

P. 3057/89



1
'89

techniki komputerowe

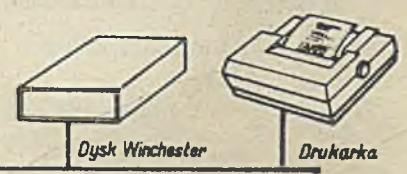


BIULETYN INFORMACYJNY

Stanowiska uczniowskie



Urządzenia wspólne



INSTYTUT MASZYN MATEMATYCZNYCH
BRANŻOWY OŚRODEK INTE

Rysunek na okładce: Klasa szkolna wyposażona w sieć
mikrokomputerową

Druk IMM zam.16/89 nakł. 1100 egz.



P.3057/89

TECHNIKI KOMPUTEROWE

Rok XXVII

Nr 1

1989

Spis treści

GUCAŁO Walentin W.: Algorytm określania zasobów obciążeniowych dla wsadowego trybu przetwarzania w warunkach ładowania zdeterminowanego	3
PIETROW A.S.; PADALKO A.G., SAKUN A.P.: Komputerowe urządzenia treningowe do szkolenia operatorów skomputeryzowanych urządzeń produkcyjnych....	7
PAPST Adam: System autorski AUTOOL	23
LIBIZOWA Ludmiła W.: Inteligentne komputerowe systemy wspomagające naukę	33
LIEBIEDIEWA Margarita B.: Dokumentacja oprogramowania dydaktycznego - wymagania ogólne	39
RAFF Andrzej: Ocena oprogramowania dydaktycznego - różne podejścia	49
Nowości techniczne. Oprac. Jan Ryżko	61
Spis treści rocznika 1988	

DWUMIESIĘCZNIK

Wydaje:

INSTITUT MASZYN MATERATYCZNYCH

ul. Krzywickiego 34 02-078 WARSZAWA tel. 28-37-29

GAŁŹOWY OŚRODEK INFORMACJI

NAUKOWEJ. TECHNICZNEJ

I EKONOMICZNEJ

Komitet Redakcyjny:

dr inż. Stanisława BONKOWICZ-SITTAUER (redaktor naczelny),
mgr Hanna DROZDOWSKA-STRZEMIŃSKA (sekretarz redakcji), mgr inż. Zdzisław GROCHOWSKI,
mgr inż. Jan KLIMOWICZ, dr inż. Piotr PERKOWSKI, mgr inż. Romuald SYNAK

kandydat nauk technicznych Walentin Wasiliewicz GUCAŁO
Instytut Inżynieryjno-Budowlany
Wilno

Algorytm określania zasobów obliczeniowych dla wsadowego trybu przetwarzania w warunkach ładowania zdeterminowanego

Jednym z podstawowych trybów przetwarzania jest tryb wsadowy. Istotną jego właściwością jest na ogół długi okres czasu, jaki zwykle upływa pomiędzy zgłoszeniem danego programu użytkowego, a rozpoczęciem jego przetwarzania. Czas ten może wynosić nawet parę dni. Jednym z problemów, które należy rozwiązać przy projektowaniu systemu z wsadowym trybem przetwarzania jest określenie minimalnych ilości zasobów systemu działającego w warunkach zdeterminowanego ładowania, a więc działającego na podstawie zgłoszeń do wykonania poszczególnych programów.

W opracowaniu A.B. Barskiego [1] wykazano, że efektywne uzyskanie rozwiązania zadań tego typu jest możliwe ze względu na ich kombinatoryczny charakter. Wobec tego poszukiwanie dostatecznie efektywnych suboptymalnych rozwiązań można oprzeć w tej sytuacji na podejściu heurystycznym [2].

Rozpatrzmy wsadowy tryb przetwarzania, w którym ładowanie można wyrazić jako zdeterminowany potok programów użytkowych o żądanych czasach rozpoczęcia, zadanych długościach okresów przetwarzania i ściśle określonych terminach zakończenia przetwarzania. Przypuśćmy, że każdy l -ty program użytkowy, może być wykonywany bez zawieszania przez dowolny procesor spośród poszukiwanej liczby procesorów. Należy określić minimalną liczbę procesorów - Z , która pozwoli na przetworzenie każdego ze zgłoszonych programów użytkowych w ściśle określonym terminie.

