2. Oprogramowanie

2.1. Wprowadzenie

Charakterystyczną cechą lat siedemdziesiątych w informatyce polskiej, podobnie jak w informatyce światowej, jest docenianie roli oprogramowania jako podstawowego warunku efektywnego stosowania informatyki. Ciągle rosnący udział kosztów oprogramowania (w stosunku do kosztu sprzętu) doprowadził w końcu lat sześćdziesiątych do zjawiska, które w krajach zachodnich określano mianem „kryzysu software'owego". W Polsce nie mieliśmy do czynienia z wyraźnymi objawami tego kryzysu, co należy zawdzięczać zastosowaniu właściwych środków organizacyjno-techniczno-ekonomicznych, różnych na każdym z etapów rozwoju informatyki. Oto one:

1. W okresie pionierskim rozwoju informatyki w Polsce (lata 1948—1962), w którym maszyny cyfrowe opracowywano w środowiskach naukowych, bezpośrednio je użytkujących, zagadnienia programowania nie były, podobnie jak na całym świecie, uważane za problem pierwszoplanowy. Niemniej już w 1959 r. powstał w Zakładzie Aparatów Matematycznych pierwszy w kraju usługowy ośrodek obliczeniowy — Biuro Obliczeń i Programów. Kadra tego Zakładu spełniała równocześnie funkcje dydaktyczne, organizując dla studentów wyższych uczelni cykle wykładów na temat konstrukcji i programowania maszyn cyfrowych. Kształcono również użytkowników maszyn: np. kursy programowania w języku Sako ukończyło 1500 osób. Efektywną pomoc na co dzień dla użytkowników maszyn cyfrowych stanowiły stałe konsultacje w Zakładzie. Z podręczników programowania (w języku Sako — 1961, w języku SAS — 1962), wydanych staraniem Zakładu, uczyło się pierwsze pokolenie programistów w Polsce.

Podobną rolę w środowisku akademickim spełniało Centrum Obliczeniowe Polskiej Akademii Nauk (COPAN). Wyposażone w maszynę Ural, stało się poważnym

ośrodkiem naukowych i przemysłowych zastosowań maszyn cyfrowych. W Centrum wykonywano olbrzymie prace obliczeniowe, dotyczące polskich reaktorów. Tutaj też powstał pierwszy podręcznik programowania w Polsce [14].

2. Drugim charakterystycznym rysem okresu pionierskiego rozwoju informatyki polskiej było bardzo wczesne rozpoznanie kluczowej roli, jaką miały odegrać wysokopoziomowe języki programowania (ówcześnie zagadnienie to określano mianem automatyzacji programowania). Już bowiem w 1959 r. zainicjowano w Zakładzie Aparatów Matematycznych PAN prace nad pierwszym w Polsce [13] translatorem języka autokodowego Sako (System Automatycznego Kodowania). Na międzynarodowej konferencji na temat języków programowania (Warszawa, 1961) demonstrowano pracę tego translatora; był to w owym czasie pierwszy autokod działający w krajach socjalistycznych.

Maszyny ZAM-2, które w latach 1961—1964 produkował w krótkiej serii Zakład Doświadczalny Instytutu Maszyn Matematycznych, były wyposażone w translatory języków Sako i SAS (System Adresów Symbolicznych).

W drugim okresie rozwoju informatyki w Polsce, w latach 1963—1970, który nazwiemy okresem przejściowym, z punktu widzenia rozwoju oprogramowania należy wyróżnić następujące trzy ważne wydarzenia:

1. W 1964 roku Instytut Maszyn Matematycznych podjął decyzję opracowania serii maszyn ZAM o wymiennym oprogramowaniu. Decyzją pochodną było podjęcie w pełni samodzielnego opracowania nowoczesnego systemu oprogramowania — SO-141.

2. W połowie lat sześćdziesiątych podjęto decyzję, że kolejna seria maszyn cyfrowych produkowana przez WZE Elwro — seria Odra 1300 — zostanie zaprojektowana tak, aby można było w pełni korzystać z oprogramowania maszyn serii 1900 firmy angielskiej International Computers Ltd. (ówcześnie ICT).

3. W 1968 roku z inicjatywy Polski i ZSRR utworzono Międzynarodową Komisję do spraw rozwoju elektronicznej techniki obliczeniowej w krajach socjalistycznych. Jedną z podstawowych decyzji było przyjęcie zasady, że wszystkie maszyny cyfrowe opracowywane pod egidą tej komisji będzie cechowała wymienność programowa.

Te trzy podstawowe decyzje określiły kierunki prac nad oprogramowaniem w Polsce nie tylko w okresie przejściowym, tj. w latach sześćdziesiątych, lecz także w pięcioleciu 1971—1975, które można nazwać okresem budowy przemysłu informatycznego w Polsce. Omówimy teraz najważniejsze osiągnięcia okresu przejściowego.

Zaczniemy od generalnych skutków wymienionych decyzji.

1. Samodzielne opracowanie systemu oprogramowania SO-141, oprócz efektu bezpośredniego, jakim było wyposażenie maszyn cyfrowych ZAM-41 w nowoczesny (w owym czasie) system operacyjny, przyniosło dużo ważniejsze skutki pośrednie — powstał mianowicie zwarty zespół konstruktorów oprogramowania, przygotowanych do podjęcia jeszcze ambitniejszych zadań.

