoszono łącznie 16 referatów, spośród których w 10 przypadkach autorami byli naukowcy polscy. Fakt lokalizacji spotkania grupy WG 2.2 w Warszawie, a także znaczna liczba referatów polskich świadczą o dużej aktywności naszych naukowców oraz o zdobyciu przez nich znaczącej pozycji w obszarze badań o kluczowym znaczeniu dla perspektywicznego rozwoju informatyki. (W.K.)

Zastosowanie monitorów ekranowych w systemie gospodarki materiałowej

W zakładzie Elektronicznej Techniki Obliczeniowej w Zielonej Górze opracowano i wdrożono dla Zaodrzańskich Zakładów Przemysłu Metalowego ZASTAL podsystem EWIG-MONITOR, który reprezentuje nowy typ podejścia do rozciągania informatycznego systemu gospodarki materiałowej. EWIG-MONITOR stanowi uzupełnienie eksploatowanego już przez ZASTAL na komputerze ODRA 1305 w ZETO systemu gospodarki materiałowej EWIG. Uzupełnienie to zostało zrealizowane w oparciu o węgierskie monitory ekranowe VTS 56100 (nowsza wersja monitorów VIDEOTON 340), które zostały zakupione przez ZASTAL.

Dla zapewnienia użytkownikowi bezpośredniego i bieżącego dostępu do danych ewidencyjnych wprowadzono tzw. Dyspozycyjną Kartotekę Materiałową (DKM), aktualizowaną w cyklu dziennym. Łączność między systemem EWIG, a podsystemem EWIG-MONITOR zapewniają dwa programy, z których pierwszy na początku każdego miesiąca tworzy w DKM zapisy stanów początkowych na podstawie kartoteki systemu EWIG, natomiast drugi wyprowadza w końcu miesiąca sprawozdanie zbioru transakcji za ten okres

przeznaczony do zaewidencjonowania w systemie EWIG.

Uproszczony schemat przetwarzania danych w podsystemie EWIG-MONITOR przedstawia rysunek 1. W ramach podsystemu realizowane są następujące operacje:

- wprowadzanie danych z dokumentów obrotu materiałowego z jednoczesną ich kontrolą (na pierwszej zmianie)
- codzienna aktualizacja stanów w Dyspozycyjnej Kartotece Materiałowej (na drugiej i trzeciej zmianie)
- wyświetlanie na monitorach ekranowych danych o stanach i obrotach materiałowych w układach żądanych przez użytkowników (następnego dnia na pierwszej zmianie)
- korygowanie błędnych danych w zapisach DKM bezpośrednio z klawiatury monitora (na pierwszej zmianie).

Równoczesne wykonywanie funkcji wprowadzenia danych oraz wyświetlania lub korygowania danych w DKM jest możliwe dzięki zainstalowaniu dwóch monitorów przyłączonych do systemu komputerowego za pośrednictwem dwóch jednostek sterujących typu JSG 7801.

PROGRAM WPROWADZANIA DANYCH

Wprowadzane z klawiatury monitorów ekranowych dane z dokumentów obrotu materiałowego zapisywane są po ich skontrolowaniu do zbioru transakcji w pamięci dyskowej. Z uwagi na możliwość awarii urządzeń peryferyjnych, przewidziano w takich przypadkach zastąpienie monitora ekranowego czytnikiem kart dziurkowanych, a pamięci dyskowej — pamięcią taśmową.

W porównaniu z rozpowszechnianym przez ELWRO Systemem Przygotowania Danych (SPD) omawiany program wykazuje następujące zalety:

- łatwość operowania monitorem dzięki użyciu minimalnej liczby prostych dyrektyw
- możliwość eliminowania większości błędów już w fazie wprowadzania dzięki wbudowaniu aparatu kontroli formalnej
- dobre zabezpieczenie danych w zbiorze transakcyjnym (awaria systemu w czasie pracy programu nie ma wpływu na późniejszy dostęp do danych).

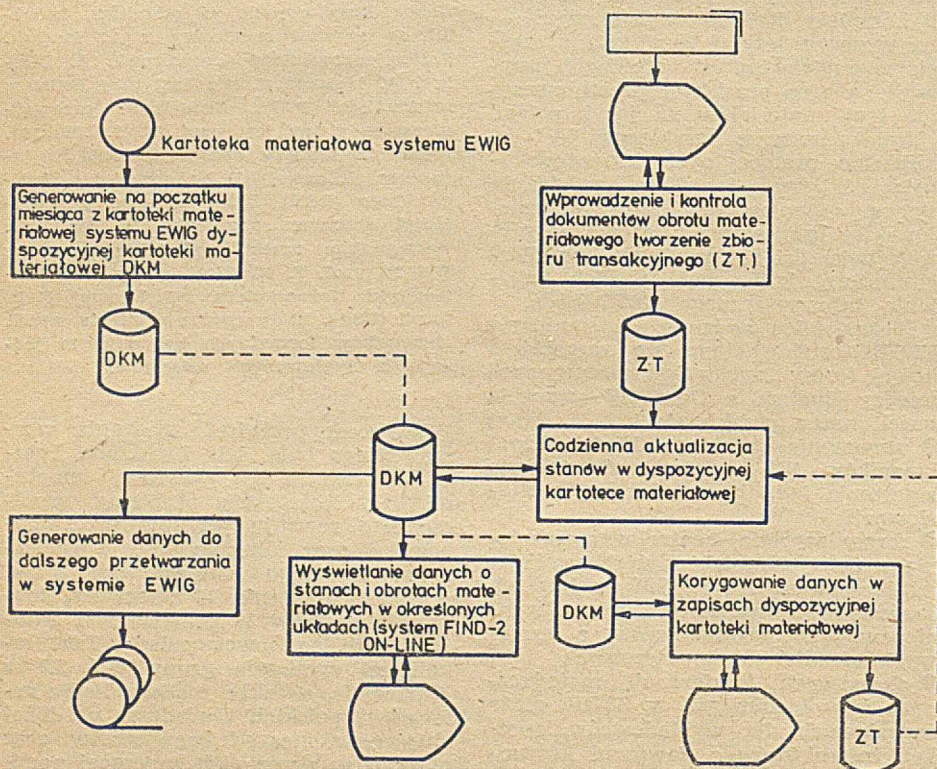
Operator monitora może współpracować z programem dwóch podstawowych trybów: wprowadzania oraz weryfikacji danych. Poszczególni operatorzy używają tych trybów niezależnie od siebie, przy czym każdy z nich tworzy lub weryfikuje określoną partię danych. Ogólne zasady pracy w obu trybach przedstawia rysunek 2.

Dane są wprowadzane w postaci ciągu znaków zgodnego z ustalonym rozplanowaniem informacji na dokumencie. Wprowadzone dane są następnie kontrolowane pod względem:

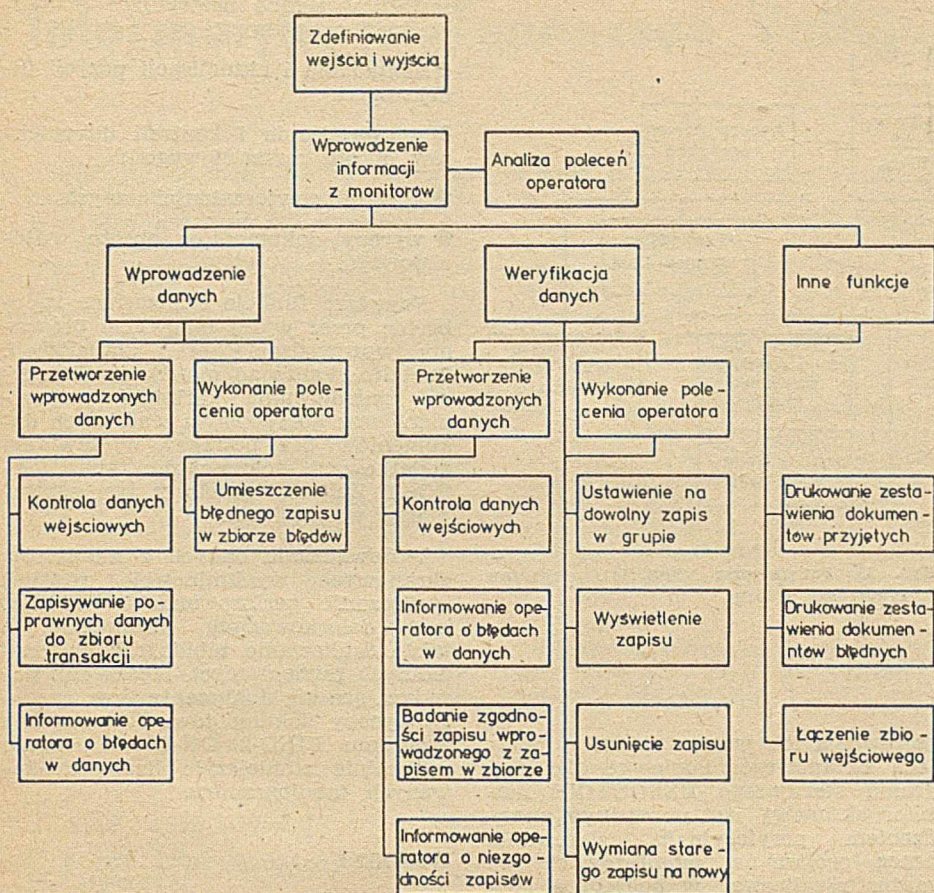
- wypełnienia pól obligatoryjnych
- występowania właściwych kombinacji znaków
- zgodności wartości pól z ustalonymi przedziałami
- powiązań między określonymi polami.

Dokumenty poprawne są automatycznie zapisywane do zbioru transakcyjnego. Jeżeli wprowadzony dokument zawiera błędy, na ekranie monitora wyświetlane są oznaczenia błędnych pól dokumentów. Dla różnych typów błędów przewidziano odpowiednie rodzaje oznaczeń. Operator może poprawić błędne pola natychmiast lub zapisać wadliwy dokument w zbiorze błędów.

Przy weryfikacji operator wprowadza dane analogicznie jak podczas tworzenia grupy danych w zbiorze. Program przeprowadza kontrolę formalną zapisu, sygnalizując wykryte błędy na



Rys. 1. Ogólny schemat przetwarzania w podsystemie EWIG-MONITOR



Rys. 2. Podsystemowe funkcje programu wprowadzającego

ekranie monitora. Jeżeli zapis jest poprawny, następuje porównanie z odpowiednim zapisem zbioru transakcji. W przypadku ewentualnej niezgodności tych zapisów, pojawia się również odpowiednia informacja dla operatora.

W trybie weryfikacji zapisy ze zbioru transakcyjnego można również sprawdzać wzrokowo, powodując w tym celu ich wyświetlenie na monitorze. Możliwe jest także usunięcie zapisu ze zbioru transakcyjnego lub zastąpienie go zapisem ponownie wprowadzonym. Ciąg powyższych operacji można rozpocząć od dowolnego zapisu w partii danych.

Oprócz powyżej opisanych czynności program może wykonać łączenie zbiorów transakcyjnych utworzonych podczas różnych przebiegów. Możliwe jest również otrzymywanie zestawień dokumentów przyjętych i błędnych w zredagowanej postaci.

PROGRAM AKTUALIZACJI DYSPOZYCYJNEJ KARTOTEKI MATERIAŁOWEJ

Utworzony przez program wprowadzania danych zbiór transakcyjny służy do aktualizacji Dyspozycyjnej Kartoteki Materiałowej. Przebieg ten jest uruchamiany codziennie na drugiej zmianie. Umożliwia to wyprowadzanie na-

stępne go dnia rano aktualnych informacji o stanach magazynowych materiałów. Program aktualizacji DKM tworzy na wyjściu zarówno zbiór dyskowy, jak i jego kopię taśmową, służącą do zabezpieczenia ciągłości działania systemu na wypadek niemożliwości korzystania z wersji dyskowej.

PROGRAM KORYGOWANIA I WYŚWIETLANIA ZAPISÓW DKM

Podstawowe funkcje programu korygowania i wyświetlania zapisów DKM przedstawia rysunek 3. Operator monitora może pracować w jednym z dwóch trybów: korygowania danych w zapisach lub wyświetlania zapisów z DKM. Operatorzy pracują niezależnie od siebie, lecz tylko jeden z nich może w danej chwili korygować zawartość DKM.

Program korzysta z dyskowej wersji DKM. Ponieważ kartoteka ta ma organizację sekwencyjną, na początku wykonywania programu tworzona jest mapa zbioru, umożliwiająca szybkie odnalezienie zapisu z żadaną wartością klucza.

W celu korygowania zawartości DKM operator wprowadza z monitora wartość klucza zapisu, który zamierza zmienić. W odpowiedzi program wyświetla żądany zapis w zredagowanej postaci wraz z nazwami pól. Operator

może zmienić zawartość określonych pól po czym wprowadza zmieniony zapis. Program kontroluje pole zmienione w tym samym zakresie, co program wprowadzania, a ponadto sprawdza zgodność zmienionego zapisu ze strukturą kartoteki. Jeżeli wykryte zostaną błędy, program informuje o tym operatora.

W przypadku, gdy pola kluczowe nie zostaną zmienione, zmieniony zapis wprowadzany jest z powrotem do kartoteki. Jeżeli pola kluczowe uległy zmianie, zapis dotychczasowy zostaje z kartoteki usunięty, a zapis zmieniony staje się podstawą do utworzenia odpowiedniej transakcji, która będzie wprowadzona do DKM w procesie aktualizacji. Operator może również usunąć zapis z kartoteki bez tworzenia nowej transakcji.

W celu wyświetlania zapisów z DKM, operator deklaruje wartość klucza zapisu, od którego dane z kartoteki DKM mają być wyświetlane. Program wyprowadza zapisy w postaci zredagowanej wraz z nazwami pól. Równocześnie na ekranie wyświetlone są 4 kolejne zapisy. Operator może w dowolnym momencie zakończyć pracę lub podać nową wartość klucza. W tym trybie może pracować wielu operatorów, jednak bez możliwości zmiany zawartości DKM.

Na zakończenie pracy program korygowania i wyświetlania DKM drukuje zestawienie wprowadzonych zmian, który zawiera:

- datę i czas zmiany oraz hasło operatora
- treść zapisów przed i po zmianie
- treść zapisów usuniętych
- treść zapisów wprowadzonych.

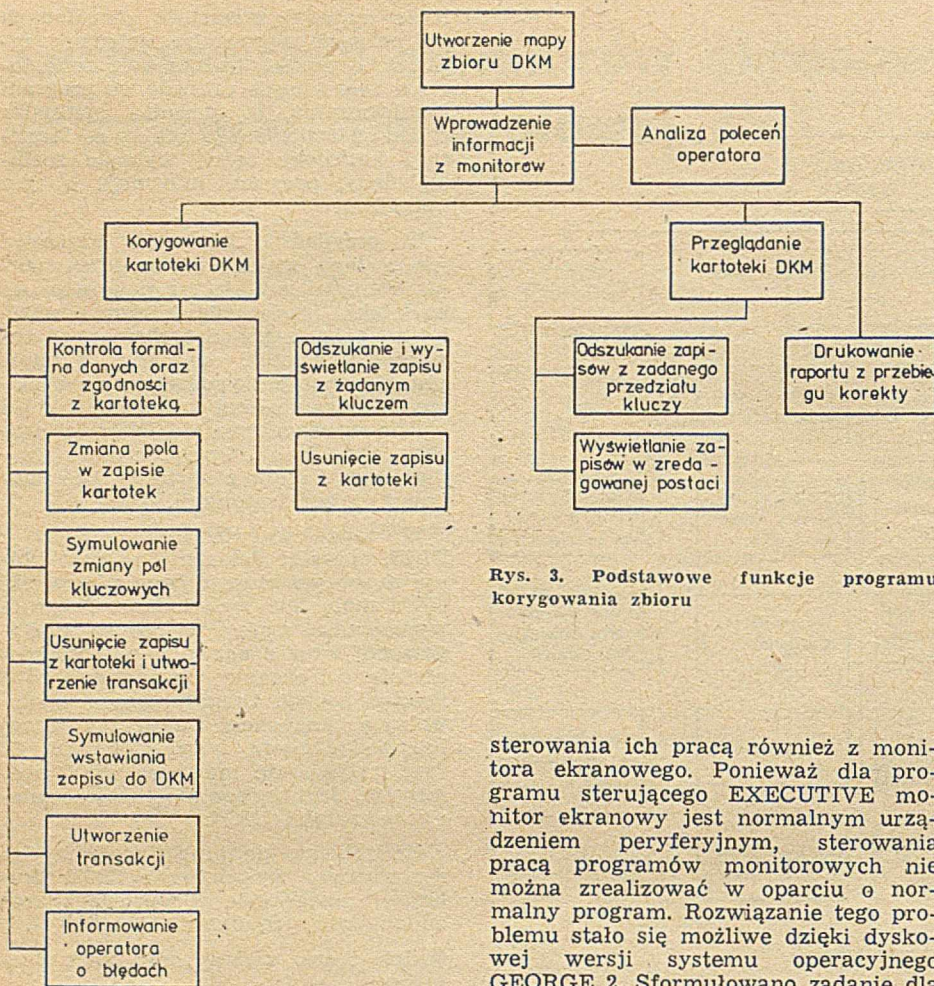
WYŚWIETLANIE SPRAWOZDAŃ Z DKM

Dla otrzymania potrzebnych zestawień danych o obrotach i stanach materiałowych wykorzystano standardowy program FIND-2 ON-LINE. Program ten umożliwia wyszukiwanie i wyprowadzanie informacji ze zbiorów po wprowadzaniu zestawu parametrów, które szczegółowo określają przebieg i zakres wyszukiwania. Zestawy parametrów mogą być wprowadzane z monitorów lub ściągane pojedynczą dyrektywą ze zbioru w pamięci dyskowej.