Oznacza się zadaną intensywność przetwarzania l -tego programu użytkowego w rozpatrywanym przedziale czasu jako S_l . Oznacza to, że przetwarzanie l_s -tego programu użytkowego, musi być rozpoczęte w chwili d_{l_s} i zakończone w terminie D_{l_s} . Przyporządkujmy przedziałom czasowym o długości T^P , rozpoczynając od chwili $t_0 = 0$, numery $1, 2, \dots, P$. Zakładamy ponadto, że przedział czasowy o numerze p ma postać $(t_{p-1}, t_p]$. Wobec tego, jeżeli przetwarzanie l_s -tego programu użytkowego rozpoczyna się w przedziale czasowym p , to może być on przetwarzany w przedziałach o numerach $p, p+1, \dots, D_{l_s}$. Wprowadzamy zmienną τ_{l_s} , oznaczającą chwilę zakończenia przetwarzania l_s -tego programu użytkownika.

Ogólnie biorąc przy przetwarzaniu programów użytkowych może zaistnieć sytuacja, w której

$$\tau_{l_s} \in (t_{p-1}, t_p];$$

$$\tau_{l_s} - t_{l_s} \in (t_{p-2}, t_{p-1}),$$

gdzie: t_1 - czas przetwarzania 1-tego programu użytkowego

Czyli, że przetwarzanie 1_g-tego programu użytkowego rozpoczyna się w (p-1)-szym przedziale czasowym, a kończy w p-tym przedziale czasowym.

Dopuszczamy również, że w rozpatrywanym systemie spełniony jest warunek:

$$t_e \ll T^p, \quad e = \overline{1, L}; \quad p = \overline{1, P} \quad (1)$$

Warunek (1) pozwala wprowadzić zbiór zmiennych sterujących:

$$\{ Y_{e_s}^p \mid e = \overline{1, L}; s = \overline{1, S_e}; p = \overline{1, P} \}$$

określających kolejność przetwarzania programów użytkowych.

Przy czym:

$$Y_{e_s}^p = \begin{cases} 1, & \text{jeżeli } t_{e_s} \in (t_{p-1}, t_p); \\ 0, & \text{w przeciwnym wypadku;} \end{cases}$$

A obliczane Z można wyrazić jako:

$$Z = \max_p \left[\frac{1}{T^p} \sum_{e=1}^L \sum_{s=1}^{S_e} Y_{e_s}^p t_e \right], \quad p = \overline{1, P},$$

gdzie: $[A]$ - część całkowita A.

Wtedy zadanie określenia minimalnej liczby procesorów jednakowej przepustowości, przetwarzających zadany zestaw programów użytkowych o znanych czasach ich przetwarzania, znanych momentach początkowych przetwarzania i ściśle określonych terminach zakończenia przetwarzania polega na określeniu kolejności ich przetwarzania:

$$Y = \{ Y_{e_s}^p \mid e = \overline{1, L}; s = \overline{1, S_e}; p = \overline{1, P} \}$$

i na określeniu takiego:

$$Z > 0$$

że:

$$Z = \min_{Y_{11}^1, \dots, Y_{L S_L}^P} \max_{p \in [0, P]} \left[\frac{1}{T^p} \sum_{e=1}^L \sum_{s=1}^{S_e} Y_{e_s}^p t_e \right] \quad (2)$$

Przy czym muszą być spełnione warunki:

$$D_{e_s} \geq Y_{e_s}^p p \geq d_{e_s}, \quad e = \overline{1, L}; s = \overline{1, S_e}; p = \overline{1, P}; \quad (3)$$

$$\sum_{p=d_{e_s}}^{D_{e_s}} Y_{e_s}^p = 1, \quad e = \overline{1, L}; s = \overline{1, S_e}, \quad (4)$$

Oznacza to, że terminy zakończenia S-tego kolejnego przetwarzania 1-tego programu, nie mogą

przekraczać w rozpatrywanym przedziale czasu, ściśle określonego (dyrektywnego) terminu, oraz że chwile rozpoczęcia przetwarzania programu nie mogą być wcześniejsze od założonych (warunek 3). Warunek (4) wyraża wymaganie, iż przetwarzanie każdego l_j -tego programu użytkowego musi rozpocząć się i zakończyć w określonych terminach.

Spełnienie warunku (1) umożliwia przejście od rozpatrywania systemu typu $n|Z$ do rozpatrywania systemu $n|1$ a więc posługiwanie się przy poszukiwaniu minimalnej koniecznej liczby procesorów, wynikami teorii kolejek dla systemu typu $n|1$ [3, 4].