2. Zaprojektowanie maszyn serii Odra 1300 w sposób umożliwiający pełną akceptację oprogramowania serii ICL 1900 zapewniło wszystkim użytkownikom maszyn Odra 1300 możliwość korzystania z bogatego oprogramowania firmy ICL — w latach sześćdziesiątych europejskiego lidera przemysłu komputerowego. Takie rozwiązanie przyczyniło się też do sukcesu eksportowego serii Odra 1300, której liczne egzemplarze pracują dziś w wielu krajach.
3. Zainicjowanie współpracy wszystkich krajów socjalistycznych w dziedzinie sprzętu i oprogramowania otworzyło perspektywę międzynarodowej wymiany programów i systemów oprogramowania w skali, jaka w dziedzinie współpracy technicznej krajów socjalistycznych nie miała dotychczas precedensu. Wszystkie konsekwencje tej decyzji są jeszcze przed nami, gdyż mimo imponujących efektów współpracy w dziedzinie sprzętu (wystawa Jednolitego Systemu EMC w Moskwie w lecie 1973 r.), w dziedzinie oprogramowania — a zwłaszcza w zakresie programów i systemów zastosowaniowych, wszystko pozostaje jeszcze do zrobienia.

2.2. Prace w dziedzinie oprogramowania w okresach pionierskim (1948—1962) i przejściowym (1963—1970)

W początkach swego rozwoju programowanie było nieodłącznie związane z konkretną maszyną cyfrową, dlatego też chronologia tego punktu została określona pojawianiem się kolejnych maszyn cyfrowych konstrukcji krajowej [34].

Maszyny XYZ i ZAM-2

Tworząc oprogramowanie tych maszyn kierowano się następującymi zasadami:
— język wyższego poziomu i język symboliczny tworzą jeden system;
— programy biblioteczne powinny być dostępne zarówno w języku symbolicznym, jak i języku wyższego poziomu.

Konsekwentna realizacja oprogramowania w myśl tych zasad doprowadziła do wyposażenia maszyn XYZ i ZAM-2 w następujące elementy oprogramowania:
— system operacyjny (bardzo uproszczony, lecz skuteczny);
— język symboliczny SAS;
— język autokodowy Sako;
— obszerną bibliotekę programów.

Pierwsze maszyny z serii ZAM-2 instalowano u użytkowników, którzy byli już przygotowani do ich intensywnego wykorzystania (Wyższa Szkoła Łączności w Zegrzu, Biuro Projektów Syntezy Chemicznej w Gliwicach, Instytut Lotnictwa). Wpłynęło to na szczególną staranność w przygotowaniu bogatego jak na ówczesne warunki oprogramowania.

Zastosowania maszyn w zarządzaniu

Próby stosowania maszyn cyfrowych w zarządzaniu podjęto w Polsce już w latach 1961—1962. Maszyna ZAM-2, na której prace te prowadzono, była przystosowana przede wszystkim do obliczeń numerycznych o charakterze naukowo-technicznym. Ponadto była to maszyna mała wg dzisiejszych standardów. Wywarło to swoje piętno na zastosowaniach — które dziś można by nazwać prostymi, ale które w owym czasie wcale proste nie były. Co więcej, ograniczoność środków technicznych zmuszała do dogłębnego dopracowywania projektów. Dbając o różnorodność rozwiązań (chodziło bowiem o zebranie możliwie szerokich doświadczeń w dziedzinie zastosowania maszyn cyfrowych w zarządzaniu), zaprojektowano i uruchomiono m.in. systemy gospodarki materiałowej (ewidencja, planowanie, dysponowanie), ewidencję eksploatacji pojazdów, planowanie produkcji (współpraca z Hutą Warszawa), planowanie i rozliczanie transakcji handlu zagranicznego (współpraca z Centralą Eksportu Kompletnych Obiektów Przemysłowych — CEKOP, Ministerstwem Handlu Zagranicznego), rozliczanie produkcji wyrobów gotowych (Zakłady Przemysłu Odzieżowego „Cora"), statystyczne opracowywanie wyników ankiet (Szkoła Główna Planowania i Statystyki, Uniwersytet Warszawski, Politechnika Warszawska i inne uczelnie). Jedną z najciekawszych prac były unikalne w owym czasie w kraju badania na temat przydatności czytnika dokumentów o ręcznie nanoszonych kreskach (zwykłym ołówkiem) w zastosowaniach administracyjnych.

W miarę rozwoju sprzętu (maszyny ZAM-21, ZAM-41) podejmowano w Instytucie Maszyn Matematycznych prace nad coraz większymi systemami. Dbając ciągle o szeroki zakres prowadzonych badań, nawiązywano kontakt z różnymi użytkownikami, reprezentującymi krańcowo różne problemy. Dla przykładu, zajmowano się w owym okresie programami fakturowania usług i kontroli wpłat abonentów telefonicznych i radiowych, normatywnym rachunkiem kosztów (dla Zakładów Przemysłu Cukierniczego im. 22 Lipca), elementami planowania produkcji (dla Zakładów Mechanicznych im. M. Nowotki), systemami przetwarzania informacji tekstowych (wyszukiwanie i indeksowanie — dla Instytutu Informacji Naukowej, Technicznej i Ekonomicznej). W ostatnich latach okresu przejściowego główny nacisk położono na typizację rozwiązań, zarówno w zakresie organizacji, jak i realizacji systemów elektronicznego przetwarzania danych. Należy tutaj wymienić typowy system gospodarki materiałowej, typowy system opracowywania wyników ankiet i przede wszystkim pakiet programów parametryzowanych, specjalnie przystosowany do typowych czynności przetwarzania danych (system Opus, zrealizowany na maszynie ZAM-41).