Oryginalną wersję programu FIND-2 ON-LINE charakteryzują następujące wady:

- brak możliwości modyfikowania dostarczonych z pamięci dyskowej parametrów, co ogranicza ich zastosowanie do tych jedynie przypadków, których warunki wyszukiwania nie ulegają zmianie
- przy wyprowadzaniu danych na monitor każdy wiersz przesyłany jest oddzielną transmisją.

W przypadku monitora VTS 56100 (również VT 340) stanowi to istotną wadę, gdyż każda transmisja z programu do monitora wymaga stałego czasu ok. 5 s, potrzebnego do ustalenia



Rys. 3. Podstawowe funkcje programu korygowania zbioru

położenia pól chronionych, co oczywiście znacznie wydłuża proces wprowadzania informacji na monitor.

W celu uniknięcia powyższych ograniczeń program FIND-2 ON-LINE został zmodyfikowany. Obecnie możliwe jest już podczas ściągania z pamięci dyskowej uzupełnienie parametrów o podane z monitora aktualne wartości pól zmiennych. W ten sposób możemy wykorzystując ten sam zestaw parametrów uzyskać dane na przykład z różnych okresów lub o różnych wartościach identyfikatorów.

Przyspieszenie wyświetlania informacji na monitorze uzyskano w ten sposób, że zawartość całego ekranu wprowadzona jest jedną transmisją.

STEROWANIE PRACĄ PROGRAMÓW MONITOROWYCH

Zbiór parametrów w pamięci dyskowej, tworzony i aktualizowany standardowym programem XMED, może zawierać wiele ich zestawów. Daje to bogate możliwości otrzymywania sprawozdań w różnych układach z różnych systemów. Sprawozdania te mogą być wyprowadzane zarówno na ekran monitora jak i na drukarkę wierszową.

W przypadku wykonywaniu kilku programów korzystających z monitorów ekranowych zachodzi potrzeba

sterowania ich pracą również z monitora ekranowego. Ponieważ dla programu sterującego EXECUTIVE monitor ekranowy jest normalnym urządzeniem peryferyjnym, sterowania pracą programów monitorowych nie można zrealizować w oparciu o normalny program. Rozwiązanie tego problemu stało się możliwe dzięki dyskowej wersji systemu operacyjnego GEORGE 2. Sformułowano zadanie dla GEORGE 2, polegające na przemienym wykonywaniu użytkowych programów monitorowych oraz specjalnego programu sterującego. Program ten zawiera listę programów użytkowych i sposoby ich uruchamiania. Komunikując się z operatorem monitora ekranowego, program odbiera dyspozycje dotyczące uruchomienia jednego z programów użytkowych. Dyspozycja ta zostaje przekształcona w odpowiedni komunikat (zdarzenie dla systemu operacyjnego), który jest analizowany przez GEORGE 2. Żądany program użytkowy jest następnie wprowadzany i wykonywany. Po zakończeniu wykonywania następuje ponowne ściągnięcie programu sterującego. Wykorzystanie GEORGE 2 umożliwia nie tylko wykonanie łańcucha programów monitorowych, ale również zapewnia odpowiednią reakcję systemu na występujące w programach zdarzenia. Działanie operatora komputera ogranicza się do jednorazowego uruchomienia programu sterującego. Dla umożliwienia współpracy z programami monitorowymi został również odpowiednio zmodyfikowany system operacyjny GEORGE 2.

EFEKTY WDROŻENIA PODSYSTEMU EWIG-MONITOR

Wdrożenie podsystemu EWIG-MONITOR spowodowało zmianę w sposobie eksploatacji systemu gospodarki materiałowej EWIG. Z eksploatacji

zostały wycofane następujące dotychczasowe programy:

- zakładania i aktualizacji pozycji indeksowych
- wprowadzania i kontroli dokumentów obrotu materiałowego
- kontroli powtarzalności danych
- wyceny dokumentów obrotu materiałowego.

Powyższe funkcje realizowane są obecnie przez wyżej omówione programy wprowadzania oraz aktualizacji. Ponadto stworzono możliwość bieżącego kontrolowania kartoteki magazyniera oraz korygowania wadliwych dokumentów bez potrzeby wystawiania dodatkowych dokumentów typu „RAPORT ZMIAN” (około 2 tys. egzemplarzy w miesiącu).

Wprowadzanie danych z monitorów ekranowych wyeliminowało również konieczność perforowania 12—15 tys. kart dziurkowanych miesięcznie, a także drukowanie tabulogramów kontrolnych zawierających błędne lub powtarzające się dokumenty oraz raporty wyceny dokumentów. Zastosowanie programu FIND-2 ON-LINE ponadto radykalnie zmniejszyło liczbę drukowanych tabulogramów.

WNIOSKI

Doświadczenia uzyskane w czasie opracowywania podsystemu EWIG-MONITOR potwierdziły fakt, że monitory ekranowe zapewniają realizację wszystkich wymienionych na wstępie funkcji.

Po przeanalizowaniu doświadczeń zebranych w okresie kilkumiesięcznej eksploatacji tego podsystemu stwierdzono ponadto, że w przypadku programów wyszukiwujących i wyprowadzających informacje oraz korygujących dane w kartotece — wykorzystanie monitorów jest w pełni zadowalające. Natomiast funkcja wprowadzania i kontroli danych, mimo że zapewnia duży stopień poprawności wprowadzonych danych, może być zrealizowana tylko w ograniczonym zakresie. Przyczyn należy szukać zarówno w ograniczonej szybkości monitora, niefunkcjonalnym układzie klawiatury monitora VTS 56100, jak również stosunkowo powolnym przesyłaniu danych pomiędzy monitorem, a jednostką centralną. Pewnym ograniczeniem programu wprowadzającego jest konieczność wprowadzania danych w postaci ciągu znaków. W najbliższej przyszłości ten sposób wprowadzania danych zostanie zastąpiony przez formatowanie danych w oparciu o wykorzystanie pól chronionych.

Obecnie prowadzone są prace, które mają na celu standaryzację opisanych rozwiązań, co umożliwi ich zastosowanie także w innych systemach.

Jan CICHON, Janusz CZEKAJ,
Ryszard JANKOWSKI
ZETO Zielona Góra

Informatyka w Tennessee Valley Authority

Wielkość zasobów, jakimi rozporządzają wielkie organizacje, sama skala tych obiektów oraz ich względnie duża elastyczność — to przyczyny, dla których organizacje takie są doskonałymi poletkami doświadczalnymi do testowania wszelkich innowacji i poszukiwania odpowiednich sposobów ich wykorzystania. Dotyczy to zwłaszcza zastosowań informatyki. W istocie, wielkość i wielość procesów informacyjno-decyzyjnych, możliwość ich organizacyjnego ogarnięcia oraz ustalenia ich efektywności są wystarczającymi podstawami do uznania wielkich organizacji za probież użyteczności informatyki w jej różnych formach sprzętowo-programowych i organizacyjnych.

Powyższą problematykę ilustruje przykład rozwoju informatyki w wielkiej państwowej organizacji gospodarczej w Stanach Zjednoczonych. Orga-

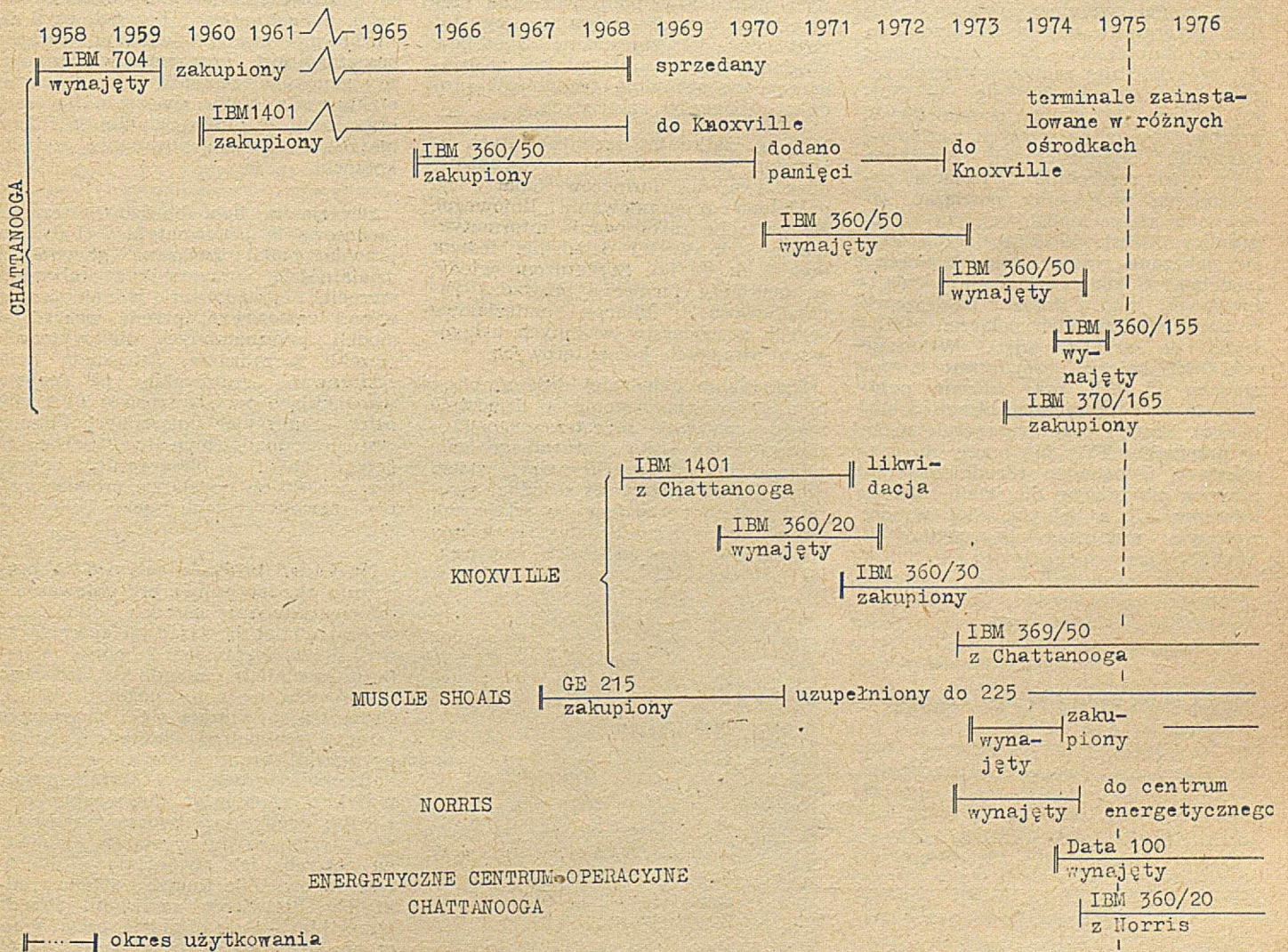
nizacją tą jest Tennessee Valley Authority (TVA) — Zarząd Doliny Tennessee, powołany decyzją prezydenta Roosevelta w 1932 r. jako jedna z propozycji Nowego Ładu (New Deal). Miało to być pierwsze duże przedsiębiorstwo federalne, specjalizujące się w regulacji systemu wodnego oraz w sprzedaży taniej energii elektrycznej z elektrowni wodnych na rzece Tennessee. Obecnie TVA zatrudnia ok. 20 tys. pracowników, dysponuje mocą ponad 23 tys. MW energii elektrycznej oraz prowadzi bardzo różnorodną działalność społeczną i gospodarczą w wielu dziedzinach. Dyrekcja TVA¹⁾ podlega bezpośrednio prezydentowi (poprzez sekretariat stanu) oraz senatowi USA.

¹⁾ Dyrekcja składa się z pięcioosobowej rady dyrektorów oraz podlegającego jej dyrektora zarządzającego

ORGANIZACJA

Z wymienionych na rysunku ośrodków Knoxville jest siedzibą władz TVA, Chattanooga jest głównym ośrodkiem operacji wodnych i energetycznych, zaś Muscle Shoals i Norris mają znaczenie pomocnicze.

Ponad 90% budżetu TVA jest związane z produkcją energii elektrycznej, toteż główne moce obliczeniowe zainstalowano w Chattanooga i dział energetyczny był ich dominującym użytkownikiem. Dlatego większość sprzętu, włącznie z pierwszym IBM 704, została zakupiona z funduszy działu energetycznego. Drugim pod względem ważności użytkownikiem było biuro projektów TVA (Dział Projektowania Technicznego i Konstrukcyjnego), gdzie wykorzystywano standardowe pakiety zastosowań inżynierskich.



Rozwój bazy sprzętowej w TVA

Tabela 1. Udział poszczególnych dziedzin działalności TVA w użytkowaniu zasobów informatycznych według podstawowych kosztów [proc.]

	Energia	Biuro projektów	Finanse	Inne
Czas pracy i zajętość pamięci komputerów	34,4	21,9	16,1	27,6
Płace personelu informatycznego	40,0	14,0	17,0	26,0
Całkowity koszt użytkowania	33,3	20,1	14,6	29,0

Od wczesnych lat sześćdziesiątych dość dużą wagę przywiązywano do prostych zastosowań (listy płac, automatyczne wystawianie rachunków odbiorcom energii). W dziedzinie bardziej wyrafinowanych zastosowań informatyki (systemów optymalizacyjnych działających w czasie rzeczywistym, modeli symulacyjnych itp.) również przeważało nastawienie pragmatyczne. Zainteresowanie i zrozumienie dla tych bardziej zaawansowanych zastosowań ograniczało się w dużej mierze do grupy podstawowych pracowników z najniższych poziomów hierarchii organizacyjnej. Wynikiem tego było rozdrobnienie i izolacja pojedynczych prac, brak powiązań zarówno w sferze modelowej, jak i programowej. Ilustruje to tabela 2, w której pokazano gęstość połączeń między modelami komputerowymi (zdefiniowaną jako iloraz istniejących połączeń i wszystkich połączeń możliwych do zrealizowania). Wskazane wielkości należy porównywać z poziomem 12—15%, przy którym pojawia się konieczność organizacji systemowej. Trzeba przy tym pamiętać, że w przypadku TVA nie mamy do czynienia z usługowym ośrodkiem obliczeniowym pracującym dla wielu różnych klientów, ale z jednym, choć wysoce złożonym organizacyjnie użytkownikiem.

Tabela 2. Gęstość powiązań wewnątrz grup zastosowań modelowych i pomiędzy nimi

—	—	—	5%	0%
—	3%	—	2%	0%
—	0%	4%	5%	0%
6%	2%	5%	6%	1%
0%	0%	0%	1%	2%

Gęstość całkowita: 3%

Gęstość 0% oznacza brak powiązań, natomiast „—” oznacza, że liczebność grup była zbyt mała, aby wyznaczyć gęstość

ROZWÓJ SYSTEMÓW

Tam, gdzie zaistniała oczywista potrzeba, pojedyncze programy czy ich zespoły były łączone w większe całości. W ten sposób powstało kilka systemów, które swym znaczeniem znacznie przekroczyły zasięg kompetencji pojedynczych pracowników. Przykładem może być system planowania operatywnego i średniookresowego na potrzeby działów energetyki oraz żeglugi i gospodarki wodnej, warunkujący prawidłową pracę obu wymienionych pionów TVA. Został on uzupełniony i rozbudowany, a co istotniejsze od 1977 r. miał być już na stałe połączony z systemem planowania inwestycji. Tworzenie tego ostatniego systemu, a także wspomnianego połączenia było już inicjowane odgórnie przez specjalnie powołaną komórkę ds. systemów informatycznych na poziomie biura dyrektora zarządzającego.