Heurystyczny algorytm proponowany dla rozwiązania zadania (2) - (4) jest oparty na poszukiwaniu kolejki systemu $n|1$, pod warunkiem spełnienia wymagania obsługi minimalizującego niezbędną zasoby. W wyniku realizacji algorytmu jest określona niezbędna liczba procesorów zapewniająca przetwarzanie zgłoszonego zestawu programów użytkowych w ściśle określonych terminach. Następnie tworzy się kolejkę systemu $n|1$ o Z -krotnie zwiększonych przedziałach czasowych T^p , $p = \overline{1, P}$, przez uszeregowanie momentów rozpoczęcia przetwarzania programów użytkowych, tak aby były zachowane ściśle określone terminy zakończenia ich przetwarzania. Jeżeli dla danego Z udaje się utworzyć taką kolejkę, a dla $Z - 1$ nie, to utworzona kolejka jest rozwiązaniem zadania. W przeciwnym razie iteracyjny proces poszukiwania jest kontynuowany.

Algorytm daje przybliżone rozwiązanie zadań typu (2) - (4), a warunek (1) pozwala uznać, że otrzymane rozwiązanie jest dostatecznie dokładne.

Realizacja algorytmu przebiega według następujących kroków:

- 1/ początek;
- 2/ obliczanie żądanej minimalnej liczby Z procesorów, wg wzoru:

$$\underline{Z} = \sum_{e=1}^L t_e s_e / K_0 \sum_{p=1}^P T^p,$$

gdzie: K_0 żądany współczynnik wykorzystania procesora;

- 3/ $p = 1$;

- 4/ ze zbioru przetwarzanych programów użytkowych

$$Q = \{q_{e_s} | e = \overline{1, L}, s = \overline{1, s_e}\}$$

wydziela się podzbiór $Q^p \subset Q$ programów użytkowych, które mogą być przetworzone w p -tym przedziale czasowym:

$$Q^p = \{q_{e_s} | e = \overline{1, L}; s = \overline{1, s_e}; D_{e_s} \geq p \geq d_{e_s}; \sum_{p=d_{e_s}}^{D_{e_s}} Y_{e_s}^p = 0\};$$

- 5/ jeżeli:

$$\sum_{q_{e_s} \in Q^p} t_e \leq \underline{Z} T^p,$$

to:

$$Y_{e_s}^p = \dots = Y_{e_s}^{\underline{Z}} = 1,$$

oraz przechodzi się do (2)

6/ elementy Q^P porządkuje się według rosnących terminów zakończenia przetwarzania programów, otrzymując porządek:

$$q_{e's_1}, \dots, q_{e's_k};$$

7/ określa się takie r , że:

$$t_{e's_1} + \dots + t_{e's_r} \leq ZTP \text{ u } t_{e's_1} + \dots + t_{e's_{r+1}} > ZTP;$$

8/ jeżeli $D_{e's_{r+1}}$ to określa się zadanie dla

p - tego przedziału czasowego, $Y_{e's_1}^P = \dots = Y_{e's_r}^P = 1$, i przechodzi do p.12

9/ $Z = Z + 1$;

10/ $\Pi K = 1$;

11/ przechodzi się do 3);

12/ $p = p + 1$;

13/ jeżeli $p > P$ i $\Pi K = 1$, to przechodzi się do 17;

14/ jeżeli $p \leq P$, to przechodzi się do 4);

15/ $Z = Z - 1$;

16/ przechodzi się do 4);