Równocześnie z realizacją i rozwijaniem wymienionych systemów prowadzono badania w zakresie technologii przetwarzania danych: metod przygotowania danych, metod kontroli poprawności danych, metod realizacji poszczególnych czynności — sortowania i in.

Oprogramowanie maszyny ZAM-41

Było to w owym czasie największe przedsięwzięcie z dziedziny oprogramowania maszyn o całkowicie oryginalnej polskiej konstrukcji. Prace rozpoczęto w 1967 r. i prowadzono w zespole ok. 60 programistów przez okres ok. 4 lat. Ilustracją możliwości obliczeniowych, uzyskanych w owym czasie na maszynie ZAM-41, oraz wielkości opracowanego oprogramowania, mogą być tabl. 2.1 i 2.2 [6].

Tablica 2.1. Dane dotyczące możliwości obliczeniowych maszyny ZAM-14 [6]

Nazwa zadania	Czas działania
Opracowanie (obejmujące sortowanie danych na 4 stacjach magnetycznych) 50 tys. dokumentów 30-znakowych zapisanych na taśmie magnetycznej i wypisanie sprawozdań w pięciu przekrojach	2 h
Opracowanie 1000 dokumentów obejmujące: wczytanie z taśmy papierowej, posortowanie w PAO i wypisanie sprawozdania	5 min
Przejrzenie taśmy magnetycznej zawierającej 2700 dokumentów 600-znakowych z wybraniem pozycji według zadanego klucza, bez czasu drukowania wybranej pozycji	1 min
Czas sortowania 100 tys. pozycji 8-znakowych na 4 taśmach magnetycznych (pozycje były przed sortowaniem ułożone w odwrotnej kolejności)	56 min
Rozwiązywanie prostego zagadnienia simpleksowego zmodyfikowaną metodą Dantzinga dla macierzy o wymiarach 37 × 34	20 min
Analiza drogi krytycznej (czas, koszty) dla sieci o 583 działaniach	3 min
Rozwiązanie prostego zagadnienia transportowego (bez ograniczeń i węzłów swobodnych) zmodyfikowaną metodą Forda-Fulkersona dla macierzy o wymiarach 169 × 33	7 s (nie licząc czasu wczytywania i drukowania danych)

W następnych latach powstała rozwinięta wersja systemu oprogramowania, znana jako SO-141, najpełniejsza i znajdująca się w eksploatacji do dnia dzisiejszego w tych ośrodkach obliczeniowych, które używają maszyny ZAM-41.

Przy opracowywaniu systemu oprogramowania SO-141. przyjęto założenie, że system operacyjny musi dać się dostosować do różnych konfiguracji maszyny. Zadanie to rozwiązano w pełni przez zastosowanie oryginalnej koncepcji generowania systemów oprogramowania.

Dla maszyny ZAM-41 utworzono dwa języki symboliczne: jeden przeznaczony dla konstruktorów oprogramowania — Pjeg (Podstawowy język generowania), drugi zaś — dla użytkowników nie związanych z budową oprogramowania — SAS-41. Liczne możliwości gospodarki pamięcią, przechodzenia od kompilacji do obliczania i odwrotnie, oraz gospodarka częściami programu czyniły z języka Pjeg bardzo dobre narzędzie do konstruowania oprogramowania. Język SAS z rozbudowanym aparatem makrodefinicji i tworzenia stałych był podstawowym narzędziem tworzenia biblioteki programów.

Tablica 2.2. Dane dotyczące liczby rozkazów zrealizowanego systemu oprogramowania [*16*]

Nazwa pozycji	Liczba rozkazów
System oprogramowania SO 41 wraz z już funkcjonującymi elementami SO 141	120 000
System operacyjny obejmujący Supervisor oraz zespół programów nadzorujących wykonanie problemu	6 000
Supervisor zabezpieczający m.in. obsługę przerwań i współpracę z urządzeniami zewnętrznymi	4 000
Translator JOM (Język służący do opisywania czołówki programu)	1 500
Translator Pjes	2 500
Translator SAS	10 000
Translator EOL	4 000
Interpreter EOL	2 500
Translator Algol	6 000
Interpreter Algol	5 000
Translator Pjeg	3 000
Biblioteka podprogramów i programów do przetwarzania danych, obejmująca głównie programy kopiujące, współpracy z TM i innymi urządzeniami we/wy	7 500
Programy z zakresu zastosowań do przetwarzania danych (Systemy)	4 000
Biblioteka podprogramów i programów z zakresu: metod numerycznych, metod programowania liniowego, metod transportowych oraz programowania sieciowego	2 400

System oprogramowania SO-141 obejmował m.in. translatory takich języków wysokopoziomowych, jak:

Cobol
Algol 60
Sako
Cemma
ZAM-GPSS
Astek
EOL

Ze względu na powszechną znajomość pierwszych dwóch języków podamy tylko, że przy realizacji Cobolu uwzględniono niemal pełny jego podzbiór wg Proposed USA Standard Cobol (*ACM SIGPLAN Notices*, Vol. 2 ,No. 2, 1967). Ponadto wprowadzono wiele nowych elementów, bądź to inspirowanych konstrukcją maszyny ZAM-41 (maszyna o długości słowa równego 24 bitom, a nie znakowa), bądź to wynikających z rodzimych tradycji przetwarzania danych za pomocą taśm dziurkowanych, na których w sposób naturalny informacja jest zapisywana niepozycyjnie [20].

Wiele przebiegów translatora Cobol realizowano w Języku EOL. Umożliwiło to m.in. sporządzenie bardzo czytelnej dokumentacji translatora i było prawdopodob-

nie pierwszą w krajach socjalistycznych udaną próbą napisania translatora języka wyższego poziomu w innym, specjalizowanym języku do programowania systemowego. Wydaje się też, że była to pierwsza samodzielna realizacja translatora języka Cobol w krajach socjalistycznych, oddana do normalnej eksploatacji.

Język Cemma [19] jest wyspecjalizowanym językiem symulacyjnym; znanym przykładem tej klasy języków jest CSMP (Continuous Systems Modelling Program) firmy IBM. Umożliwia on modelowanie procesów opisywanych liniowymi i nieliniowymi równaniami różniczkowymi zwyczajnymi o współczynnikach zależnych lub niezależnych od czasu. Modele takich procesów mogą być poddawane dowolnym wymuszeniom zdeterminowanym oraz zakłóceniom stochastycznym. Język Cemma dzięki swej prostocie i elastyczności pozwala w łatwy sposób opisywać złożone układy dynamiczne. Jest to język nieproceduralny — programy dla maszyny cyfrowej są przygotowywane na podstawie odpowiednich schematów blokowych reprezentujących symulowany model.

Programista piszący program w języku Cemma ma do dyspozycji zestaw elementów blokowych (ok. 30 rodzajów operacji) realizujących funkcje matematyczne, generatory funkcji, funkcje nieliniowe, funkcje logiczne, a także elementy pamiętające wartości ekstremalne. Możliwa jest symulacja złożonych i dużych układów, gdyż liczba elementów na schemacie blokowym może wynosić nawet 500.

Z innych własności języka Cemma na maszynie ZAM-41 warto wymienić:

— możliwość powtarzania obliczeń ze zmianą wartości parametrów;

— możliwość automatycznego doboru parametrów przez maszynę na podstawie określonego kryterium, czyli optymalizacji modelu ze względu na zadane kryterium;

— możliwość występowania na schemacie zdefiniowanych pętli algebraicznych, czyli możliwość rozwiązywania funkcji uwikłanych;

— siedem różnych algorytmów całkowania (sześć ze stałym i jeden ze zmiennym krokiem całkowania);

— możliwość jednoczesnego rejestrowania do 50 przebiegów w postaci numerycznej i do 50 wykresów graficznych (wykonywanych na drukarce wierszowej).

Język ZAM-GPSS jest językiem przeznaczonym do symulowania procesów dyskretnych. Stanowi on adaptację do maszyny ZAM-41 języka GPSS/360, opracowanego w firmie IBM dla maszyn IBM/360. Model w języku ZAM-GPSS składa się z ciągu bloków, z których każdy opisuje pewną jednostkową czynność zachodzącą w modelowanym systemie. Schemat blokowy modelu może być formalnie rozpatrywany jako zbiór wzajemnie powiązanych symboli operatorów logicznych i arytmetycznych, reprezentujących własności modelowanego systemu lub procesu, które są interesujące dla użytkownika. W każdym procesie można wyróżnić grupę elementarnych wielkości i pojęć abstrakcyjnych, za pomocą których można skonstruować opis (model) tego procesu.

Dla celów symulacji procesów dyskretnych w języku ZAM-GPSS zdefiniowano 13 typów wielkości:

bloki	funkcje arytmetyczne
zadania	funkcje boolowskie
urządzenia	kolejki
magazyny	tablice
przełączniki logiczne	pamięci
zmienne arytmetyczne	listy (zbiory) użytkownika
zmienne boolowskie	

oraz wprowadzono określone konwencje i założenia formalne odpowiadające podstawowym czynnościom lub warunkom, występującym w rzeczywistym procesie. Z każdą z tych podstawowych wielkości są związane zbiory atrybutów numerycznych lub logicznych, opisujących własności i stan tych wielkości w czasie.

Proces symulacji można rozważać jako ciągi wzajemnie powiązanych, oddziałujących na siebie zdarzeń, przy czym przez zdarzenie rozumiemy każdą zmianę stanu modelu, następującą pod wpływem wzajemnych oddziaływań między wielkościami. Stanem modelu jest zbiór wartości numerycznych i logicznych wszystkich zdefiniowanych w nim parametrów, atrybutów i wskaźników w wyróżnionej chwili.