Trudności hamujące równie szybki wzrost systemów w innych dziedzinach działalności TVA wynikały np. z braku odpowiednich kompetencji administracyjnych (w rolnictwie). Inną przyczyną był — np. w dziale ochrony środowiska naturalnego — brak sprawdzonych metod albo po prostu brak rzeczywistego zapotrzebowania czy też zbyt duże zróżnicowanie sposobów działania, jak to miało miejsce w dziale rozpowszechniania technologii produkcji nawozów sztucznych.

Na początku lat sześćdziesiątych uzyskano pewien sukces w dziedzinie doradztwa dla farmerów (oparty na pakietach programowania liniowego). Na następne zastosowanie informatyki trzeba było w tej dziedzinie jednak czekać ok. 10 lat. Stwierdzono wtedy, że niektórzy farmerzy prowadzą rachunkowość i bilanse materiałowe swoich gospodarstw w innych usługowych ośrodkach komputerowych.

Nakreślono (odgórnie) dość ambitny plan stworzenia systemu w dziedzinie zagospodarowania przestrzennego i ochrony środowiska naturalnego, nie zdołano jednak wyjść poza system dynamicznej kartografii z jednej strony i system doradztwa dla właścicieli parceli leśnych z drugiej. Programy optymalizujące pracę elektrowni węglowych ze względu na zanieczyszczenie atmosfery w różnych warunkach pogodowych okazały się bezużyteczne wobec zmian i zaostreżenia w tej dziedzinie przepisów w 1975 r. Modele chemiczno-biologiczne zbiorników wodnych nie były na tyle dopracowane, by można je użytkować w jakimkolwiek systemie. Najbardziej sprzyjające warunki do rozszerzania systemów informatycznych zaistniały w dziedzinie produkcji i dystrybucji nawozów sztucznych. Istniały tam zarówno programy związane z technologią i planowaniem produkcji, jak i te sprzedają. Pracownicy użytkujący poszczególne programy nie byli jednak zainteresowani połączeniami, które wymagałyby od nich wiele wysiłku, a przyniosłyby każdemu z nich niewielkie efekty.

NOWE PODEJŚCIE

Dopiero w końcu roku 1975, a więc prawie dwadzieścia lat po wprowadzeniu sprzętu informatycznego można było zaobserwować pewną zmianę w stosunku kierownictwa TVA do działalności informatycznej. Programy i pakiety przestały być już uważane za przedłużenie suwaka logarytmicznego, nomogramów i liczydeł, czyli za służbową, lecz indywiduallyną sprawę poszczególnych pracowników. Uznano, że ponoszone nakłady i ewentualne efekty mają istotne znaczenie dla całej organizacji, a zatem działalność informatyczna wymaga powołania osobnej komórki.

Decyzja ta była bardzo na czasie. Zastosowania informatyki były co prawda nadal związane głównie z działalnością pracowników niższych szczebli, opracowano jednak coraz więcej zastosowań, przede wszystkim modeli, przeznaczonych dla wyższych szczebli zarządzania. Integracja tych zastosowań rzutowałaby na decyzje warunkujące przyszły rozwój TVA, np. pełną realizację połączonych trzech systemów: planowania inwestycji energetycznych (1) z planowaniem operatywnym (2) na podstawie prognozowania rozwoju regionu rzeki Tennessee (3).

Na jesieni 1975 r. w całym TVA pracowało ok. 360 osób z przygotowaniem informatycznym; ośrodek obliczeniowy zatrudniał 96, zaś dział energetyczny — 100. Natomiast komórka głównego specjalisty zatrudniała jednego pracownika, czyli głównego specjalistę. Proporcja ta wiązała się z przyjęciem zasady minimalnej ingerencji. Główny specjalista miał przede wszystkim nadzorować rozwój zastosowań informatyki i zapobiegać dublowaniu wysiłków; kiedy zaś tworzono już zarysy systemów lub wystąpiła ewidentna potrzeba ich stworzenia — kierować odpowiednie fundusze i inne środki dla ułatwienia realizacji. Przede wszystkim — nie szkodzić, czego nauczono się po kilku nieudanych próbach ingerencji. Założono, że użytkownik sam wie najlepiej, co mu jest potrzebne.

Tabela 3. Zaangażowanie w poszczególnych komórkach organizacyjnych TVA (stan z 1975 r.)

- 1) pracownicy z przygotowaniem informatycznym (w stosunku do wszystkich pracowników umysłowych)
2) z przygotowaniem informatycznym (w stosunku do ogółu pracowników)

Komórka organizacyjna	1	2
Biuro dyrektora zarządzającego	2/104	2/104
Finanse, działy administracyjne	67/959	67/959
Działy turystyczne i zarządzania nieruchomościami parkowymi	0/563	0/767
Ośrodek obliczeniowy	96/112	96/119
Zaopatrzenie	2/592	2/733
Biuro projektów i realizacja inwestycji	40,3/27 40	40,3/9946
Ochrona środowiska	23,5/268	23,5/268
Zarząd lasów i rybołówstwa	11,5/128	11,5/128
Dział energetyczny	99,8/2963	99,8/9526
Gospodarka wodna	19,6/517	19,6/554
Razem	861,7/8916	361,7/23104

ICL 7700

W kwietniu br. firma ICL wprowadziła na rynek nowy system przetwarzania informacji tekstowej. Umożliwia on zarówno przetwarzanie danych, jak i typową obróbkę tekstów.

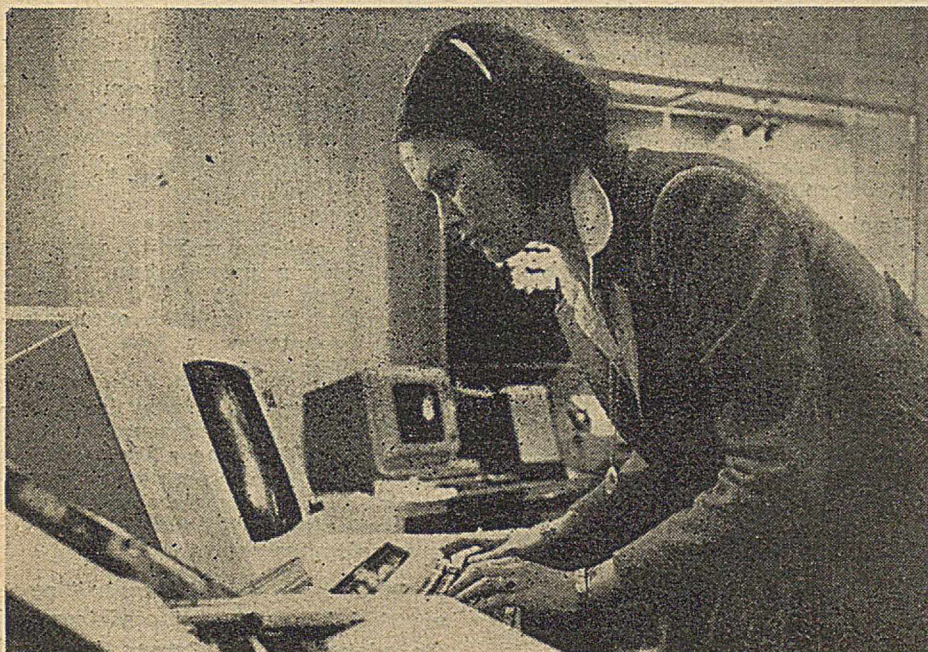
System 7700 może działać samodzielnie lub w powiązaniu z komputerami serii ICL 1900, 2900 DME i serii 2903/4. Przy pracy automatycznej procesor 7700 wczytuje, redaguje, zapamiętuje i drukuje tekst według standardów stosowanych w operacjach obróbki tekstów. Oprogramowanie umożliwia dołączanie lub skreślanie całych fragmentów tekstów, co zapewnia łatwą korektę tekstów przed wydrukowaniem. Tekst i informacje przechowuje się na dyskach elastycznych, z których każdy ma pojemność do 100 stron formatu A4.

Aktualizacja przechowywanej wersji dokumentu przebiega automatycznie. Można również tworzyć nowy dokument wykorzystując tekst z poprzedniej jego wersji. System 7700 może współpracować z urządzeniami do fotokładu, jak również może komunikować się z innymi systemami 7700.

W powiązaniu z komputerami głównymi i przy użyciu pakietu programowego Wordskil Manager system 7700 może transmitować dokumenty do komputera głównego oraz zapewniać dostęp do dużych zbiorów informacji.

Oprogramowanie systemu 7700 umożliwia automatyczne rozsyłanie dokumentu za pośrednictwem komputera głównego bez interwencji operatora.

(T.J.)



System 7700 składa się z dwóch monitorów ekranowych z klawiaturami, dwóch dysków elastycznych i drukarki gwarantującej wysoką jakość pisania korespondencji

Zakup komputera IBM 370/165 w roku 1973 mógł być dokonany dzięki statusowi TVA jako organizacji federalnej. Na początku 1975 r. nie uskarżano się już na zbyt wielkie luzy w obciążeniu istniejącej konfiguracji sprzętu, a zwłaszcza na nadmiar pojemności pamięci. Zastosowania informatyki były w niektórych przypadkach (operatywny rozdział obciążeń energetycznych) konieczne, a także opłacalne.

Stan ten osiągnięto bez dodatkowych wydatków, seminariów i szkoleń. Możliwe, że poniesiono pewne straty nie wprowadzając wcześniej systemów informatycznych obejmujących większe obszary działalności TVA. Wprowadzenie ich wymagałoby istotnych zmian w metodach zarządzania i strukturze organizacyjnej, z zasady dość kosztownych.

Jan W. OWSIŃSKI

IKD 1980

Serię międzynarodowych imprez informatycznych, w które szczególnie obfituje rok przyszedł, uzupełniła ostatnio zapowiedź kongresu IKD 1980 (Internationaler Kongress für Datenverarbeitung), który odbędzie się w dniach 7-10 października w Berlinie Zachodnim.

Kongresy berlińskie organizowane są od 1974 r. co dwa lata. Czwarty kongres — IKD 1980 — będzie się odbywał pod hasłem „Opanowanie technologii informacji wyzwaniem lat osiemdziesiątych”. Hasłu temu podporządkowano pięć głównych tematów obrad:

- Tendencje przetwarzania danych a technologia informacji
- Rozwój i problemy nowoczesnego oprogramowania
- Przewidywane systemy zastosowań
- Komputer a świat pracy
- Zadania szkolnictwa w erze komputerowej.

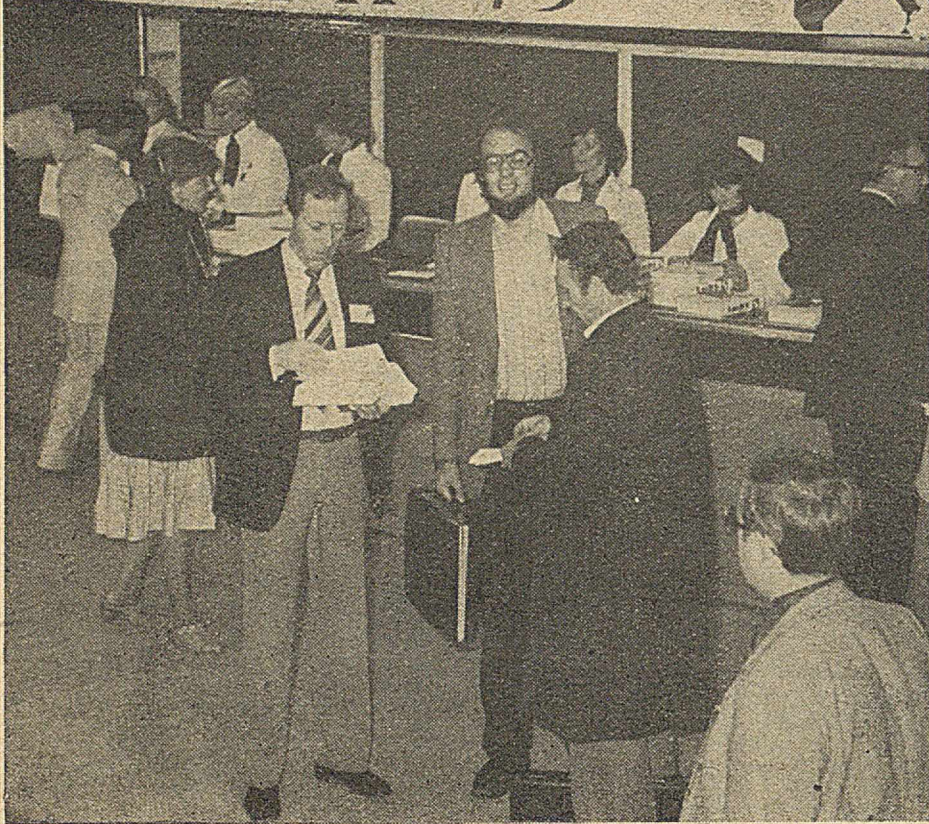
Organizatorzy podkreślają roboczy charakter konferencji, ukierunkowanej głównie na rozwiązywanie problemów praktyki informatycznej oraz intensywną wymianę poglądów.

Kongres połączony będzie z giełdą oprogramowania, która coraz częściej na podobnych imprezach zastępuje pokazy sprzętu. W giełdzie będą uczestniczyć nie tylko producenci oprogramowania i producenci sprzętu, ale także instytuty zastosowań oraz użytkownicy dysponujący uniwersalnym oprogramowaniem.

Termin nadsyłania konspektów referatów — objętość: do ok. 500 słów — upływa z dniem 15 stycznia 1980 r. O przyjęciu referatu organizatorzy IKD zawiadomią w połowie marca. Następnie — do 15 lipca 1980 r. — skróć referat (objętość: do 4 stron maszynopisu) należy przesłać do opublikowania w materiałach konferencyjnych. Wyznaczono też czas, w jakim trzeba zmieścić się wygłaszając referat: 20 minut. Językami oficjalnymi kongresu będą niemiecki, angielski i francuski.

Bliższych informacji udzielają bezpośrednio organizatorzy: AMK Ausstellungs-Messe-Kongress-GmbH, Abteilung II/3, Messedamm 22, D-1000 Berlin 19. (WK)

Euro IFIP 79



Kongresy IFIP odbywają się co trzy lata i są najważniejszym międzynarodowym forum w dziedzinie informatyki na świecie. Ze względu na zapowiadającą się 9-letnią przerwę między kolejnymi kongresami w Europie (Sztokholm — 1974, Paryż — 1983) zorganizowano w br. (25—28 września) europejski kongres IFIP w Londynie. Przewodniczącym Komitetu Programowego EuroIFIP 79 został prof. S. A. Overgaard z Norwegii, a przewodniczącym Komitetu Organizacyjnego — G. A. Fisher z Brytyjskiego Towarzystwa Komputerowego (ang. *British Computer Society*). Techniczną stronę organizacji kongresu powierzono przedsiębiorstwu ON-LINE Conferencés Ltd. (obok zamieszczamy kalendarz imprez, jakie organizuje ON-LINE w 1980 r.).

Na kongres zgłoszono ok. 200 referatów, z których 74 przyjęto do wygłoszenia. Ponadto wygłoszono 26 referatów zamówionych przez organizatorów oraz kilkanaście komunikatów i sprawozdań dotyczących działalności komisji technicznych (*Technical Committee* — TC) IFIP.

W kongresie wzięło udział ok. 1000 osób z 45 krajów wszystkich kontynentów. Najliczniej reprezentowana była ojczyzna gospodarzy (ponad 400 osób), następnie Norwegia, Holandia, RFN, Dania, Francja, Finlandia, Włochy. Wśród uczestników znaleźli się również przedstawiciele wszystkich europejskich krajów socjalistycznych.

Otwarcia kongresu dokonał lord Trenchard, sekretarz stanu w brytyjskim Ministerstwie Przemysłu. Pierw-

sze przemówienie wygłosił prof. P. A. Bobilier, przewodniczący IFIP.

Wśród referatów wygłoszonych na sesji plenarnej szczególne miejsce zajęło wystąpienie przedstawiciela kombinatu ROBOTRON, G. Merkla, Zgodnie z tytułem — „Projektowanie i produkcja komputerów w warunkach krajów socjalistycznych” — referat zawierał przegląd działań podjętych przez kraje członkowskie RWPG, a związanych z realizacją bezprecedensowego wspólnego zamierzenia, jakim jest Jednolity System Elektronicznych Maszyn Cyfrowych.

Referując założenia organizacyjne przedsięwzięcia, autor dużo miejsca poświęcił organizacji przemysłu komputerowego NRD.

Kombinat ROBOTRON wyrósł na producenta znaczącego w skali europejskiej. Obroty handlowe kombinatu wyniosły w 1978 r. ponad 5 mld marek NRD. Ponad 60% wyrobów eksportuje się do 60 krajów, przy czym blisko połowę całej produkcji kombinatu stanowi sprzęt komputerowy.