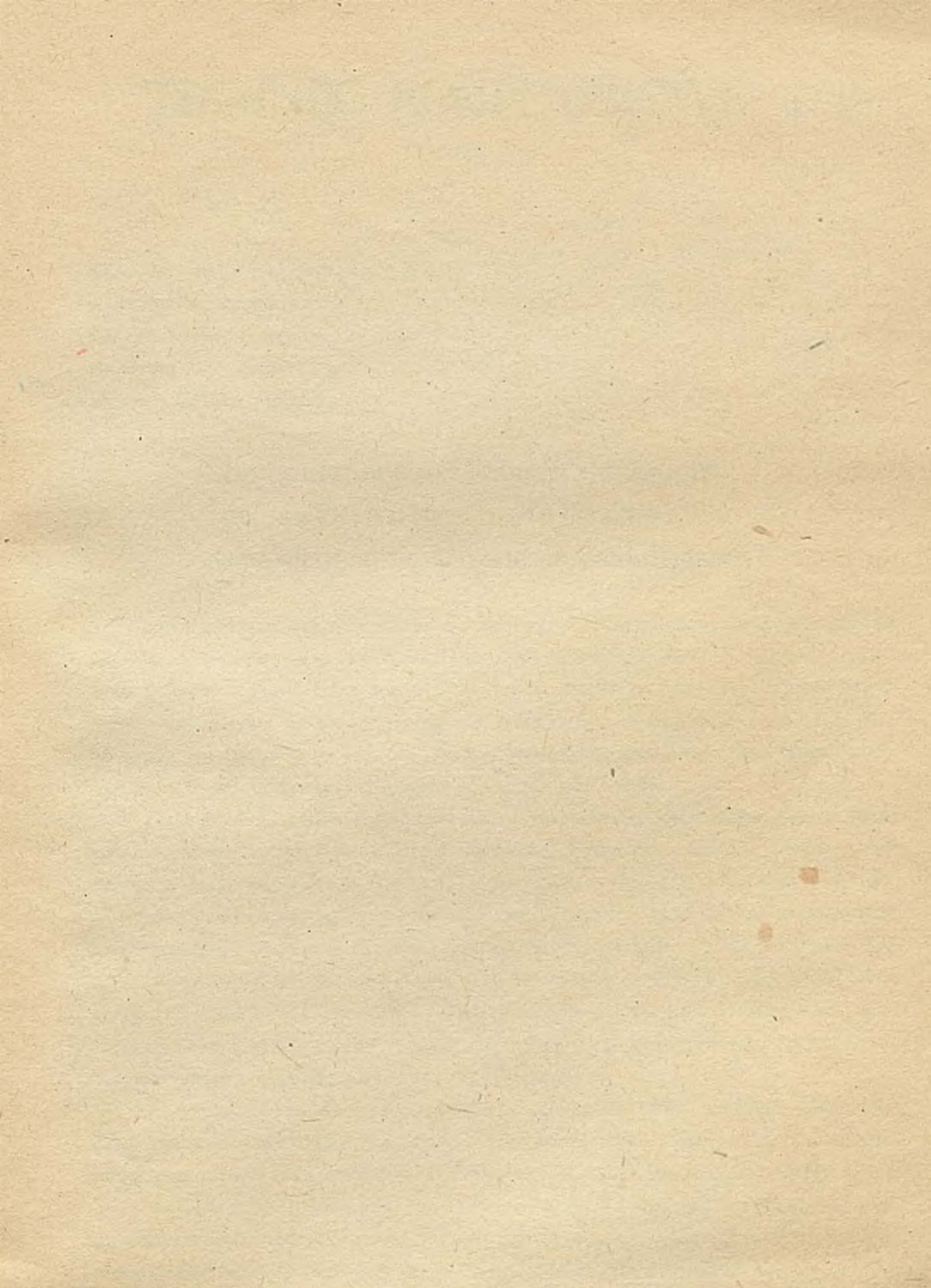
17/ koniec pracy.

Opisany algorytm został zrealizowany w języku PL-1. Dla różnych wariantów danych otrzymanie suboptymalnego rozwiązania trwało nie dłużej niż 60 minut czasu komputera JS 1035, przy liczbie zmiennych nie większej od 10 000.

Proponowany w niniejszej pracy model i algorytm mogą być użyteczne we wstępnym etapie projektowania systemu przetwarzającego programy użytkowe w trybie wsadowym.

Literatura

- [1] Барский А.Б.: Проектирование равнолеглых процессов обчислениowych. Машыностројение Москва 1980 s.192
- [2] Dagtiariow J.U.: Badanie operacji. Wyższa szkoła 1986 s.320
- [3] Konwiej R. W., Makswiell W. L., Miller L. W.: Teoria planowania. Nauka 1975 s.360
- [4] Tanajew W. S., Szkurba W.W.: Wstęp do teorii planowania. Nauka 1975 s.256



Kandydat nauk technicznych A.S. PIETROW

WNII Proftechobrazowanijs

Kandydat nauk technicznych A.G. PADAŁKO

Kandydat nauk technicznych A.F. SAKUN

Sibirskij Institut Metałurgii

Komputerowe urządzenia treningowe do szkolenia operatorów skomputeryzowanych urządzeń produkcyjnych

Jednym z ważniejszych kierunków postępu naukowo-technicznego jest automatyzacja przemysłu, oparta na technice mikroprocesorowej i na komputerach. Trzeba jednak zdawać sobie sprawę, że wprowadzanie do przemysłu nowoczesnych technologii wpływa w istotny sposób nie tylko na zapotrzebowanie na poszczególne grupy zawodowe pracowników, lecz również powoduje jakościowe zmiany charakteru i treści czynności wykonywanych przez robotników.