Język Astek (Analiza Statystycznych Danych Eksperymentalnych) [18] służy do opisu i programowania wstępnej obróbki eksperymentalnych danych otrzymywanych z rzeczywistego procesu. Pozwala na wygodne dokonanie identyfikacji właściwości statycznych i dynamicznych badanego obiektu przy użyciu metody regresji, korelacji i analizy widmowej.

Struktura języka systematyzuje opis problemów obliczeniowych, formalizuje sposób korzystania ze zbioru procedur, ułatwia programowanie oraz umożliwia szybkie badanie wielu wariantów analizy. Operatory logiczne pozwalają podejmować podczas obliczeń proste decyzje dotyczące selekcji danych pomiarowych i wyboru określonych parametrów oraz wyboru wariantów analizy. W zbiorze instrukcji języka, oprócz instrukcji dotyczących działań na zbiorach, wspomnianych już prostych operacji logicznych i arytmetycznych, znajdujemy następujące instrukcje specjalne:

SQR służy do wyciągania pierwiastka kwadratowego; operacja taka może być automatycznie wykonywana na kolejnych elementach zadanego zbioru.

LOGARYTM, EXPONENT służy do obliczania odpowiednich funkcji z elementów zbioru, parametru lub liczby.

ŚREDNIA, WARIANCJA służy do działania na zbiorach.

EKSTREMUM umożliwia znalezienie elementu minimalnego lub maksymalnego danego zbioru.

DYSTRYBUANTA służy do znajdowania empirycznej dystrybuanty zmiennej losowej, której wartości są reprezentowane przez elementy zbioru (specyfikatorami instrukcji określa się liczbę odcinków podziału oraz identyfikator zbioru).

CHITEST, LAMBDATEST służą do weryfikowania hipotezy o normalności zmiennej losowej, reprezentowanej przez elementy zbioru — odpowiednio za pomocą testu χ^2 i testu λ Kołmogorowa.

Instrukcja REGRESJA i związane z nią instrukcje:
WE — wektory własne
MINIMAX
FTEST
służą do wykonywania analizy właściwości statycznych tego eksperymentu; współczynniki modelu liniowego tego eksperymentu obliczone za pomocą programu REGRESJA mogą być następnie zweryfikowane testem F. Fishera (FTEST).
Do badania uzyskanego modelu oraz związków między poszczególnymi zmiennymi służą instrukcje obliczania wartości własnych macierzy struktur i odpowiadających tym wartościom wektorów własnych, obliczania ekstremum powierzchni kwadratowej (MINIMAV) oraz obliczania współczynników korelacji między zbiorami pomiarów (KORELACJA — instrukcja ta może również służyć do obliczania autokorelacji). Do badania własności dynamicznych modelu stosuje się analizę widmową. Użytkownik języka Astek dysponuje w tym zakresie funkcjami FFT (Fast Fourier Transform) i WIDMO. Funkcja FFT służy do obliczania współczynników rozwinięcia w szereg Fouriera na podstawie algorytmu opracowanego przez J. W. Cooleya i J. W. Tukeya w 1965 r., natomiast WIDMO oblicza funkcje gęstości widmowej.
Już ten krótki opis pozwala się zorientować, że Astek jest bardzo silnym i wygodnym narzędziem analizy statycznej i dynamicznej danych eksperymentalnych. Język Astek jest używany w Polsce przede wszystkim w instytutach przemysłu chemicznego.
Język EOL jest przeznaczony do przetwarzania symboli. Zakres jego zastosowań jest dość szeroki, aczkolwiek pierwotnie był on pomyślany przede wszystkim jako narzędzie do pisania kompilatorów języków wysokopoziomowych. W tej roli był stosowany np. przy pisaniu kompilatorów języków Cobol (wspomniane uprzednio) i Algams na maszyny ZAM-41. Prace nad językiem EOL rozpoczęto już w 1966 r.; można go więc zaliczyć do pierwszych przedstawicieli języków programowania systemowego [15].

System Opus

Pakiet programów parametryzowanych Opus [17] był wynikiem prac teoretycznych nad metodami przetwarzania (technologią) danych do zastosowań administracyjno-gospodarczych.
Prace nad pakietem Opus rozpoczęto w 1967 r., a więc wtedy, gdy idea programowania parametryzowanego dopiero zaczynała zyskiwać popularność. Przy opracowywaniu pakietu przyjęto następujące założenia:
— poszczególne programy powinny wykonywać typowe czynności z zakresu przetwarzania danych;
— działanie każdego programu powinno zależeć od parametrów, wczytywanych przez program na początku działania;