Omawiając dotychczasowe osiągnięcia i perspektywy dalszej współpracy krajów socjalistycznych, G. Merkel zwrócił uwagę na dwa nowe elementy dotyczące kooperacji: 1° wymianę oprogramowania na zasadzie umów między partnerami oraz 2° koordynację prac podejmowanych przez akademie nauk krajów socjalistycznych, a także podkreślił, że w najbliższych latach przewiduje się rozszerzenie zakresu działania krajowych organizacji serwisowych i rozwój międzynarodo-

wej kooperacji w dziedzinie produkcji sprzętu.

Autor referatu uważa, że organizacja wspólnego przedsięwzięcia stwarza doskonale warunki do współzawodnictwa między producentami komputerów, a współzawodnictwo takie sprzyja rozwojowi techniki, postępowi.

Dyrektor naczelny firmy AMDAHL, Jack Lewis, mówił na temat rozwoju wielkich systemów komputerowych. Jego zdaniem nie należy się spodziewać szybko dużych odstępstw od typowych obecnie rozwiązań, należy natomiast liczyć się z tendencjami do stosowania pamięci operacyjnych o coraz większych pojemnościach.

Omawiając tzw. systemy rozproszone J. Lewis wyraził się bardzo krytycznie o stworzonej wokół tej sprawy reklamie w prasie fachowej i na konferencjach, stwierdzając, że jego zdaniem duże systemy rozproszone (oparte na minikomputerach) będą miały mniej zalet niż tradycyjne rozwiązania z dużą maszyną centralną. Zaletą nowoczesnych minikomputerów powinna być wielka prostota ich wykorzystywania, podczas gdy budowa złożonych wielominikomputerowych systemów (sieci) spowoduje nadmierną komplikację systemów operacyjnych.

Wypowiedź dyrektora naczelnego firmy ICL, dr. C. Wilsona, można w skrócie przedstawić następująco:

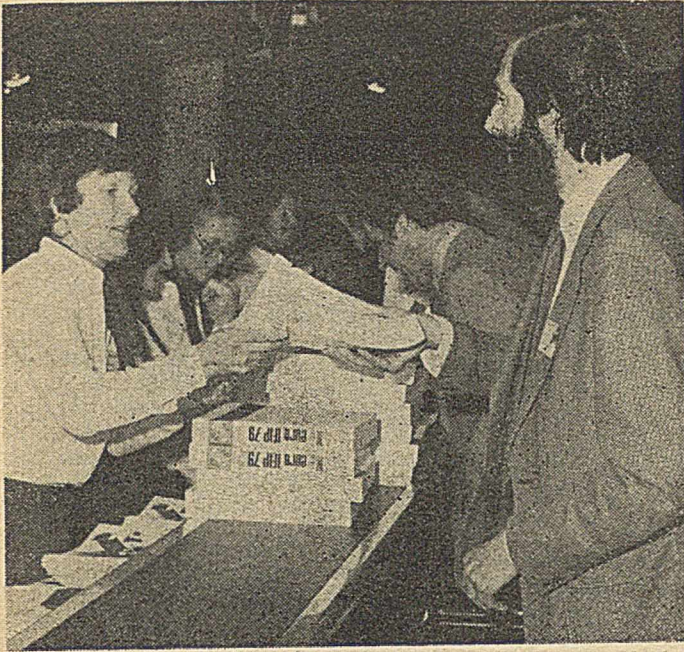
Dostawcy i producenci kompletnych (w domyśle — dużych) systemów komputerowych utrzymują się na rynku w latach 80-tych tylko pod warunkiem, że systemy te będą łatwiejsze w eksploatacji niż dotychczas. Jest to do osiągnięcia poprzez ścisłą współpracę producentów i użytkowników komputerów.

Dr Wilson również sceptycznie zapatrywał się na stosowanie minikomputerów (pod wpływem aktualnej mody na przetwarzanie zdecentralizowane) w niektórych zastosowaniach. Podobnie jak J. Lewis poruszył on sprawę nadmiernej komplikacji systemów operacyjnych nadzorujących sieci minikomputerowe (co jest zaprzeczeniem sensu stosowania minikomputerów z natury rzeczy pomyślanych jako prostsze do stosowania niż duże komputery) oraz problemy związane z utrzymaniem rozproszonych baz danych i ich aktualizacją.

Równie ciekawe referaty wygłosili: B. L. Meek z uniwersytetu w Londynie — na temat roli i znaczenia standardów dla języków oprogramowania komputerów

R. Lorkin z firmy IBM — o systemie informacyjnym dla kierownictwa dużego ośrodka obliczeniowego
V. Lane i J. Corcoran z Wielkiej Brytanii — na temat problemów zarządzania dużymi kompleksami obliczeniowymi wykorzystującymi tzw. przetwarzanie rozproszone

A. Butrimenko z IIASA omówił sprawę sieci komputerowej organizowanej dla ułatwienia współpracy instytucji naukowo-badawczych krajów współpracujących ze sobą w ramach IIASA



Recepcja — goście tłumnie przybywają na obrady



Od prawej: lord Trenchard, G. A. Fisher i W. R. Atkinson

A. Endres z firmy IBM poruszył kwestie oceny programów i metod programowania.

A. d'Agapeyeff z firmy CAP omówił problemy, jakie stoją przed producentami programów w związku z masową produkcją minikomputerów.

I. L. Auerbach w swoim referacie zatytułowanym „A Future Strategy and Structure for IFIP” poruszył sprawę podstawowych jego zdaniem metod działania IFIP jako najważniejszej organizacji międzynarodowej, głównego forum do wymiany opinii na podstawowe tematy z dziedziny informatyki.

Zwrócił też uwagę na zakres działania komisji technicznych, które powinny koncentrować się na zagadnieniach leżących u podstaw rozwoju tej dziedziny, a nie na zastosowaniach informatyki.

Rozwój zastosowań, natomiast, powinien być przedmiotem zainteresowania tzw. grup specjalnych zainteresowań (ang. *Special Interest Group* — SIG). Ponadto dla ułatwienia międzynarodowej współpracy, poza głównymi kongresami IFIP, należy organizować kongresy regionalne, gdyż koszty udziału w kongresach przekraczają możliwości niektórych organizacji członkowskich. Sprawę kosztów poruszył też w dyskusji nad referatem prof. Dorodnicyn, członek delegacji ZSRR na kongres.

Ostatni referat wygłosił prof. E. W. w Tokio i Melbourne. St. Jaskólski

Na zakończenie głos zabrał ponownie prof. P. A. Bobilier, zapraszając zebranych na ósmy kongres IFIP, który odbędzie się w październiku 1980 r. w Tokio i Melbourne. St. JASKÓLSKI

Kalendarz imprez ONLINE w 1980 r.

Znane na terenie międzynarodowym brytyjskie przedsiębiorstwo organizacji konferencji i wystaw informatycznych (ONLINE*) ogłosiło ostatnio kalendarz planowanych na 1980 r. imprez. Ich tematyka odzwierciedla aktualne zainteresowania i problemy użytkowników, wynikające z istniejących i przewidywanych trendów rozwojowych sprzętu, oprogramowania i zastosowań. Może to być interesujące nie tylko dla ewentualnych polskich uczestników tych imprez, ale może też stanowić inspirację dla organizatorów podobnych imprez krajowych.

Konferencja nt. zdalnego odczytu (Remote Sensing)
26—28 lutego, Londyn, International Press Centre

Konferencja i wystawa nt. wizualnego przetwarzania danych (Viewdata'80)
26—28 marca, Londyn, Wembley Conference Centre

Konferencja nt. sprzedaży detalicznej w latach osiemdziesiątych i związanych z nią problemów zarządzania zapasami (Retailing in the 80's — Inventory Management For Profit)
16—17 kwietnia, Londyn, International Press Centre

Wystawa i seminaria nt. mikrokomputerów (Mersey Micro Show)
30 kwietnia — 2 maja, Liverpool, Adelphi Hotel

Konferencja w powiązaniu z wystawą nt. przetwarzania tekstów (Word Processing)
20—23 maja, Londyn, Wembley Conference Centre

Konferencja nt. sieci komputerowych (Networks'80)
3—5 czerwca, Londyn, Bloomsbury Centre Hotel

Konferencja nt. zintegrowanej łączności na potrzeby przedsiębiorstw (Integrated Business Telecommunications)
22—24 lipca, Londyn, Wembley Conference Centre

Konferencja na temat rozproszonego przetwarzania (Distributed Processing)
10—11 lipca, Londyn, Royal Lancaster Hotel

Wystawa i seminaria nt. mikrokomputerów (1980 Microcomputer Show)
22—24 lipca, London, Wembley Conference Centre

Konferencja nt. grafiki komputerowej (Computer Graphics '80)
12—14 sierpnia, Birmingham, Metropole Hotel

Konferencja nt. systemów dla małych przedsiębiorstw (Small Company Systems)
16—17 września, Londyn, International Press Centre

Konferencja nt. optoelektroniki (Optical Electronics)
7—9 października, Londyn, Royal Lancaster Hotel

Konferencja nt. przetwarzania obrazów (Image Processing)
12—13 listopada, Londyn, International Press Centre

Bliższe szczegóły na temat powyższych imprez można otrzymać pod adresem: **ONLINE CONFERENCES LTD, Cleveland Road, Uxbridge UB8 2DD (W. Brytania).**

* Charakterystyka tego przedsiębiorstwa zawarta jest w relacji z konferencji EURO-COMP 76 (INFORMATYKA nr 12/76, s. 39)

Współpraca polsko – rumuńska

Obydwie strony z zadowoleniem stwierdzają, że ich współpraca — zwłaszcza w trzech ostatnich latach — rozwija się pomyślnie. Odnotowuje się już praktyczne wykorzystywanie elementów koncepcji i rozwiązań partnerów w pracach własnych.

Powyższy akapit mógłby być fragmentem protokołu z pobytu delegacji rumuńskiej w Polsce (lub odwrotnie), gdyby nie fakt, iż protokoły rzeczywiste są z roku na rok coraz bardziej wypełnione konkretnymi merytorycznymi. Mimo że różnice w wyposażeniu komputerowym uniemożliwiają bezpośrednią wymianę pakietów oprogramowania i mimo że organizacja obsługi informatycznej budownictwa w obydwu krajach jest zasadniczo odmienna.

Z organizacją usług informatycznych dla budownictwa w Polsce czytelnicy INFORMATYKI mogli się zapoznać już wcześniej — choćby w numerze 5/79. Pragniemy przeto scharakteryzować pokrótce stan tej dziedziny zastosowań informatyki w Socjalistycznej Republice Rumunii.

Istnieje w Rumunii Ministerstwo Budownictwa Przemysłowego, skupiające ok. 1/3 potencjału budowlanego kraju. Istnieje oddzielne Ministerstwo Materiałów Budowlanych. A to natomiast, co w Polsce nazywa się budownictwem ogólnym (mieszkania, szkoły, szpitale, sklepy itp.) jest zdecentralizowane i podporządkowane władzom wojewódzkim (40 województw, zwanych *judet* — czyt. żudec). Ponadto, poszczególne resorty gospodarcze dysponują własnymi organizacjami budowlanymi typu polskich zjednoczeń, głównie o charakterze specjalistycznym, przy czym większość typowych prac budowlanych wykonują dla nich resort budownictwa przemysłowego.

Ministerstwo Budownictwa Przemysłowego ma pod swoim zarządem 12 zjednoczeń (zwanych „trustami”) o regionalnym zasięgu działania, 6 zjednoczeń specjalistycznych działających w skali kraju oraz centralne zarządy odpowiedzialne przed resortem za transport budowlany, gospodarkę maszynami ciężkimi, zaopatrzenie w materiały itp.

Organizacja całokształtu usług informatycznych jest w Rumunii w zasadzie zdecentralizowana. Tei podstawę stanowią przede wszystkim autonomiczne „wojewódzkie”, ogólnodostępne ośrodki obliczeniowe. Istnieją ponadto ośrodki specjalistyczne, będące własnością ministerstw, niektórych jednostek centralnych i dużych organizacji gospodarczych, zwłaszcza przemysłowych.

Wynoszenie ośrodków informatycznych odznacza się wysokim stopniem jednolitości sortetu (zdecydowanie dominują komputery serii FELIX C, wg licencji francuskiej CII). Komputery o

innej proveniencji (IBM, UNIVAC, ICL, CDC) występują na ogół w ośrodkach na tyle specjalistycznych, że nie mają istotnego wpływu na utrudnianie wymiennosci oprogramowania użytkowego w skali kraju.

Centrum Organizacji i Cybernetyki w Budownictwie (*Centrul de Organizare oi Cibernetica in Constructii*), zwane w skrócie COCC, działa w resorcie budownictwa przemysłowego. Jego rola w znacznym jednak stopniu wykracza poza powinności świadczenia usług temu resortowi. COCC jest w istocie placówką wywierającą główny wpływ na kształtowanie stanu i rozwoju zastosowań informatyki w całym rumuńskim budownictwie.

Na działalność COCC składają się trzy główne kierunki:

1. Projektowanie, wdrażanie i rozpowszechnianie użytkowych systemów informatycznych dla wszystkich rumuńskich organizacji budowlanych oraz dla urzędów centralnych, w zakresie dotyczącym budownictwa. Główny nurt prac jest podporządkowany koncepcji zintegrowanego systemu zarządzania produkcją budowlano-montażową o nazwie SICOP. Sukcesywnie opracowuje się kolejne podsystemy, których dokumentacja jest następnie dostarczana terenowym wojewódzkim ośrodkom przetwarzania danych. Zasadą jest obowiązkowe stosowanie tych podsystemów przez przedsiębiorstwa i zjednoczenia budownictwa.

2. Bezpośrednie świadczenie usług informatycznych dla własnego ministerstwa, dla urzędów centralnych (m.in. KC RPK, Komisja Planowania) w zakresie budownictwa oraz dla zjednoczeń i przedsiębiorstw budowlanych w rejonie Bukaresztu. Zadanie to spełnia ośrodek obliczeniowy w COCC, wyposażony w cztery komputery: FELIX C-256 (2 szt.), FELIX C-512 oraz FELIX C-1024. Zbieranie danych do przetwarzania na użytek ministerstwa i urzędów centralnych odbywa się za pośrednictwem terenowych „minifili” COCC, o których będzie dalej mowa.

3. Szkolenie kadr kierowniczych całego rumuńskiego budownictwa, w zakresie obejmującym projektowanie, technologię i organizację produkcji budowlanej. Szkolenia są obowiązkowe, a każdy ich uczestnik musi się przeszkolić ponownie po pięciu latach (jeśli piastuje jeszcze funkcję kierowniczą). Systematycznie zwiększa się udział tematyki informatycznej. Dla COCC stanowi to istotne ułatwienie na rzecz skutecznego propagowania podsystemów użytkowych. Prowadzi się też szkolenie bardziej na informatykę ukierunkowane, dla kadry bezpośrednio z informatyką współpracującej. Świadectwa ze szkolenia w COCC są brane pod uwagę przy awansach w pracy.

Działalność terenową prowadzi COCC w oparciu o wspomniane „minifilie”, czyli tzw. biura obliczeniowe, organizacyjnie ułożone w przedsiębiorstwach budowlanych, jednak według pragmatyki zawodowej i placowej COCC. Wykonują zadania ustalone przez COCC w porozumieniu z poszczególnymi przedsiębiorstwami. Zadania te polegają przede wszystkim na wdrażaniu i nadzorowaniu eksploatacji systemów informatycznych, z przygotowaniem i weryfikacją nośników danych włącznie. Ponadto biura te prowadzą dla swych przedsiębiorstw obliczenia na elektronicznych urządzeniach średniej techniki obliczeniowej (FELIX 15/30), a dla COCC przygotowują dane potrzebne do przetwarzania dla urzędów centralnych. Przejętne zatrudnienie w filii wynosi 25 osób: 10 specjalistów i 15 operatorów do przygotowania nośników danych i do bieżących obliczeń na miejscu.

Dwustronna współpraca pomiędzy Centrum ETOB i COCC istnieje od wielu lat. Istotnemu ożywieniu uległa jednak po uzupełnieniu kontaktów centralnie planowanych, kontaktami opartymi na tzw. wymianie bezdekwizowej, umożliwiającej regularne spotkania specjalistów bezpośrednio pracujących nad systemami informatycznymi. Stało się dobrym zwyczajem wzajemne informowanie na każdym spotkaniu, jakie elementy rozwiązań partnera zastosowano we własnych pracach.