Funkcjonalna analiza charakteru pracy na poszczególnych stanowiskach pokazuje, że w warunkach wysoko zmechanizowanej i zautomatyzowanej produkcji dominujące stają się funkcje kontroli, diagnostyki, a nawet podejmowanie decyzji o charakterze zarządzającym. Wzrasta też znaczenie czynności organizacyjno-planistycznych, natomiast przeważająca większość, a często nawet wszystkie czynności wykonawcze lub technologiczne w dużej mierze przejmują maszyny, gniazda obróbcze, półautomatyczne lub automatyczne linie technologiczne, obrabiarki sterowane numerycznie itp. nowoczesne oprzyrządowanie przemysłowe. Zapewnia to automatyzację i ciągłość procesu technologicznego.

W nowoczesnym przemyśle w coraz większym stopniu następuje połączenie czynności intelektualnych i fizycznych, a w wyniku tego stopniowo zanika różnica między pracownikami fizycznymi i umysłowymi. Już obecnie tradycyjne prace umysłowe, takie jak planowanie, zarządzanie, kontrola itp. np. u ustawiacza linii automatycznych, zajmują ponad połowę czasu pracy (u wytapiacza, przy wytopie metali najwyższej jakości stali do 70%, a u operatorów rozdzielni automatycznego sterowania fabryką chemiczną - nawet do 90%) [1].

Wynika z tego, że we wszystkich złożonych procesach produkcyjnych sterowanie wykonuje jeden lub nieliczna grupa wykwalifikowanych specjalistów - operatorów.

W warunkach komputeryzacji procesów technologicznych czynności zarządzające czy też sterujące, wykonywane przez operatorów, w przeważającej większości są w programie komputerowym, natomiast operator musi znać zasady eksploatacji tych skomputeryzowanych urządzeń w trybie programowym pozwalającym na programowanie komputerów do celu wprowadzania odpowiednich korekt do pracy całego zautomatyzowanego oprzyrządowania. Konieczne jest więc radykalne podniesienie zawodowych umiejętności operatorów, co osiągnąć można tylko w warunkach właściwego procesu nauczania, tj. gdy w procesie nauczania będą szeroko stosowane właściwe urządzenia treningowe oparte na technice komputerowej.

W niniejszym artykule rozpatruje się techniczno-pedagogiczne aspekty budowy komputerowego systemu sterującego urządzeniem treningowym. Rozpatruje się też zagadnienia związane z algorytmem działania takiego systemu w zależności od typu kompletnego urządzenia treningowego.

Wspomniane operatorskie urządzenia treningowe na ogół zawierają w części sterującej komputer i urządzenia peryferyjne, umożliwiające wymianę informacji między uczniem a komputerem. Za pomocą tych środków technicznych imituje się, bardziej lub mniej dokładnie - w zależności od celów i etapów nauczania - rzeczywisty pulpit sterowniczy. W ich zestaw mogą wchodzić panele informacyjne z pracującymi przyrządami kontrolno-pomiarowymi i schematami poglądowymi oraz wskaźnikami cyfrowo-analogowymi, środki sygnalizacji dźwiękowej i świetlnej, urządzenia sterujące, jak też monitory ekranowe.

Zalety operatorskich urządzeń treningowych sterowanych komputerem pozwalają na modelowanie rzeczywistych procesów, upraszczając je lub komplikując w zależności od indywidualnych warunków i możliwości ucznia. Zmieniając natomiast parametry procesu nauczania, takie jak czas, ilość okazywanej do kontroli informacji, złożoność tej informacji itd. pozwalają modelować różne awaryjne sytuacje. Urządzenia takie pozwalają też kształtować różne umiejętności i nawyki profesjonalne, jak też kontrolować przebieg procesu nauczania oraz otrzymywać obiektywne i wiarygodne oceny jakości nawyków zawodowych uczniów.

Istniejące w ZSRR i za granicą koncepcje budowy urządzeń treningowych umożliwiają określenie uogólnionej struktury operatorskiego urządzenia treningowego zawierającego w swoim zestawie:

- system modelowania sytuacji (warunków) zewnętrznych,
- system modelowania dynamiki obiektu sterowania,
- system oceny rezultatów działalności ucznia,

- systemy analizy działań ucznia,
- system analizy psychicznego stanu ucznia,
- systemy realizacji zawodowych trybów nauczania,
- systemy kształtowania intelektualnych umiejętności i nawyków,
- systemy kształtowania nawyków zawodowej działalności kontrolnej.