— struktura parametrów dla różnych programów powinna być możliwie jednolita;
— dla programu o prostych funkcjach, parametry powinny być proste; w miarę komplikowania się funkcji programu, struktura parametrów może się stawać bardziej złożona;
— w pierwszej wersji programów należy ograniczyć się do zbiorów sekwencyjnych i działań metodą interpretacji — w dalszych wersjach należy rozważyć metodę generowania i działanie na zbiorach niesekwencyjnych.
Jako czynności typowe dla przetwarzania danych przyjęto:
— kontrolę danych na maszynowych nośnikach informacji (papierowych);
— konwersję danych z papierowych nośników informacji na taśmę magnetyczną;
— wielotaśmową konwersję zbiorów;
— sortowanie, łączenie;
— redagowanie i drukowanie wyników;
— czynności pomocnicze, jak np. zakładanie, kontrola oraz powielanie zbioru danych.
Realizację pakietu Opus zakończono w 1971 r. Składa się on z 13 programów obejmujących w sumie ok. 40 tys. rozkazów. Pakiet stosuje się m.in. w Systemie Ewidencji i Rozliczeń Wyrobów Gotowych opracowanym dla Zakładów Przemysłu Odzieżowego „Cora". System ten posiada trzy ewidencje, aktualizowane w cyklu miesięcznym i tworzy ponad 20 sprawozdań. Składa się on z ponad 80 programów, z których tylko 3 trzeba było napisać specjalnie dla tego systemu, a pozostałe 90% wygenerowano przy użyciu systemu Opus.
Zastosowanie programów Opus w tym systemie było praktycznym sprawdzeniem ich przydatności; umożliwiało jednocześnie istotne skrócenie łącznego czasu projektowania i wdrażania systemu. Praktyka wykazała, że przygotowanie parametrów i sprawdzenie działania programu pochodzącego z pakietu Opus zajmuje ok. 5 dni roboczych. Łatwo też można rozbudowywać system i modyfikować go, gdy potrzeby przedsiębiorstwa dyktują konieczność wprowadzenia zmian.

Maszyna ZAM-41 w systemach pracujących na bieżąco

Na zakończenie omawiania oprogramowania maszyny ZAM-41 podamy dwa przykłady jej stosowania w systemach pracujących na bieżąco (*ang.* on-line), związanych z transmisją danych.
Już w latach 1966—1967 opracowano w Instytucie Maszyn Matematycznych eksperymentalny system transmisji danych z maszyną cyfrową ZAM-41 alfa. Celem pracy było zbadanie możliwości budowy systemu transmisji danych przy użyciu komutowanego łącza miejskiego, pozwalającego na synchroniczne, bezpośrednie wprowadzanie informacji z łącza telefonicznego do maszyny cyfrowej oraz

na przesłanie w łącze przetworzonej informacji. Stosowano urządzenie transmisji danych UTD 1 o dwóch przełączalnych szybkościach 600 i 1200 bodów, próby przesyłania prowadzono na 10-kilometrowym odcinku komutowanego łącza miejskiego. Badania eksperymentalnego systemu prowadzono przez ok. 100 godzin — uzyskano stopę błędów ok. $4 \cdot 10^{-4}$, co odpowiadało wartościom spotykanym w literaturze dla komutowanych łączy telefonicznych, przy szybkościach transmisji od 600 do 1200 bodów.

Drugim, dużo większym przedsięwzięciem programowym w tej dziedzinie było opracowanie w latach 1967—1968 systemu oprogramowania maszyny ZAM-41 do pracy na bieżąco w sieci transmisji danych, w której dominującą rolę odgrywało zbieranie danych (*ang.* data acquisition).

System ten wykazywał wszystkie cechy istotne dla tej klasy zastosowań, a przede wszystkim:

— możliwość bezpośredniej współpracy z odległymi urządzeniami końcowymi;

— możliwość manipulowania informacją przesyłaną między urządzeniami końcowymi a jednostką centralną w postaci zwartych „paczek", zwanych dalej meldunkami, napływającymi w momentach przypadkowych;

— możliwość wyróżniania różnych stopni ważności przesyłanych meldunków.

2.3. Prace w dziedzinie oprogramowania w okresie „przemysłowym" polskiej informatyki (1971-1975)

Próbując scharakteryzować okres „przemysłowy" polskiej informatyki w aspekcie prac nad oprogramowaniem, można, uwzględniając brak wystarczającego dystansu czasowego, wymienić następujące najważniejsze cechy:

1. Instytuty (takie jak Instytut Maszyn Matematycznych i Centrum Obliczeniowe PAN) oraz wyższe uczelnie techniczne i uniwersytety przestały być monopolistami w dziedzinie opracowywania oprogramowania typu podstawowego oraz zastosowaniowego o charakterze uniwersalnym; ich rolę w tym zakresie przejęły inne organizacje.

2. Opracowaniem, rozszerzaniem, popularyzowaniem i utrzymywaniem oprogramowania podstawowego zaczęli się zajmować w tym okresie producenci sprzętu informatycznego lub ich specjalne agendy. Na przykład dla maszyn Odra 1300 oraz maszyn Jednolitego Systemu EMC (R-32) funkcje te sprawują przede wszystkim Wrocławskie Zakłady Elektroniczne Mera-Elwro oraz Elwro-Service, a dla systemów minikomputerowych Mera 300 — ich producent, czyli Zakłady Wytwórcze Przyrządów Pomiarowych Era.

3. Poważnym wytwórcą mniej lub bardziej uniwersalnego oprogramowania zastosowaniowego stały się Zakłady Elektronicznej Techniki Obliczeniowej (ZETO), rozmieszczone na terenie całej Polski i należące do Zjednoczenia Informatyki.