Do roku bieżącego współpraca obejmowała głównie zastosowania informatyki do zarządzania produkcją budowlano-montażową na szczeblu przedsiębiorstwa oraz do zarządzania transportem budownictwa. Ostatnio ustaliliśmy, że tematyka zostanie rozszerzona o obsługę zarządzania produkcją na szczeblu zjednoczeń i resortów, o zagadnienia projektowania i eksploatacji systemów użytkowych opartych na bazach danych oraz o problemy użytkowe świadczenia usług informatycznych z wykorzystaniem teletransmisji, w tym zwłaszcza tworzenia, aktualizacji i dystrybucji zbiorów danych.

Na podstawie dotychczasowych wyników jesteśmy przekonani, że przyjęty program i tryb współpracy między Centrum ETOB i COCC dostarczy obydwu stronom powodów do satysfakcji zawodowej, a i jednostki organizacyjne budownictwa w obydwu krajach też na tym skorzystają. Jesteśmy przekonani do tego stopnia, że zdecydowaliśmy się o tym poinformować także czytelników INFORMATYKI.

Dr inż. Nicolae TOTOS
Naczelny Dyrektor Centrum Organizacji i Cybernetyki w Budownictwie (COCC) Bukareszt
Mgr inż. Wincenty ŁADA
Z-ca Naczelnego Dyrektora Centrum ETOB Warszawa

III samochodowy rajd integracji

Tradycyjnym już, bo trzyletnim, zwyczajem odbył się w październiku br. turystyczny rajd samochodowy, o nazwie jak w tytule. Organizatorem tej dorocznej imprezy jest Koło Polskiego Związku Inżynierów i Techników Budownictwa przy przedsiębiorstwie ETOB w Krakowie. Tym razem Koło zostało wydatnie wsparte (duchowo i materialnie) przez Międzyzakładowy Klub Techniki i Racjonalizacji Przemysłu Budowlanego w Krakowie, co uczestnicy odczuli m.in. w postaci zaliczanego do rajdowej punktacji testu z dziedziny racjonalizacji i wynalazczości w budownictwie.

Słowo *integracja* należy bez wątplenia do klasycznego repertuaru tzw. wytrychów pojęciowych, nie tylko dlatego, że ma o wiele miłsze dla ucha polskie odpowiedniki.

W rajdzie, o którym mowa, pojęcie to znajduje jednak wyraz raczej konwencjonalny („Słownik Wyrazów Obcych”, PWN, 1971), oznaczając *proces zespalania się ... w rozpatrywanej społeczności, wyrażający się częstością kontaktów, ich intensywnością oraz wspólnotą ideową*. Taka jest przynajmniej idea rajdu, o tyle potwierdzana praktyką, że udział w nim etobowców i ich użytkowników kształtuje się, jak dotąd, mniej więcej po połowie.

Po to jednak, aby impreza mogła nazwą rajdu się szczyścić, obowiązuje kontrolowana jazda według itinerera, są próby sprawnościowe, testy okolicznościowe i informatyczno-budowlane oraz inne sprawdziany uczestników, wedle inwencji organizatorów. Jest także zwiedzanie interesujących zakładów przemysłowych.

Rajd pierwszy przebiegał Jurą Krakowsko-Częstochowską, ze zwiedzaniem Huty Szkła Gospodarczego w Żąbkowicach (możliwość interesujących zakupów) oraz Huty Katowice. Rajd drugi to Pińczów, Kielce, Jędrzejów, ze zwiedzaniem fabryki domów. Rajd trzeci wreszcie odbył się w Bieszczadach, z uprzednim zwiedzeniem Krośnieńskich Hut Szkła (zakupy w sklepie patronackim), wejściem na Połoninę Wetlińską (ponieważ przez Hnatowe Berdo, szczyt osiągnęło niewiele ponad 20% uczestników, z czego prawie połowa pomagała sobie rękami), złożeniem kwiatów pod pomnikiem Generała Waltera w Jabłonkach oraz tradycyjnym dla rajdu ogniskiem, tym razem w Polańczyku.

Kronikarsko rzecz traktując, należy odnotować, że w rajdzie uczestniczyło ponad sześćdziesiąt osób — w dwu-

dziestu dwóch na ogół zintegrowanych załogach, Komandorem Rajdu był Henryk Rajchel (tytuły i stanowiska opuszcza się tu całkiem świadomie), przewodniczącym Jury Sportowo-Ogólnego był Józef Orski (użytkownik, zwycięzca poprzedniego rajdu), przewodniczącym Jury d'Appel — niżej podpisany (ETOB), a rajd wygrała kielecka załoga nr 20 — Halina i Jan Starczyńscy (użytkownicy). Największe sukcesy towarzyskie odnosił Ryszard Jaworski (z małżonką), główny organizator dotychczasowych rajdów. Nie można też nie wspomnieć — i to w temperaturze formowania szkła — o gospodarzu na terenie Krośnieńskich Hut Szkła, Władysławie Warzkowskim.

Uchwalono, że rajd czwarty odbędzie się jesienią 1980 r. w Beskidzie Żywieckim i — być może — wykroczy nawet poza polskie granice.

Fotografii z rajdu nie zamieszczamy, ponieważ technika wydawania INFORMATYKI sprofanowałaby przez najstarszych ludzi bodaj nie pamiętaną tak soczysto-kolorową jesień, jaka przy temperaturze ok. 25°C była wręcz zachłannie konsumowana przez uczestników etobowsko-bieszczadzkiej imprezy. (wład)

WCT/184/K/79

ZE ZJEDNOCZENIA INFORMATYKI

Po raz drugi w Jagniątkowie i Jeleniej Górze

Trzy lata temu ZETO Jelenia Góra powzięło szczęśliwą decyzję utworzenia ośrodka, który dzięki bliskości gór jest wymarzony dla czasowiczów, zaś z racji potrzeb szkoleniowych Zjednoczenia Informatyki stanowi świetną bazę dla licznych kursów organizowanych dla informatyków.

Zacząto skromnie — od adaptacji wykupionego budynku. Obecnie także drugi — wzniesiony od podstaw budynek — jest już po ostatecznej kosmetyce. Ośrodek szkoleniowo-wypoczynkowy może aktualnie przyjąć sześćdziesięciu kursantów lub czasowiczów. Można mieć nadzieję, że już w przyszłym roku baza noclegowa powiększy się o dalszych 15 miejsc, ośrodek zostanie wreszcie okolony ogrodzeniem, a kto wie, czy nie powiększony o nowy budynek stołówkowy (co już dziś jest konieczne — bo stołówka jest niewielka) i wyposażony w budowany opodal asfaltowy kort tenisowy. Na pewno zostanie uporząd-

kowany ostatecznie teren, na którym stoją budynki.

Już w tym miejscu należy powiedzieć, że sporo tych dokonań jest wynikiem troski o funkcjonalność obiektu, przejawianej przez jego kierownika, mgr. Mariana Piechockiego. Oddany bez reszty Ośrodkowi, snuje dalsze plany jego rozbudowy, które jeżeli doczekają się realizacji, przeistoczą go w istne cacko.

Zmartwieniem kierownika Ośrodka, a pewnie i dyrektora jeleniogórskiego ZETO, jest słabe wykorzystanie pomieszczeń w okresach niewczasowych. O ile zimą i latem bawi tu komplet czasowiczów, o tyle wiosną i jesienią, zarezerwowaną na kursy szkoleniowe, frekwencja mocno spada. Powinno to dać trochę do myślenia przedsiębiorstwu z sieci Zjednoczenia Informatyki a może i samemu Zjednoczeniu (?)

Okazji do pobieżnego raportu o stanie Ośrodka w Jagniątkowie dostarczył mi (drugi już) pobyt na mistrzostwach tenisowych informatyków. Po-

dobnie jak i poprzednie, rozegrane zostały na kortach w Jeleniej Górze. Podobnie jak rok temu — zdecydowanie wygrali je gospodarze. Podobnie jak rok temu były pełne wydarzenia, o których długo się będzie mówiło w środowisku informatyków.

Organizacyjnie mistrzostwa udały się jeszcze lepiej niż poprzednie. Można mieć nadzieję, że przyszłoroczne będą jeszcze bardziej udane. Namawiałbym tylko Zjednoczenie Informatyki, aby „zafundowało” kolejne mistrzostwa informatykom, także tym spoza swoich zakładów. Skomplikuje to zapewne organizację turnieju, ale zaowocuje pod postacią szerszej integracji informatyków z całej Polski. Wydatek nie taki znow wielki, a korzyść ogromna. Uzasadni to także aspiracje do używanej nazwy turnieju: „Mistrzostwa Polski Informatyków”.

O potrzebie zaś imprez tego typu co jeleniogórskie — nie trzeba się chyba rozwodzić.

K. BERNATOWICZ

ROBOTRON 4201 do zarządzania

W końcu 1977 r. zainstalowany został w Zjednoczeniu „Elektromontaż” pierwszy w Polsce system minikomputerowy ROBOTRON 4201. O wyborze tego systemu na potrzeby usprawniania procesu zarządzania przemysłową produkcją urządzeń elektrycznych zdecydowały wysokie parametry użytkowe systemu, a w szczególności pojemność pamięci operacyjnej oraz możliwość przyłączenia stosunkowo dużych pamięci zewnętrznych, co nie jest osiągalne w innych dostępnych typach minikomputerów.

System ROBOTRON 4201 należy do rodziny maszyn ROBOTRON 4000 oraz 4200 i stanowi ich kolejną wersję rozwojową. Przy odpowiednim doborze urządzeń peryferyjnych minikomputery ROBOTRON 4000, 4200 i 4201 stwarzają możliwość zastosowań do rozwiązywania zagadnień ekonomicznych, obliczeń naukowo-technicznych, sterowania procesami technologicznymi, w procesach symulacyjnych i laboratoryjnych.

OGÓLNA CHARAKTERYSTYKA MINIKOMPUTERA

Minikomputery ROBOTRON należą do rodziny maszyn Jednolitego Systemu. Są one stosowane jako samodzielnie pracujące systemy minikomputerowe lub jako stacje lokalnego przetwarzania o symbolach EC 8404 i EC 8505 w dużych systemach transmisji danych o charakterze satelitarzym z zastosowaniem np. EC 1040 jako komputera centralnego.

System ROBOTRON 4201 w konfiguracji podstawowej może składać się z następujących urządzeń:

- minikomputer ROBOTRON 4201 z pamięcią operacyjną o pojemności od 8 do 32 K słów 16-bitowych
- urządzenia peryferyjne do przetwarzania danych
- urządzenia peryferyjne do zdalnego przetwarzania danych
- urządzenia peryferyjne do sterowania procesami.

System jest wyposażony w oprogramowanie maszynowe MOS oraz problemowe (POS). Może on również rozwiązywać zadania w czasie rzeczywistym.

Możliwości eksploatacyjne charakteryzują następujące parametry:

- słowo 16-bitowe
- ferrytowa pamięć operacyjna o czasie cyklu 1,3 mikrosekundy
- lista rozkazów: 54 rozkazy efektywne
- 20 kanałów przerwań, z tego 16 kanałów zewnętrznych
- 16 adresów przyłączy sterujących w kanale programowym (każde do przyłączenia 1—2 urządzeń zewnętrznych)
- szybkość obliczeniowa ok. 74 tys. operacji/s
- wbudowany zegar czasu rzeczywistego
- sektorowy podział pamięci (po 512 słów)
- szybkość blokowego przesyłania danych do PAO: 0,212 Mb/sek.

Instalację systemu minikomputerowego ROBOTRON 4201 w Zjednoczeniu „Elektromontaż” przewidziano w dwu etapach, a rodzaj i liczbę urządzeń zewnętrznych uzależniono od aktualnie wdrażanego zakresu zastosowań.

W I etapie (1977 r.) zainstalowano:

- jednostkę pamięci operacyjnej o pojemności 16 K słów
- 2 drukarki mozaikowe (daro 1156 i 1160)
- 2 jednostki pamięci bębnowej o pojemności po 98 304 słów
- półautomatyczny system przygotowania danych daro 1415
- elektryczną maszynę do pisania jako pulpit operatorski
- 2 czytniki taśmy dziurkowanej CT 2100
- perforator taśmy papierowej daro 1215
- 2 jednostki pamięci taśmowej MBE 4000 (po 2 przewijaki).

W etapie tym zrezygnowano z urządzeń przygotowania danych na kartach papierowych.

W II etapie rozbudowy konfiguracji systemu ROBOTRON 4201 (1979 r.) zainstalowano 2 dodatkowe jednostki pamięci taśmowej, 4 jednostki pamięci dyskowej ISOT 1370 i drukarkę wierszową EC 7184. Ponadto powiększono pamięć operacyjną do pojemności 32 K słów oraz zakupiono system przygotowania danych na kasetowej taśmie magnetycznej daro 1372 wraz z konwerterem tej taśmy na taśmę półcalową daro 1255.

Powyższa konfiguracja zainstalowana została w Centralnym Ośrodku Badawczo-Rozwojowym „Elektromontaż”, z tym, że pamięci bębnowe i drukarkę daro 1160 przekazano na uzupełnienie konfiguracji drugiego egzemplarza systemu ROBOTRON 4201 zainstalowanego w Poznaniu.



Na pierwszym planie jednostka centralna KRS 4201 32 K. W głębi kontrola pracy drukarki wierszowej EC 7184 oraz przewijaka pamięci taśmowej ISOT 5003

MOŻLIWOŚCI ZASTOSOWAŃ I OPROGRAMOWANIE

W zakresie zagadnień ekonomicznych system może być stosowany do przetwarzania danych w dziedzinie gospodarki materiałowej, krótkookresowego planowania produkcji, optymalizacji pracy transportu. W obszarze obliczeń naukowo-technicznych można wykonywać różnego typu obliczenia techniczne, laboratoryjne, symulacje, obliczenia statystyczne i optymalizacyjne. W zakresie sterowania procesami system ROBOTRON 4201 może zbierać i przetwarzać dane z dowolnych procesów oraz kontrolować i kierować ich przebiegiem, automatyzować procesy laboratoryjne, sterować urządzeniami i maszynami, a także pracować w systemie zdalnego przetwarzania danych.

Jak już wspomniano, ROBOTRON 4201 wyposażony jest w oprogramowanie maszynowe MOS oraz oprogramowanie zorientowane problemowo POS. Oprogramowanie problemowe ukierunkowane jest na konkretne zastosowania, redukując znacznie czas pisania programów użytkowych.

W zależności od typu zastosowania producent dostarcza odpowiednią wersję oprogramowania MOS i POS. MOS obejmuje języki programowania i ich translatory oraz systemy operacyjne.

produkcją urządzeń elektrycznych

Jako maszynowy język programowania stosowany jest język symboliczny SYPS 4200. Oprócz tego języka stosowane są też MAKROSYPS 4200 oraz FORTRAN 4200.

Językami problemowymi są: PEPS, DIWA 4200 oraz FORTRAN IV. Użytkownik w zależności od potrzeb może stosować następujące systemy operacyjne:

- UEAS i SOEK, zorientowane na pracę z pamięcią operacyjną
- SOET, zorientowany na pracę z pamięcią bębnową
- OSPO, zorientowany na pracę z pamięcią dyskową
- ESKO, przeznaczony do pracy w czasie rzeczywistym
- FOBS, przeznaczony dla zastosowań naukowo-technicznych korzystających z programów w języku FORTRAN IV.

R 4201 w Zjednoczeniu „Elektromontaż” korzysta dotąd z systemu operacyjnego SOET, który z chwilą pełnego rozruchu pamięci dyskowych zostanie zastąpiony systemem OSPO.

Oprogramowanie POS zapewnia bardziej efektywne wykorzystywanie systemu R 4201 dzięki możliwości korzystania przez użytkownika z procedur zwanych modułami programowymi (niem. *Programmbausteine*), skracających czas pisania programów o około 40%.

ZASTOSOWANIE W PRODUKCJI URZĄDZEŃ ELEKTRYCZNYCH

Doświadczenia wielu przedsiębiorstw przemysłowych — zarówno w kraju, jak i za granicą — wykazały, że możliwość operatywnego zarządzania produkcją przemysłową przy zastosowaniu techniki komputerowej istnieje jedynie w przypadku instalacji sprzętu w bezpośredniej bliskości procesu produkcyjnego. Zasada ta została przyjęta również w Zjednoczeniu „Elektromontaż”, z tym, że maszyna cyfrowa spełnia tu dwie funkcje, a mianowicie:

- obsługuje przetwarzanie danych dla przedsiębiorstwa pilotującego system informatyczny „Sterowanie produkcją urządzeń elektrycznych” (STER-UE) we Wrocławiu

- obsługuje wybrane moduły tego systemu eksploatowane we wszystkich 14 podległych przedsiębiorstwach na potrzeby Centrali Zjednoczenia.