Ogólnie rzecz biorąc, przy nauce za pomocą operatorskich urządzeń treningowych, na robocze stanowisko ucznia przekazuje się informacje o modelowanej sytuacji zewnętrznej, o stanie obiektu sterowanego, jak również informacje niezbędne dla samokontroli wykonywanej bezpośrednio, w trakcie nauki. Za pomocą systemów analizy działań i psychicznego stanu ucznia opracowuje się dane ilościowe charakteryzujące prawidłowość działalności ucznia i jego kondycję psychiczną, które to dane są kompleksowo opracowywane przez system oceny rezultatów działania ucznia.

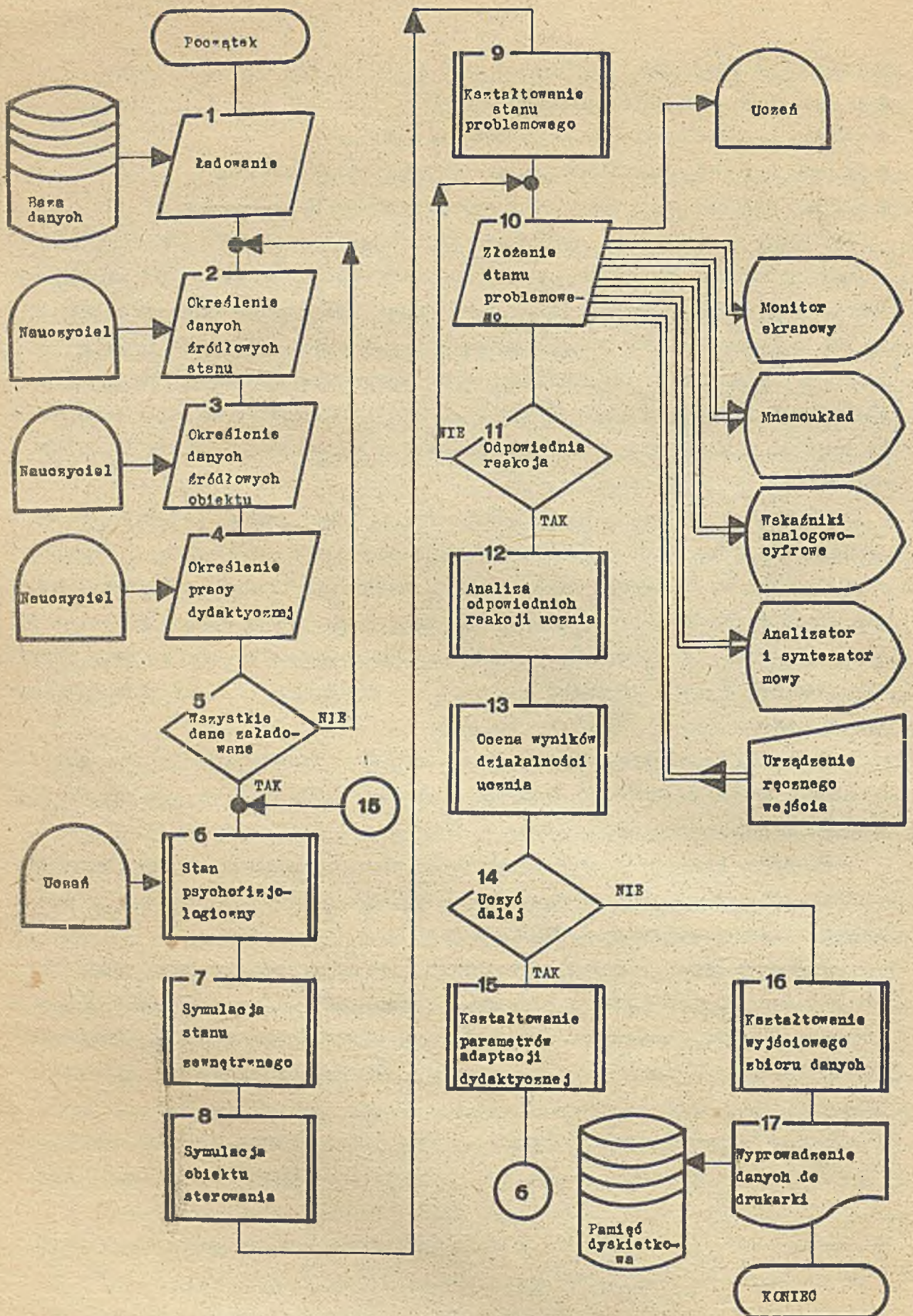
Zgodnie z wprowadzonymi danymi przy pomocy systemów intelektualnych realizuje się w żądanych reżimach nauczania kształcenie umiejętności i nawyków oraz kształtowanie nawyków zawodowych. Ponadto również poprzez wprowadzenie odpowiednich danych steruje się jego systemami modelowania warunków zewnętrznych i dynamiki obiektu sterowanego. Sterowanie to polega na wykonywaniu przez komputer odpowiednich programów użytkowych, realizujących właściwe modele matematyczne i właściwe algorytmy nauczania.