4. Twórcami najciekawszych programów problemowych zostają coraz częściej zespoły fachowców danej dziedziny techniki, którzy zdobyli odpowiednią wiedzę w zakresie „kuchni" informatycznej. Korzystając (najczęściej) z wysokopoziomowych języków programowania, tworzą bardzo ciekawe programy zastosowaniowe — szczególnie w dziedzinie automatyzacji projektowania.
5. Zespoły profesjonalnych informatyków w instytutach naukowo-badawczych, uczelniach technicznych i uniwersytetach podejmują bardziej perspektywiczne zagadnienia programowania (dotychczasowe obciążenie zagadnieniami produkcji oprogramowania nie pozwalało na szersze prowadzenie takich prac). Można do nich zaliczyć:
— budowę metatranslatorów;
— języki do opisu semantyki programów;
— języki dostępu i definicji banków danych;
— języki konwersacyjne i rozszerzalne;
— programowanie heurystyczne.
Uwzględniając tylko najistotniejsze fakty z dziedziny produkcyjnej działalności przemysłu informatycznego w Polsce w zakresie oprogramowania, należy wymienić następujące pozycje.

Dyskowy system operacyjny DOS/JS

Wrocławskie Zakłady Elektroniczne Mera-Elwro wyposażają produkowane przez siebie maszyny R-32 w dyskowy system operacyjny Jednolitego Systemu EMC — DOS/JS [22], opracowany wspólnym wysiłkiem krajów socjalistycznych, a następnie adaptowany i konserwowany przez każdego z krajowych producentów.
W skład dyskowego systemu operacyjnego DOS/JS wchodzą translatory i programy usługowe, pracujące pod kontrolą programu sterującego. System operacyjny DOS/JS wymaga jednostki centralnej o minimalnej pojemności pamięci operacyjnej 32 kb i co najmniej jednej pamięci dyskowej.
System operacyjny DOS/JS zapewnia użytkownikowi możliwość tworzenia bibliotek programów w pamięci dyskowej, przy czym programy mogą być zapisane w bibliotece w postaci źródłowej, wynikowej (tj. po ich przetłumaczeniu przez odpowiedni translator) lub w postaci gotowej do wykonywania (tj. po skonsolidowaniu przez program łączący). Programy źródłowe mogą być złożone z segmentów pisanych w różnych językach, a następnie połączonych w jeden program. W przypadku dużych programów przy stosunkowo małej pamięci operacyjnej można stosować w systemie operacyjnym DOS/JS tzw. nakładanie programu. Polega ono na kolejnym wprowadzeniu z pamięci zewnętrznej do pamięci operacyjnej aktualnie koniecznych segmentów na miejsce zajmowane przez segmenty już niepotrzebne (stąd nazwa „nakładanie").
System operacyjny DOS/JS umożliwia wieloprogramowość (do trzech programów) ze stałymi obszarami programowymi w pamięci operacyjnej, których wielkości zo-

stają określone w czasie generowania systemu. W skład systemu operacyjnego DOS/JS, oprócz niezbędnych programów usługowych, takich jak:
— program łączący (*ang.* linkage editor);
— bibliotekarz (*ang.* librarian);
— programy sortowania;
— systemowe programy pomocnicze;
— autotest;
wchodzą translatory następujących języków programowania
— Assembler;
— Fortran IV (Basic Fortran IV);
— Report Program Generator (RPG);
— PL/I.
Należy dodać, że dla maszyn R-32 przewidziano znacznie doskonalszy system operacyjny — OS/JS.

Oprogramowanie maszyn serii Odra 1300

Wrocławskie Zakłady Elektroniczne Mera-Elwro dostarczają maszyny serii Odra 1300 wraz z bogatym oprogramowaniem podstawowym i zastosowaniowym ([23], [24]). Oprogramowanie to może pracować pod kontrolą prostego programu sterującego, tzw. Egzekutora (Executive firmy ICL dla serii 1900) bądź pod kontrolą bardziej rozbudowanych systemów operacyjnych George 1, 2 lub 3, przewidzianych przez firmę ICL dla większych zestawów serii 1900. Orientacyjne zestawienie tego bogatego oprogramowania przedstawiono na rys. 2.1.
Szczególnie interesującą inicjatywą Zakładów Mera-Elwro jest nawiązanie ścisłej współpracy z Wyższą Szkołą Ekonomiczną we Wrocławiu w zakresie wspólnego opracowania nowoczesnego, przystosowanego do specyfiki zarządzania zakładu w warunkach socjalistycznej gospodarki, automatycznego systemu przetwarzania informacji — KOSIP.

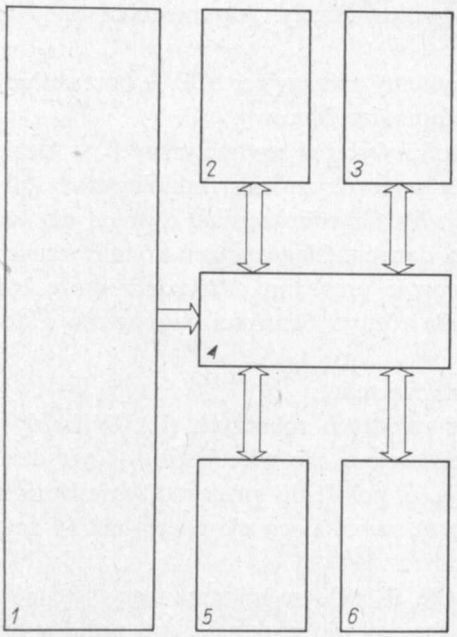
Oprogramowanie systemów minikomputerowych Mera 300

Staranne rozwiązanie oprogramowania minikomputerów ma szczególne znaczenie wobec niewielkiej (zazwyczaj) pamięci operacyjnej i skromnej listy rozkazów. Oprogramowanie zestawów minikomputerowych Mera 300 ma strukturę trójpoziomową:

poziom 0 — narzędzia programowe do budowy oprogramowania;

poziom 1 — systemy operacyjne zbudowane na narzędziach programowania;

poziom 2 — programy użytkowe pracujące pod nadzorem odpowiednich systemów operacyjnych.