Aktualnie system STER-UE składa się z następujących jednostek przetwarzania i modułów:

Jednostka przetwarzania (symbol)	Moduł (symbol)
Techniczne przygotowanie produkcji (TP-1) Planowanie produkcji urządzeń elektrycznych (PP-2)	Zakładanie i aktualizacja bazy normatywnej (TP-1.0.01) Emissja zleceń produkcyjnych na wyroby (PP-2.2.01) Planowanie kwartalne (PP-2.2.02)
Sterowanie zapasami materiałowymi (SZM-3)	Zakładanie i aktualizacja stanów magazynowych materiałów i wyrobów gotowych (SZM-3.1.01) Kontrola materiałowego zabezpieczenia produkcji (SZM-3.1.02)
Kontrola realizacji produkcji urządzeń (KRP-4)	Rozliczanie zleceń produkcyjnych (KRP-4.1.01) Rejestr produkcji zakończonej (KRP-4.1.02)
Gospodarka wyrobami gotowymi (GWG-5)	W fazie projektowania

Moduły „Planowanie kwartalne” oraz „Rejestr produkcji zakończonej” funkcjonują zarówno na poziomie przedsiębiorstw, jak i zjednoczenia.

Na potrzeby przedsiębiorstwa pilotującego we Wrocławiu system emituje m.in. następujące tabulogramy wyników:

- rozwinięcia strukturalne wyrobów wg poziomów montażu
- limity potrzeb materiałowych, robocizny i czasu maszyn
- plany produkcyjne wg wydziałów
- plany kwartalne wg poszczególnych wyrobów
- raporty produkcyjne
- portfel zleceń wg odbiorców i kierunków eksportu
- analizy stanów zapasów materiałowych (porównanie w stosunku do normatywów).

W przedsiębiorstwie pilotującym systemem objętych jest łącznie kilkaset wyrobów (rozdzielnice Uniblok i Problok, stacje CTS, rozdzielnice typu ZUR).

Na potrzeby Zjednoczenia maszyna bilansuje zlecenia produkcyjne na około 4200 wyrobów wg grup odbiorców, kierunków eksportu, priorytetu pilności, obiektów szczególnie ważnych dla gospodarki, a także emituje plany kwartalne w ujęciu ilościowym i wartościowym wg grup wyrobów.

Doświadczenia eksploatacyjne wykazały, że ROBOTRON 4201 charakteryzuje się dużą niezawodnością działania (w czasie całego okresu eksploatacji uszkodzeniu uległa jedynie drukarka). Ponadto nie wymaga klimatyzacji i może pracować w normalnych warunkach biurowych. Warunki instalacyjne sprowadzają się jedynie do ustalenia temperatury w pomieszczeniu w granicach 5–40°C (dla pamięci dyskowych ISOT 1370: 5–35°C), dopuszczalnych zmian 5°C/h oraz wilgotności względnej powietrza 80% przy temperaturze 30°C (dla ISOT 1370: 80% przy 25°C).

W październiku br. zainstalowano kolejny system cyfrowy w przedsiębiorstwie „Elektromontaż” w Poznaniu, a w 1980 r. następny zainstaluje się w Krakowie.

Istotnym mankamentem podstawowych konfiguracji KRS 4201 jest dość prymitywne wejście oparte na taśmie i kartach dziurkowanej, które — jak już wspomniano — w Zjednoczeniu „Elektromontaż” zostało zastąpione klawiaturowymi urządzeniami do przygotowania danych na taśmie magnetycznej kasetowej daro 1372 oraz urządzeniem daro 1255 do konwersji tego rodzaju zapisu na taśmę półcokolową do pamięci taśmowej ISOT 5003.

Mgr Lech ŁASICA i mgr inż. Kazimierz ZIELIŃSKI
Zjednoczenie Produkcji i Montażu
Urządzeń Elektrycznych Budownictwa
„Elektromontaż”
Warszawa

ANDRZEJ SIEMIŃSKI

Biblioteka Główna i Ośrodek Informacji Naukowo-Technicznej
Politechniki Wrocławskiej

Kompresja baz danych za pomocą

W zastosowaniach maszyn cyfrowych do celów gospodarczych bądź do wyszukiwania informacji czas przesyłania danych pomiędzy pamięcią operacyjną i pomocniczą ma znaczny wpływ na czas wykonania całego zadania. Wraz ze wzrostem tego typu zastosowań coraz większego znaczenia nabiera kompresja danych. W szczególności dotyczy to systemów wyszukiwania informacji. Wiąże się to z szybkim powiększaniem się wykorzystywanych zbiorów. Stopień starzenia się informacji zmienia się w zależności od dziedziny wiedzy, lecz nawet w najszybciej rozwijających się gałęziach nauki trudno świadomie rezygnować z prac opublikowanych przed więcej jak np. 3 lata. Skalę problemu można łatwo dostrzec analizując tempo rocznego przyrostu oraz wielkości już nagromadzonych baz danych systemu Selektywnej Dystrybucji Informacji działającego na Politechnice Wrocławskiej [5]. Zestawienie danych ilościowych, dotyczących 5 baz danych wybranych z 8 aktualnie eksploatowanych, zamieszczone jest w tabeli 1.

Kompresja danych pozwala na zwiększenie efektywności działania tych systemów poprzez obniżkę kosztów ich eksploatacji wynikłą ze zmniejszenia rozmiarów zbiorów oraz liczby transmisji między pamięcią operacyjną i pomocniczą w trakcie przetwarzania.

W artykule zaprezentowano algorytm dekodowania kodów zmiennej długości (KZD) i związany z nim algorytm wybierania ciągów znaków wykorzystywanych w kompresji danych.

PARAMETRY KOMPRESJI I POMIAR WIELKOŚCI INFORMACJI

Podstawowe parametry charakteryzujące każdy algorytm kompresji to: współczynnik kompresji i czas wymagany na kodowanie i dekodowanie.

Dodatkowymi czynnikami oceny algorytmu są: odporność na przekłamanie oraz stopień wrażliwości na zmianę własności statystycznych kodowanych danych.

Rozpatrywanie ilości informacji w odniesieniu do pojedynczego człowieka, z uwagi na brak możliwości sformalizowanego opisu zasobu jego wiadomości i potrzeb, jest trudne do przedstawienia ilościowego, a ponadto nie daje podstaw do uogólnień. Można je osiągnąć przeprowadzając badania na grupie ludzi i uśredniając uzyskane wyniki. Jedne z pierwszych badań w tym zakresie wykonał Shannon [1] na początku lat pięćdziesiątych. Badania te dotyczyły wartości informacyjnej języka naturalnego. W ich trakcie badana osoba starała się odgadnąć litera po literze (począwszy od początku) treść nieznanego sobie zdania.

Po każdym zapytaniu udzielono jej odpowiedzi tak lub nie. Dla języka angielskiego uzyskana w ten sposób wartość informacyjna litery wyniosła około 1,4 bity.

Zaskakująco niewielką wartość informacyjną 1 litery uzyskana w doświadczeniach Shannona (statystycznie rzecz biorąc każda udzielona odpowiedź niosła najwyżej 1 bit informacji) rokuje nadzieje na osiągnięcie co najmniej trzykrotnego skrócenia długości zapisu języka naturalnego w porównaniu ze standardowym sposobem jego przedstawiania. Jedna litera zapisywana jest w zależności od rodzaju maszyny cyfrowej w jej pamięci jako ciąg 6 lub 8 bitów. Należy jednak wziąć pod uwagę, że tak niska wartość informacyjna litery była wynikiem wykorzystywania przez ludzi własności syntaktycznych oraz semantyki odgadywanych zdań. Analizy takiej nie są w stanie przeprowadzić współczesne komputery.

Jeżeli do celów kompresji chcemy wykorzystać semantykę przetwarzanych danych, to wówczas człowiek musi sam podjąć się zadania wyznaczenia odpowiednich kodów, np. stosowanie liter M i K do celu identyfikacji płci pracownika, czy też cyfrowe kody zdobytego wykształcenia. Analogiczne procesy zachodzą zresztą od dawna na gruncie języka naturalnego, gdzie dążąc do zwiększenia efektywności porozumienia się, wprowadzono skróty literowe jak PKP, PKS itp. W wielu przypadkach kompresję można uzyskać dzięki dokładnej analizie rodzaju zapisywanych danych w poszczególnych polach, czy też stosowanego zestawu znaków. Jeżeli jednak stopień złożoności bazy uniemożliwia podobne proste działanie, to wówczas wykorzystywane przez algorytm kompresji własności bazy mają z konieczności charakter czysto statystyczny. Kompresja możliwa jest dzięki temu, że różne znaki (bądź też ich ciągi) pojawiają się z różnymi częstościami. Sprowadza się to najczęściej do znalezienia w bazie zbioru ciągów znaków występujących równie często, a następnie przydzielenia im oraz wszystkim pojedynczym znakom kodów stałej długości n bitów, gdzie $n = \log_2 N$

N — liczba wybranych ciągów znaków (c) + liczba pojedynczych znaków (s)

n — długość binarnego kodu stałej długości.

Ze względów praktycznych wybiera się zwykle najmniejszy zbiór spełniający powyższy warunek. Znalezienie w dużej bazie takiego zbioru jest trudne i czasochłonne, a ponadto w trakcie kodowania należy rozwiązać problem niejednoznaczności kodowania wynikający z nakładania się na siebie wybranych ciągów znaków, co ilustruje przykład obok.

Tabela 1. Zestawienie danych ilościowych dotyczących baz systemu SDI Politechniki Wrocławskiej

Nazwa bazy danych	Taśmy magnetyczne (szt.)		Dokumenty (tys.)		Liczba znaków* (mln)	
	zasoby	roczny przyrost	zasoby	roczny przyrost	zasoby	roczny przyrost
CAC	173	44	800	200	960	240
INSPEC	93	24	560	140	672	168
PASCAL	42	11	500	125	600	150
INIS	23	12	180	90	216	108
ASSISTENT	8	8	40	40	48	48

*) Podane liczby znaków dotyczą przechowywanych opisów łącznie z analizami

kodów zmiennej długości

Wejściowy ciąg znaków: KOMPRESJA

Ciągi znaków wybrane do kodowania:

KO, KOM, PR, JA, MPRS, SJA i wszystkie pojedyncze znaki

Możliwe sposoby kodowania:

1. K-O-M-P-R-E-S-J-A = 9 kodów
2. KOM-PR-E-SJA = 4 kody
3. KO-MPRES-JA = 3 kody

Jak wykazały badania, zagadnienie to zdaje się mieć w praktyce mniejsze znaczenie, niż przypuszczano [2]. Trudność sprawia jednak samo znalezienie takiego zbioru. Nie można bowiem a priori określić z wystarczającą dokładnością, jaką częstość występowania na bazie dokładnie $2^n - s$ ciągów znaków, ani też na odwrót, tzn. ile ciągów znaków występuje z prawdopodobieństwem równym pewnej liczbie. W rezultacie uzyskane ciągi różnią się częstościami występowania, co wpływa na obniżenie współczynnika kompresji.

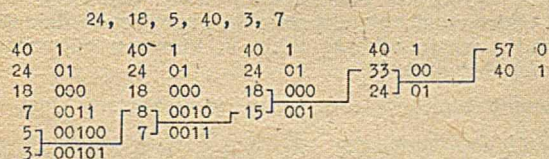
Alternatywne rozwiązanie polega na wyłowieniu z bazy najsilniej związanych ciągów znaków. Ciąg taki charakteryzuje się tym, że warunkowe prawdopodobieństwo występowania w nim poszczególnych znaków osiąga wartość znacznie większą, niż wynosi średnia dla bazy. Jeżeli warunkowe prawdopodobieństwo wystąpienia znaku jest duże, wówczas znak ten niesie ze sobą niewiele informacji i można mu przydzielić krótki kod uzyskując w ten sposób kompresję. Za kryterium wyboru par znaków do połączenia może służyć np.

$P(a/b)$ prawdopodobieństwo wystąpienia znaku a pod warunkiem pojawienia się bezpośrednio przed nim znaku b .

Algorytm wyboru silnie związanych ciągów znaków działa iteracyjnie, na początku analizuje współwystępowania pojedynczych znaków, a następnie za pomocą określonego kryterium (np. $\max P(a/b)$), dokonuje wyboru jednej z par, po czym przydziela jej unikalny identyfikator i traktuje ją jako nowy „pojedynczy” znak. Proces ten kontynuowany jest aż do osiągnięcia zadanej z góry liczby ciągów znaków, bądź też wyczerpania się czasu przeznaczanego na wykrywanie własności statystycznych bazy. Modyfikacja kryterium wyboru par znaków w trakcie działania algorytmu pozwala na uzyskanie zbioru ciągów znaków o własnościach uznanych za korzystne, np. nie mających zbyt wielkiej rozbieżności w częstościach występowania.

Stosowanie za kryterium $\max P(a/b)$ prowadzi do powstania długich, lecz bardzo rzadko występujących ciągów, co ogranicza ich zastosowanie do kompresji. Dlatego celowa jest wstępna selekcja ciągów, polegająca na odrzuceniu ciągów występujących nie dość często. Zastosowanie jako kryterium maksymalnej częstości występowania „pojedynczego” znaku prowadzi do powstania ciągów o stosunkowo niewielkich różnicach w częstościach pojawiania się.

Wejściowe częstości pojawiania się znaków:



Algorytm przydzielania kodów

Przydzielone kody:

- 24-01, 18-0010, 5-00100, 40-1
3-00101, 7-0011

Rys. 1. Algorytm Huffmana tworzenia kodów o minimalnej redundancji

Uzyskany w ten sposób ciąg znaków można już teraz przypisać kody. Problem znalezienia najlepszego zestawu kodów dla takiego zbioru ciągów został rozwiązany przez Huffmana [3]. Jego algorytm tworzenia kodów o minimalnej redundancji jest powszechnie znany i z tego powodu jest zilustrowany jedynie na rysunku 1. Otrzymane w ten sposób kody mają różne długości, co powoduje dużą czasochłonność dekodowania.

DEKODOWANIE KZD

Przy dekodowaniu kodów stałej długości n bitów najczęściej wykorzystuje się tablicę dekodowania (TD) zawierającą 2^n elementów. W każdym z nich zapisany jest ciąg znaków przypisanych odpowiedniemu kodowi. Dekodowanie sprowadza się do pobierania skompresowanego tekstu w ciągach po n bitów. Ich wartość liczbowa wskazuje na odpowiedni element TD. Zastosowanie podobnej metody przy kodach zmiennej długości (KZD) prowadzi do słabego wykorzystania pamięci operacyjnej. Niech A oznacza zbiór kodów zmiennej długości, w_i wartość liczbowa kodu i , zaś d_i jego długość (d_{\max} to maksymalna długość kodu $i \in A$). TD musi wtedy mieć $2^{d_{\max}}$ elementów, z których każdy ma dodatkowo zapisaną liczbę bitów uznanych za nieprzetworzone ($d_{\max} - d_i$). Kodowi i jest przypisanych $2^{d_{\max} - d_i}$ kolejnych elementów TD począwszy od elementu o numerze kolejnym $w_i \cdot 2^{d_{\max} - d_i}$. Jest to pokazane na rysunku 2a. Jak łatwo zauważyć, rozwiązanie takie prowadzi do znacznej nadmiarowości TD. Liczbę redundantnych elementów R_{TD} — redundancja tablicy dekodowania.

$$R_{TD} = \sum_{i \in A} 2^{d_{\max} - d_i}$$

R_{TD} — redundancja tablicy dekodowania.

Niski stopień wykorzystania TD dyskwalifikuje tę metodę.

a/

Kody : 0, 10, 110, 111

Znaki : AB C CDE BAE

Tablica dekodowania:

A B 2	A B 2	A B 2	A B 2	C 1	C 1	C D E 0	B A E 0
---------	---------	---------	---------	-------	-------	-----------	-----------

b/

Tablica dekodowania:

A B *	C *	C D E *	B A E *
---------	-------	-----------	-----------

- * identyfikator węzła końcowego
- identyfikator węzła pośredniego

Rys. 2. a) Najprostsza tablica dekodowania KZD; b) tablica dekodowania o strukturze drzewa

Naturalne podejście do problemu dekodowania KZD polega na skonstruowaniu TD w postaci drzewa, którego struktura odpowiada postaci kodów tak, jak jest to przedstawione na rysunku 2b. Liczbę niezbędnych elementów TD określają poniższe równania:

$$I = \sum_{i=1}^{d \max} B_i; \quad B_i = \frac{B_{i+1}}{2} + N_i; \quad i = 1, 2, \dots, d \max, \\ B_{d \max + 1} = 0$$

gdzie N_i to liczba kodów $\in A$ o długości n bitów
 i to liczba niezbędnych elementów TD.

Niestety dekodowanie przebiega wtedy bit po bicie, co znacznie wydłuża jego trwanie zmniejszając przydatność algorytmu.