W wyniku uogólnienia podstawowych modeli matematycznych i algorytmów nauczania można sformułować uogólniony algorytm czyli strukturę logiczną, której schemat przedstawiono na rys. 1.

Pokazany algorytm powinien umożliwiać etapowe działanie urządzenia treningowego. Możliwe są trzy podstawowe reżimy (tryby) pracy urządzenia: przygotowawczy, treningowo-uczący i dokumentacyjny.

W czasie pracy w trybie przygotowawczym w pierwszym etapie nauczyciel lub instruktor, z pomocą bloku nr 1 (rys. 1) wprowadza do pamięci komputera programy modelujące, nauczające, diagnostyczne i kontrolujące, których podstawowymi (strukturalnymi) elementami są następujące procedury programowe, bloki i moduły:

- modelowanie zewnętrznego środowiska i obiektu sterowanego, jak też modelowanie dynamiki środowiska i dynamiki obiektu przy uwzględnieniu możliwości ucznia,
- zadanie i korekta warunków początkowych i wstępnych danych przeznaczonych do treningu konkretnego ucznia,



rys. 1. Schemat logiczny algorytmu sterowania treningiem

- formułowanie sytuacji problemowej; sytuacja problemowa jest przedstawiona uczniowi za pomocą urządzenia audiowizualnego, bloków przyrządowych, wskaźników analogowych i cyfrowych, podawana jest mu informacja o stanie i funkcjonowaniu modelowego obiektu oraz informacja o sterowaniu środowiskiem zewnętrznym,
- przedstawienie za pomocą właściwych indykatorów audiowizualnych sformułowanej sytuacji problemowej,
- określenie właściwych reakcji (odpowiedzi) ucznia na przedstawioną sytuację problemową, zgodnie z wymaganym znaczeniem semantycznym i liczbowa ocena otrzymanych rezultatów,
- identyfikacja psychofizjologicznej kondycji ucznia w momencie rozpoczynania treningu i w trakcie nauczania,
- adaptacyjne sterowanie procesem nauczania. Polega to na określeniu poziomu złożoności i ustaleniu kolejności przedstawiania sytuacji problemowych oraz nadawaniu tempa prowadzenia treningu itp. Wykonuje się to na podstawie rezultatów kontroli psychofizjologicznego stanu ucznia i osiągniętych przez niego rezultatów,
- dokumentowania na nośnikach magnetycznych i papierowych, rezultatów treningu, osiągniętego poziomu mistrzostwa zawodowego lub zaleceń dla danego ucznia, ze wskazaniem możliwych dróg i sposobów poprawy umiejętności zawodowych.

W drugim etapie trybu przygotowawczego, jeżeli to konieczne, nauczyciel, za pomocą bloku nr 2, wykorzystując klawiaturę lub pulpit komputera, koryguje początkowe dane dotyczące modelu warunków zewnętrznych; podobnie w etapie trzecim, za pomocą bloku nr 3, koryguje dane początkowe dotyczące modelu sterowanego obiektu.

W czwartym etapie nauczyciel posługując się blokiem nr 4 - ustawia reżimy pracy urządzenia treningowego np. kontrolę dopuszczenia do pracy na urządzeniu, nauczanie odbioru technologicznego lub zasady prowadzenia konkretnych procesów przemysłowych, ale może być to też zdawanie kolokwium kwalifikacyjnych lub tym podobne. Na tym tryb przygotowawczy pracy urządzenia treningowego kończy się i urządzenie, za pomocą bloku nr 5, zostaje przełączone na tryb treningowo-nauczający.