Rys. 2.1. Główne elementy oprogramowania maszyn cyfrowych serii Odra 1300
1 — pakiety zastosowań (obliczenia z zakresu inżynierii budowlanej; pakiety obliczeń systemów; pakiety obliczeń systemów energetycznych; sterowanie obrabiarek; analiza sieci wodnej, gazowej i ciepłowniczej; programowanie liniowe; problemy transportowe; analiza sieciowa; systemy bilansowania zapasów; Prompt — system kierowania i kontroli produkcji; Pert — dla przeprowadzenia analizy czasu i kosztów przydziału środków i planowania wieloprojektowego; Scan — kontrola zapasów magazynowych; Find — system operowania zbiorami; Nic — system tworzenia indeksów i katalogów; Nimms — zintegrowany system zarządzania i kierowania produkcją i wiele innych); *2* — *egzekutor* (modułowy program sterujący ułatwiający pracę operatora i nadzorujący wieloprogramową pracę maszyny; *3* — *systemy operacyjne* (stanowiące rozszerzenie czynności egzekutora i automatyzujące obsługę programów George 1, 2, 3; MOP — system operacyjny dla pracy wielodostępnej); *4* — *maszyna cyfrowa serii Odra 1300*; *5* — *języki programowania* (Plan, Cobol, Algol, Fortran i inne); *6* — *biblioteka podprogramów* (gotowe do użytku segmenty programowe, które można włączyć do każdego programu)
[Reprodukowano za zezwoleniem Ośrodka Badawczo-Rozwojowego Maszyn Cyfrowych „Elwro"]

W skład poziomu 0 wchodzi program Nucleus, służący do sterowania pracą programu użytkowego i jego uruchamiania oraz translator języka symbolicznego Sawik. Nucleus umożliwia wprowadzanie i wyprowadzanie programów w postaci binarnej, steruje obsługą przerwań programowych, dostarcza programiście podstawowych usług w zakresie uruchamiania programów jak np. startowanie programu od wskazanego rozkazu, wstawianie rozkazu pod wskazany adres, wypisywanie zawartości rejestrów, wypisywanie zawartości pamięci itp.
Język Sawik udostępnia programiście rozkazy składające się z części operacyjnej w postaci dwuliterowego skrótu mnemotechnicznego i wyrażenia adresowego. W skład wyrażenia mogą wchodzić cztery operacje arytmetyczne o jednakowej

mocy, wykonywane kolejno od lewej do prawej strony. Argumentami operacji mogą być: liczba, etykieta lub zmienna.

Aktualnie na poziomie 1 są dostępne dwa systemy operacyjne: SOWA oraz służący do automatyzacji prac biurowych system Komputer Biurowy.

System SOWA umożliwia obliczanie wartości wyrażeń arytmetycznych w trybie dialogowym. Wyrażenia arytmetyczne mogą zawierać: zmienne jednoliterowe, stałe liczbowe, działania arytmetyczne, nawiasy. W systemie można również używać operacji do wprowadzania i wyprowadzania danych. Ciąg wyrażeń arytmetycznych i operacji wprowadzania danych może tworzyć procedurę. Procedura może być wielokrotnie wywoływana i wykonywana dla różnych wartości parametrów aktualnych.

System Komputer Biurowy daje użytkownikowi m.in.:

1. Możliwość wykonywania działań na 16 rejestrach roboczych (R). W każdym rejestrze może być umieszczona liczba zawierająca nie więcej niż 14 cyfr dziesiętnych, znak + lub — oraz informacja o położeniu przecinka dziesiętnego. W rejestrze R można także umieszczać tekst, zawierający nie więcej niż 16 znaków.
2. Dostęp do 256 rejestrów pomocniczych (P). W każdym rejestrze tego typu można przechowywać informację tekstową o długości nie większej niż 8 znaków lub liczbę o formacie jak wyżej, w postaci upakowanej.
3. Listę 59 instrukcji: arytmetycznych, logicznych, przesłań między rejestrami RR, RP, PR, wprowadzania i wyprowadzania informacji, skoków warunkowych i bezwarunkowych, definiowania etykiet i podprogramów.

Zbiór programów użytkowych (poziom 2) jest systematycznie rozbudowywany przez zespół programistów producenta zestawów Mera 300. Aktualnie programy tego poziomu są pisane przy pomocy instrukcji dostępnych w systemie Komputer Biurowy i obejmują trzy grupy programów: do obliczeń z dziedziny statystyki matematycznej, do obliczeń inżynieryjnych oraz do automatyzacji prac biurowych.