Ciekawy sposób dekodowania, pozwalający jednocześnie zarówno na zmniejszenie rozmiarów TD, jak i częściowe jego przyspieszenie, został zaprezentowany w [4]. Kody wykorzystywane w nim muszą mieć własność Fano, tzn. musi istnieć taki sposób uporządkowania ich wg rosnącej wartości liczbowej, by odpowiadające im długości tworzyły ciąg niemalejący. Kody takie porządkuje się w szereg wg malejącej częstości pojawienia się odpowiadających im ciągów. Numer kolejny kodu l określa się mianem jego indeksu (I_l). W kolejnym kroku budowy TD uzyskany szereg dzieli się na grupy, a w ramach każdej z nich musi zachodzić $w_1 - w_{l-1} = 1$, tzn. różnica wartości liczbowych następujących po sobie bezpośrednio kodów równa się 1. Na podstawie tego podziału określa się wartość bazową (WB) i wartość graniczną (WG) dla każdej z grup. Dla grupy Y

$$WG = \max(w_i), \quad WB = w_1 - I_1 \\ I \in Y$$

Przydział WB i WG jest przedstawiony na rysunku 3. W trakcie dekodowania wczytuje się od razu liczbę bitów równą minimalnej długości kodu i porównuje się ich wartość z pierwszą WG. Jeżeli WG jest mniejsza, to dopisuje się nowy bit do ciągu i porównanie przeprowadza z kolejną WG. Operacja ta powtarza się aż do znalezienia $WG \geq$ wartości bitów. Wskazuje ona na WB, która odjęta od wczytanego ciągu bitów daje indeks poszukiwanego kodu.

Kody	00	01	100	101	110	1110	1111
Grupy	I		II		III		
Indeks	0	1	2	3	4	5	6

Grupa	WB	WG
I	0	1
II	2	6
III	9	15

Rys. 3. Określanie wartości bazowych i granicznych

Jak widać, podstawową zaletą ww. algorytmu są niewielkie rozmiary TD. Jeśli chodzi o szybkość tego procesu, to kody o minimalnej długości wymagają jednokrotnego odwołania do TD, zaś kody najdłuższe do tylu odwołań, ile wynosi liczba grup kodów.

W dalszej części artykułu przedstawiony jest algorytm, którego zadaniem jest przyspieszenie dekodowania poprzez zmniejszenie liczby odwołań do TD kosztem zwiększenia jej rozmiarów. Możliwe jest przy tym sterowanie zmianami jednej z tych wielkości poprzez modyfikację drugiej; pozwala to na optymalizację procesu dekodowania.

ALGORYTM SZYBKIEGO DEKODOWANIA KZD

Przy tworzeniu algorytmu przyjęto założenie, że czas dekodowania można znacznie skrócić dzięki zastąpieniu wielokrotnego badania zawartości poszczególnych bitów pamięci poprzez jednokrotne odwołanie się do tablicy dekodowania (TD). Tablica ta zawiera 2^n elementów (n jest liczbą bitów badanych za jednokrotnym odwołaniem się do

TD). Każdy z nich składa się z (i) znaków przypisanych temu elementowi oraz podaje liczbę bitów, które należy uznać za nieprzetworzone. Każdemu elementowi TD przypisana jest całkowita liczba kodów. Oznacza to, że jednokrotne odwołanie się do TD pozwala znaleźć wszystkie kody całkowicie mieszczące się w n bitach.

Wydzielenie ostatnich n nieprzetworzonych bitów jest realizowane za pomocą przesuwania zawartości akumulatorów komputera. W celu przyspieszenia tego procesu zakodowany tekst pobierany jest w formie wygodnych do operowania „porcji” np. słów. Przykładowa TD przedstawiona jest na rysunku 4.

Kody : 0 10 110 111
 Znaki : A C CDE BAE

Tablica dekodowania

A A A 0	A A 1	A C 0	A 2	C A 1	C 1	C D E 0	B D E 0
---------	-------	-------	-----	-------	-----	---------	---------

Rys. 4. Tablica dekodowana wykorzystywana w proponowanym algorytmie

Algorytm realizuje trzy funkcje. Pierwsza z nich obejmuje wyodrębnienie ostatnich n nieprzetworzonych bitów, pobranie odpowiedniego elementu TD i przesłanie zdekodowanego ciągu znaków do bufora wyjściowego. Druga powoływana jest w przypadku konieczności pobrania kolejnej „porcji” bitów do zdekodowania, a trzecia — gdy liczba niezdekodowanych bitów przekracza pojemność „porcji”.

Algorytm został skonstruowany w taki sposób, by skrócić wykonywanie najczęściej powoływanej funkcji pierwszej. Na całkowitą szybkość dekodowania znaczny wpływ ma rozmiar TD oraz stopień jej wykorzystania określany w poniższy sposób:

$$W = \frac{k}{n}$$

n — liczba bitów badanych przy jednym odwołaniu do TD
 k — średnia liczba bitów uznawanych za nieprzetworzone w trakcie jednego odwołania do TD
 W — współczynnik wykorzystania TD.

Szybkość dekodowania można w przybliżeniu obliczyć (oszacować) za pomocą równania:

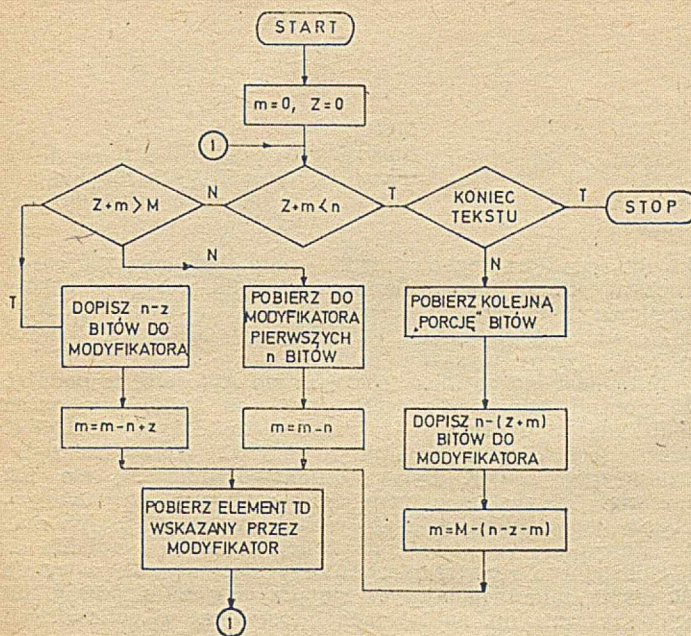
$$T = \frac{N}{n} T_1 + \frac{N}{n^2} (1 - W) \cdot T_2 + \frac{N}{n^2} (1 - W) \cdot T_3 \cdot P_3$$

T — czas dekodowania
 N — liczba bitów skompymowana do tekstu
 T_1 — czas wykonywania funkcji i
 P_3 — prawdopodobieństwo, że po pobraniu nowej porcji bitów tekstu liczba bitów uznanych za nieprzetworzone przekroczy jej rozmiar.

W praktyce okazało się, że funkcja trzecia jest wykorzystywana tak mało razy, że do oszacowania szybkości dekodowania wystarczy uwzględnić dwa pierwsze składniki sumy. Jak widać, zwiększenie n przyspiesza dekodowanie, a ponadto warto wybrać takie n , któremu odpowiada wysoki współczynnik W .

Schemat blokowy algorytmu zamieszczony jest na rysunku 5.

Algorytm w przedstawionej i oprogramowanej wersji działa w przypadku, gdy n jest nie mniejsze od maksymalnej długości kodu. Jeżeli wymagana do tego celu TD jest zbyt duża, to możliwe jest jej rozbitcie na dwie lub więcej części i zapisywanie w nadrzędnej TD zamiast ciągu znaków wskaźników do jednej z pochodnych TD. Jest to przedstawione na rysunku 6. W tym konkretnym przypadku rozbitcie TD pozwoliło na prawie trzykrotne (z 32 na 12) zmniejszenie jej ogólnych rozmiarów.

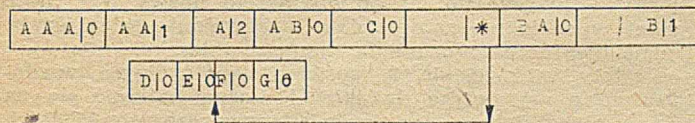


M - liczba bitów w „porcji” skompresowanego tekstu pobieranego do dekodowania
 m - liczba bitów nieodekodowanych w „porcji”
 n - liczba bitów badanych za jednym odwołaniem do TD
 z - liczba bitów nieodekodowanych w ostatnim odwołaniu do TD

Rys. 5. Schemat blokowy algorytmu dekodowania KZD

Algorytm dekodowania kodów zmiennej długości został zaprogramowany w języku PLAN 3. Osiągnięte szybkości realizacji poszczególnych pętli funkcji są przedstawione w tabeli 2. Dla porównania w tabeli tej zamieszczono także szybkości wykonania wybranych instrukcji języka PLAN 3.

Kody - 0, 11, 100, 10100, 10110, 10111, 10101
 Znaki - A B C D E F G



Rys. 6. Rozbicie tablicy dekodowania

Podane wartości odnoszą się do komputera ODRA 1305. Do sumarycznego czasu dekodowania doliczyć należy jeszcze operacje związane z wczytaniem danych i ich wyprowadzeniem.

Tabela 2. Przybliżone czasy trwania funkcji algorytmu i wybranych instrukcji języka PLAN (ODRA 1305)

Rodzaj funkcji lub instrukcji	Czas (μs)
Funkcja 1	80
Funkcja 2	90
Funkcja 3	90
Przesunięcie logiczne zawartości akumulatorów o 5 bitów (SLL)	6,6
Mnożenie dwu liczb (MPY)	15
Dzielenie dwu liczb (DVS)	40
Dodawanie dwu liczb (ADX)	3,6
Odejmowanie dwu liczb (SBX)	4,4
Pobranie do akumulatora liczby (LDX)	2,2
Pobranie do akumulatora znaku (LDCH)	5,4
Operacja pusta (NULL)	1,4

Ostatnio można zaobserwować w systemach informatycznych szczególnie szybki wzrost rozmiarów zbiorów danych. Coraz większą więc wagę przywiązuje się do efektywności ich wykorzystywania. W tej sytuacji obniżka kosztów eksploatacji oferowana przez kompresję staje się coraz atrakcyjniejsza. Nie bez znaczenia jest też fakt zwiększenia stopnia zabezpieczenia zbiorów: odczytanie danych zapisanych KZD przez osobę nie znającą tych kodów jest praktycznie niemożliwe.

LITERATURA

[1] Shannon C. E.: Bell System Technical Journal, 1951, vol. 30, s. 54-57
 [2] Schuegraf E. S., Heaps H. S.: A comparison of algorithms for data compression by use of fragments as language elements. Information Storage and Retrieval, 1974, vol. 10, s. 309-319
 [3] Huffman D. A.: A method for the construction of minimum redundancy codes. Proceedings of IRE, 1952, vol. 40, s. 1098-1101
 [4] Connel J. B.: A Huffman - Shannon - Fano code. Proceedings of IEEE, 1973, vol. 61, s. 1046-1047
 [5] Rzońca I., Szablowski B.: Założenia systemu retrospektywnego wyszukiwania informacji w sieci komputerowej PITAGORAS. Raporty Biblioteki Głównej i OINT Politechniki Wrocławskiej 1979 r., SPRAWOZDANIA nr 2

Czytajcie i prenumerujcie

INFORMATYKĘ

Repetytorium języka PLAN

Motto:

Błogosławieni dobrochętni — albowiem oni posiądą bruk piekielny.
(IX Błogosławieństwo)

Po dziesięciu latach produkowania komputerów rodziny ODRA 1300 i handlowania unikalnymi egzemplarzami dokumentacji na „czarnym rynku informatycznym” ukazał się wreszcie w księgarniach ogólnodostępny opis podstawowego języka programowania tych maszyn¹⁾. W takiej sytuacji WNT-owski wydawca siłą rzeczy ściąga na siebie odium niezadowolonia; i nie pomoże tu ani zwiększenie nakładu — siedmiotysięcznego! — ani usunięcie w następnym wydaniu pewnych usterek. O ile bowiem informatycy krajowi pogodzili się raczej z faktem odstawiania okrągło o dekadę od czołówki światowej, o tyle nie mogą ścierpieć następnej dekady opóźnienia w stosunku do własnej produkcji. Co nie przeszkadza, że podręcznik Zygmunta Drażka i Józefa Jurkowskiego, adresowany nie tylko do zawodowych programistów, praktycznie już nie zdąży dotrzeć do studentów kierunków informatycznych na wyższych uczelniach, ani tym bardziej do studentów szkół policealnych, i wszyscy będą czekali niecierpliwie na trzecie i dalsze wydania, choćby nawet w międzyczasie miała się ziścić groźba wycofania komputerów rodziny ODRA 1300 z produkcji!

Powysze emfazy nie oznaczają bynajmniej, że piszącego te słowa nie ogarnia pewne zakłopotanie. Tak się bowiem złożyło, że podręcznik szcześcińskich autorów ukazał się niemal równocześnie z chlubą WNT, jaką jest dzieło prof. W. Nowickiego „O ścisłość pojęć i kulturę słowa w technice”. A oznacza to ustawienie niezwykle wysokiej poprzeczki.

Pewną pociechą jest znany fakt, że informatycy już przywykli do podręczników wydawanych w znacznie gorszy sposób i z licznymi błędami redakcyjnymi — a na tle dotychczasowych podręczników języka PLAN książka Drażka i Jurkowskiego wypada jakościowo niewątpliwie wyżej. Z pewnością dla wielu odbiorców nowego podręcznika najistotniejsze są tylko kody i przykłady programowe, wszelkie zaś niefortunne werbalizacje mają znaczenie drugorzędne. Przykładowo, jeżeli czytelnik zauważy, że raptem w podręczniku figuruje zdanie w rodzaju *akumulator jest to słowo, w którym przechowuje się wyniki*, odbierze to jako odtwarzalny skrót myślowy. Jest to nieścisłość tak powszechna na kursach programowania, że trudno się dziwić Autorom, iż przepuścili ją w druku. Ostatecznie bowiem na kursach programowania podaje się pewne „kuchenne recepty”, a od uściślonych sformułowań (czasem aż przesadnie) są podręczniki akademickie.

Krótko mówiąc, nie ma złych podręczników, choćby na pozór starannie wydanych, w których na 280 rysunków blisko 200 przedstawia przykłady nieczytelnego wypełnienia formularzy zaznaczonych przesadnie grubą kreską — są tylko źli studenci i kursanci, zdolni wybrzydzać na każdego wydawcę i autora. Przypomina tu się anegdotka o słynnym radzieckim astronomie, twórcy jednej z popularniejszych teorii kosmogonicznych, prof. Fiesienkowie. Otóż uczony ten zwykł na pierwszym wykładzie z góry zapowiadać, że rozmyślnie pozostawia kilkadziesiąt błędów redakcyjnych nie poprawionych, bo wykrycie tych błędów będzie przedmiotem indywidualnych egzaminów końcowych. Dokładną liczbę błędów w skrypcie znali tylko asystenci, którzy wszystkim zgłaszającym poniżej 500 błędów radzili nie pokazywać się profesorowi na oczy.

Nie chcąc być posądzonym o złośliwość, proponowałbym niewiele zmieniać w następnych wydaniach, poza koniecznym wytłumaczeniem wszystkich blisko 200 przykładowych wypełnień formularzy programowych — ale to już raczej

kłopot redaktora technicznego — natomiast dołączyć coś w rodzaju „klucza dla wykładowców”. Tam wszystkie skróty myślowe i różne mniej lub bardziej wyraźne „kiksy”, jakie zdarzyły się nawet Filadelfijskiej Orkiestrze Symfonicznej, byłyby wykazane i omówione.

Natomiast można mieć trochę za złe Autorom, że podając 25 pozycji bibliograficznych, wymienili tylko oryginalne materiały londyńskie ICL, warszawskie skrypty CODiKK oraz wrocławską dokumentację ELWRO. Wprawdzie łódzkie skrypty TNOiK były stosunkowo ubogie²⁾, ale już materiały gdyńskie z pewnością zasługują na uwagę³⁾. Należy jednak podkreślić, że w żadnym z powyższych materiałów nie występowały na końcu rozdziałów zadania do samodzielnego rozwiązania. Autorzy sformułowali w sumie 40 takich zadań, nadających się przy odrobinie odwagi do wykorzystania np. na lekcjach informatyki w policealnych szkołach ekonomicznych, kursach programowania bez możliwości dostępu do komputerów typu ODRA, dłuższych pokazach itp. Ostatecznie przecież każdy wykładowca będzie miał „własną” terminologię i zawsze będzie jakoś próbował wybrzydzać na Autorów. Powysze oznacza niedosyt trudniejszych zadań, gdzie sformułowania werbalne, choćby nawet nieporadne, nie przytłumiają wielkości ukazywanego problemu. Takie w zasadzie są jednak nie do pomyślenia bez pracy „na żywo”⁴⁾.

Przechodząc do konkretów, trzeba ograniczyć się do omówienia kilku przykładowych usterek (pełny wykaz przekazano Autorom do wykorzystania w następnym wydaniu). W krótkim wprowadzeniu (rozdział 1) użyto np. skrótu myślowego w sformułowaniu, że PLAN pozwala w optymalny sposób (podkreślenia recenzenta) wykorzystywać możliwości komputerów rodziny ODRA 1300; ściślej zaś byłoby mówić o optymalnym wykorzystaniu możliwości kodu wewnętrznego. Osobiście też uważam, że gdy w tekście mowa o „1900” to należało użyć określenia ICL, w odnośniku zaś napomknąć o ICT, a nie na odwrót. Są to jednak drobne mankamenty, których przeciętny odbiorca w 10 przeszło lat po obumarciu nazwy ICT może nawet w ogóle nie zauważyć, a wychwalanie zalet języka będzie uważać za zrozumiałą manierę autorską.

Niestety w rozdziale 2, poświęconym ogólnym wiadomościom o arytmetyce dwójkowej można natknąć się na pewne niebezpieczne uproszczenia przy pozorach ścisłości. I tak wzór (1.1) nie odnosi się do wszystkich liczb, a jedynie do tzw. skończenie wymiernych, tj. takich, które przy danej podstawie p mają skończone rozwinięcie cyfrowe. W wyjaśnieniu pod wzorem podano, że wskaźnik i przebiega tylko 0 do n , podczas gdy we wzorze widać, że wskaźnik ten przebiega od m do n (gdzie m jest liczbą całkowitą ujemną). W tymże wyjaśnieniu pomieszano znaki podstawowe systemu pozycyjnego (kilka wierszy wcześniej nazwane cyframi) z kolejnymi cyframi rozwinięcia, jak również tak sformułowano zdanie, że dotyczy ono tylko znaku a_n (pierwszej cyfry rozwinięcia) oraz a_m (ostatniej cyfry rozwinięcia). Oczywiście wszyscy w końcu domyślają się co Autorzy mieli na myśli, sprawia to jednak pozory matematycznego bałaganiarstwa, łatwego zresztą do usunięcia. Mówiąc w dalszej części o akumulatorach,

¹⁾ Niwiński Tadeusz: Zasady programowania w języku PLAN, TNOiK, Łódź 1974, nakład 500 egz., 52 s., cena 15 zł

²⁾ Gołuńska Irena (tłum.): Podręcznik programowania w języku PLAN dla maszyn cyfrowych ICT serii 1900. ZETO-Gdynia 1967, nakład 100 egz.; cena 50 zł. Korekta tłumaczenia (Plan Training Manual ICT 1900 Series, London 1964), wybór rozdziałów i opracowanie redakcyjne: Andrzej Ramuś

³⁾ W bieżącym sezonie szkoleniowym kursy języka PLAN, połączone z ćwiczeniami komputerowymi, prowadzą jeszcze m.in.: ZETO-Łódź, PTE-Gdańsk, OW NOT w Kielcach, Krakowie, Poznaniu i Wrocławiu oraz niektóre oddziały TNOiK

⁴⁾ Drażek Zygmunt i Jurkowski Józef: Język programowania PLAN 3, WNT, seria Informatyka, Warszawa 1978; opiniodawca: Wiesława Pulczyn, red. naukowy: Halina Tempeczyk, red. techn.: Franciszek Bondaruk; 324 stron, 281 rys. + 37 tabl.; bibl. 25 poz. Cena (przy dotacji MNSzWiT) 65 zł; wydanie I, nakład 7000+260 egz., ark. wyd. 21,2. Składano w Pozn. Zakł. Graf. im. M. Kasprzaka

identyfikowanych cyframi od 0 do 7, w tym samym akapicie mówi się, że np. 8 oznacza ósmą komórkę. Adresem (binarnym), komórki nazwano nie kolejny numer komórki w pamięci, ale w ... programie. Przy omawianiu słów indeksowych o 15-bitowej części adresowej podano, że wystarczy to do objęcia całej pamięci operacyjnej, chociaż na s. 65 określono 2^{15} jako pojemność minimalną komputerów ODRA 1300. Nieuważny Czytelnik też może nauczyć się obrzydliwego żargonu, że ułamek nie jest liczbą.

Dosyć wyczerpująco omówiono nośniki informacji (rozdział 3). Ale wiersz 10 nie znajduje się w *niezadrukowanej części karty*. W podpisie pod rysunkiem przedstawiającym perfokod kartowy brak zaznaczenia, że to kod ICL. Osobiście mnie razi określenie *wytłaczanie dziurek* (raczej są one wycinane!), adiustatorowi mogło to jednak nie przeszkadzać. Ale już akapit omawiający symbole zmian poziomu interpretacji kodu perfotaśmowego (tzw. znaki przesunięcia alfa, beta i delta) jest po prostu niezrozumiały dla osoby nie dysponującej oryginalną dokumentacją.

Bardziej merytoryczne błędy zawiera rozdział 4, traktujący o pamięci operacyjnej. Rys. 4.1, 4.2 i 4.3 sugerują, że dane można umieszczać w PAO od słowa 0, gdy w rzeczywistości można to czynić dopiero od słowa 40. Pewne nieporozumienie może wywołać mylenie słów z komórkami; np. podrozdział 4.3 dotyczy nie słów a komórek zarezerwowanych w maszynie, zaś słowa zarezerwowane to takie, których programista nie może używać np. jako etykiet.

Omawiając strukturę programu PLAN-owskiego (rozdział 5) dosyć niejasno określono technikę nakładkową, definiując ją jako wyprowadzanie określonych segmentów na zewnętrzny nośnik. Dla zwykłego czytelnika daleko bardziej zrozumiałe byłoby chyba określenie, że jest to rozczytywanie nośników zewnętrznych do tych samych obszarów PAO, bez szkody dla prawidłowego wykonywania całości programu. Są również zdania noszące ślady przerwania zabiegów adiustacyjnych (*Nazwa etykiety odnosi się do pierwszej wielkości danej albo określonej pozycją w wierszu — s. 77*). Nie wiem dlaczego Autorzy preferują subprogram zamiast podprogram. A już o ból głowy sprawia zdanie, że *dyrektywa główna pojawiająca się w zasięgu działania dyrektywy pomocniczej spowoduje automatyczne ograniczenia obszaru działania dyrektywy pomocniczej tylko do takich zapisów, które są ograniczone zapisem symboli obu dyrektyw*. Dalej, rys. 5.8 ma ilustrować użycie instrukcji ALLOT, ale jej właśnie brakuje na tym rysunku. Trochę niejasno określono dyrektywę #MACRO, którą najłatwiej zrozumieć jako określenie podprogramu otwartego. Przy omawianiu dyrektywy #SWITCH nie wyjaśniono bliżej co to są parametry kompilatora; niejasne jest też określenie *wiersz sterujący kompilacją* (=dyrektywa?). Również #SET można było określić jako nadawanie wartości zmiennej oznaczonej. Osobną sprawą jest posługiwanie się mylącym określeniem *obszar zmiennych*, bo dalej w tekście użyto nieprawidłowej formy dopełniaczowej *obszaru zmiennego*, co by oznaczało mianownik *obszar zmienny*, a to już coś zupełnie innego.

Typowym przykładem niepanowania nad ścisłością jest określenie, że kompilator umieszcza operandy literalne w pamięci dolnej. Bo przecież tam są umieszczone tylko ich adresy, same zaś literały są właśnie umieszczane w pamięci górnej. Zwykłym błędem maszynowym jest określenie, że gwiazdka (*) informuje kompilator o typie operandu, gdy jest to po prostu adres bieżącego rozkazu.

Najwięcej uwag nasuwa się przy czytaniu rozdziału 6, omawiającego instrukcje programowe. Przede wszystkim trzeba zauważyć, że nie omówiono tu wszystkich instrukcji języka PLAN 3. I tak poza szczególnymi symbolami (BSP, BTM, REW, WTM itd.), oznaczającymi pseudo-makrokody różnych przypadków instrukcji PERI, brak do kompletu instrukcji ERN, MVCH, SMO i innych. Odczuwa tu się

brak komentarza, że istnieją dalsze rozszerzenia języka PLAN 3, nie zastosowane w naszych ODRACH ze względu na brak przystawki EDMF, ale możliwe do realizacji na posiadanych w kraju komputerach rodziny ICL 1900. Nie trudno zauważyć skutki dosłownego tłumaczenia angielskiego określenia *set* w odniesieniu do rejestrów jedno-bitowych. Autorzy konsekwentnie „ustawiają” lub „zerują” wskaźniki — osobiście wolałbym mówić o „jedynkowniu”, bo ustawiać można zarówno na 1 jak i na 0. Ale de gustibus...

Niezrozumiały jest opis rozkazu LDXC, gdyż działania na liczbach dwu-długich niedopowiedziane są Interpretacja przykładu 6.10 jest błędna. Nie podkreślono, że operand literalny wyklucza modyfikację. Przykład z przeliczaniem czasu na sekundy (rys. 6.35) zawiera błędne użycie instrukcji MPA (skoro wynik umieszczamy w 4 słowach). Przy instrukcji dzielenia pojawił się specyficzny termin dla ilorazu z nadmiarem i niedomiarem: liczby dolne (!) i górne (!). Przy instrukcjach skokowych nie wyjaśniono co to jest typ „4” i „5”, bo na s. 104 wymieniono tylko cztery typy operandów. Tablica 6.12 zawiera oczywisty błąd drukarski odnośnie do instrukcji TXU. W dodatku instrukcja ta nie jest jeszcze omówiona, a już podaje się związany z nią przykład (rys. 6.39). Przy operacjach logicznych nie zaznaczono, że słowo N(M) pozostaje niezmienione. W przykładzie przebudowy słowa indeksowego (rys. 6.46) występuje jeszcze nie omawiana instrukcja SLL. W tablicy 6.14 nieściśle zdefiniowano rolę wskaźnika V. Wiele instrukcji opisano bardzo powierzchownie, ale z drugiej strony pełny ich opis zawiera dopiero dokumentacja fabryczna, do której mankamentów zdążyli się już wszyscy dawno u nas przyzwyczaić.

Końcowe rozdziały mają już charakter bardziej syntetyczny i „ponad-bitowy”. Mimo „dobrych chęci” recenzentowi nie udało się znaleźć w nich jakichś istotniejszych punktów zaczepienia. W ten sposób książka pozostawia raczej pozytywne wrażenie, jednak w części analitycznej zmusza bezsprzecznie do sięgnięcia po materiały dokumentacyjne.

Być może, że założony przez Autorów układ podręcznika nie jest wygodny przy nauce języka PLAN, ponieważ przy małej objętości niemożliwością jest opisanie wszystkich jego reguł i przypadków praktycznych. Szczególne trudności występują właśnie przy odróżnianiu operacji arytmetycznych jako zwykłych instrukcji i jako makrosów. Sprawie tej, należałoby poświęcić wiele uwagi w niewątpliwie potrzebnym następnym wydaniu. Ale właściwie odważyłbym się postulować generalną zmianę układu podręcznika: język wewnętrzny, PLAN 1, PLAN 2, a dopiero jako trzecią iterację — PLAN 3. I to wszystko przy dołączeniu odpowiednich indeksów oraz odrębnej wkładki z kodem komputerów ODRA.

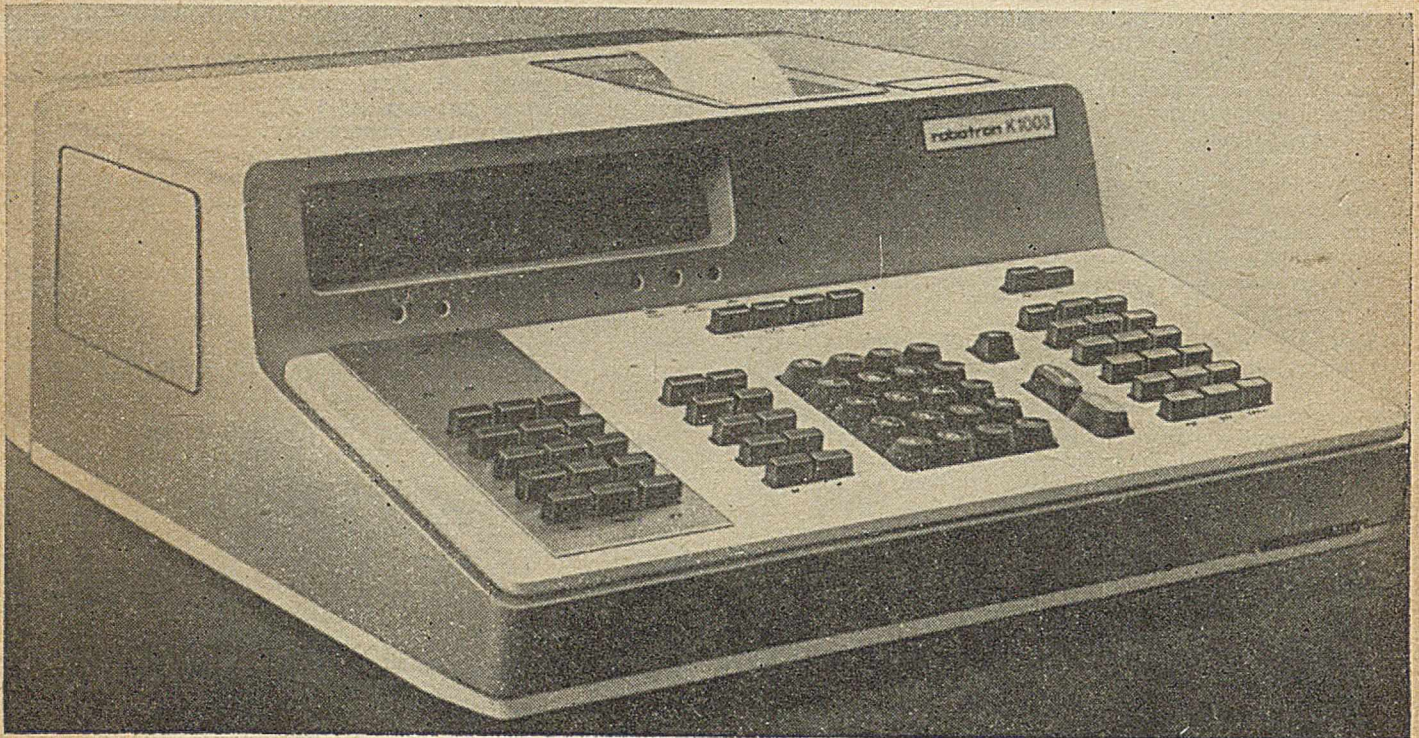
Na marginesie tych uwag szczegółowych nasuwa się pewne generalizujące spostrzeżenie. Otóż wadą wielu podręczników jest dosłowne odpisywanie przykładów z literatury oryginalnej, a nawet z zachowaniem oryginalnych angielskich słów w nazwach zmiennych. Należy podkreślić, że Autorzy z całą świadomością chcieli uniknąć tego wadliwego mankamentu, ale przez to narobili sobie kłopotów natury dydaktycznej, z których nie byli w stanie wybrnąć do końca w pierwszym wydaniu, a które mogą dorównać nawet do pomówienia ich o nieporadność stylistyczną.

Powyższe nie zmienia faktu, że jeżeli kiedykolwiek będę prowadził kurs programowania w języku PLAN, to przede wszystkim — ale nie wyłącznie — będę posługiwać się podręcznikiem Zygmunta Drażka i Józefa Jurkowskiego.

Adam B. EMPACHER

robotron

K 1003



Oto zalety, które można bardzo łatwo wymienić

1. Jest bardzo wydajnym minikomputerem z wieloma wbudowanymi funkcjami
2. Można go zaprogramować bez konieczności nauczenia się języka programowania
3. Zapamiętuje programy na kartach magnetycznych
4. Nawet bardzo specyficzne obliczenia można realizować za pomocą krótkiej sekwencji naciśnięć klawiszy
5. Wyposażony jest w układ automatycznego przełączania z obliczeń stałoprzecinkowych na zmiennoprzecinkowe
6. Do wyprowadzania wyników można w razie potrzeby zastosować drukarkę z zapisem na taśmie

Ambasada NRD
Biuro Rady Handlowego
Wydział Polityki Handlowej
Dział Informatyki
i Maszyn Biurowych
Aleja I Armii
Wojska Polskiego 2-4
01-524 Warszawa

robotron

Robotron Export-Import
Przedsiębiorstwo
Handlu Zagranicznego
Niemieckiej Republiki
Demokratycznej
DDR 108 Berlin,
Friedrichstrasse 61