Współczesne psychologiczne koncepcje budowy adaptacyjnych, matematycznych modeli formowania, przedstawionych uczniowi sytuacji problemowych, przewidują uwzględnienie psychofizjologicznego stanu ucznia nie tylko na początku treningu, ale i w czasie jego trwania.

Dlatego praca ucznia z urządzeniem treningowym rozpoczyna się określeniem, za pomocą bloku nr 9, psychofizjologicznych danych ucznia. Pobranie tych danych może być wykonane znanymi w psychologii inżynierskiej metodami umożliwiającymi nie tylko pomiar poszczególnych parametrów, ale i wprowadzenie ich do kom-

putera. Mierzone są takie parametry człowieka, jak dane z elektrokardiogramu, częstotliwość pulsu, elektryczna reakcja skóry itp.

Blokiem nr 7 i 8 za pomocą odpowiednich procedur, na podstawie istniejących w systemie danych o przebiegu procesu treningu, o psychofizjologicznej kondycji ucznia w chwili rozpoczęcia nauki, o trybie nauki, oraz na podstawie danych o środowisku i o obiekcie sterowanym, formułuje się parametry modelu matematycznego. Model ten jest wykorzystywany następnie przez blok nr 9 dla generacji kolejnej sytuacji problemowej.

Blok nr 10 za pomocą właściwego zestawu przyborów audiowizualnych oraz wskaźników analogowych i cyfrowych, imituje model warunków zewnętrznych i obiektu sterowania. W ten sposób zapewnia się operatywne przedstawienie uczniowi sformułowanej w bloku nr 9 sytuacji problemowej. Sytuację tę przedstawia się wykorzystując różne efekty dźwiękowe i hasła słowne.

Następnie uczeń analizuje sformułowaną za pomocą bloku nr 10 sytuację przemysłową, ocenia stan i dynamikę przebiegu modelowanego procesu technologicznego i podejmuje decyzję o ewentualnej konieczności jego korekty. Wspomnianą korektę wykonuje uczeń odpowiednimi narzędziami sterowania procesem technologicznym. Przy czym za pomocą bloku nr 11 sterujące działania ucznia są przekazywane do bloku nr 12, gdzie są porównywane z ich normatywnymi wartościami.

Za pomocą bloku nr 13 realizuje się kompleksową, jakościową i ilościową ocenę pracy ucznia zarządzeniem treningowym w konkretnej sytuacji produkcyjnej. Przy czym na bieżąco powiadamia się ucznia o prawidłowych i błędnych jego działaniach.

Blok nr 14 przy osiągnięciu przez ucznia założonych dla treningu celów i zadań przeprowadza urządzenie treningowe do fazy dokumentacyjnej, inicjując przy tym bloki nr 16 i 17. W przeciwnym wypadku tzn. jeżeli uczeń nie osiągnie wymaganych wyników, do pracy włącza się blok nr 15, realizujący korekcję parametrów matematycznego modelu adaptacyjnego nauczania i początkowych danych tego modelu, oraz konkretnych danych imitujących warunki zewnętrzne i dane imitujące stan sterowanego obiektu.

Następnie cykl treningu powtarza się według wyżej opisanego algorytmu.

Blok nr 16 sporządza wykaz popełnionych przez ucznia błędów i formułuje praktyczne zalecenia w celu ich poprawienia. Ten wykaz wraz z końcową oceną, nowymi parametrami matematycznego modelu adaptacyjnego nauczania i danymi do modelu warunków zewnętrznych i do imitacji obiektu sterowanego, wyprowadza się na drukarkę i na nośnik magnetyczny.

Każda z wymienionych procedur programowych logicznego schematu strukturalnego (rys. 1), realizuje odrębny model matematyczny lub zbiór modeli matematycznych; nie są one jednak przedmiotem niniejszego artykułu.

Jako bazę techniczną kompletnego operatorskiego urządzenia treningowego, umożliwiającego w pełnym zakresie realizację przedstawionych na rys. 1 programowych procedur, do niedawna mógł być rozpatrywany tylko wyspecjalizowany sterująco-obliczeniowy system składający się z komputera średniej lub dużej mocy obliczeniowej, o szybkości nie mniejszej niż 500 tys. prostych operacji/s i o pojemności pamięci powyżej 500 KB. Takie komputery wymagają dla swojej pracy dużego zespołu specjalistów (operatorzy, programiści, elektronicy).

Duży koszt takiego komputera, złożoność obsługi oraz eksploatacji, niezależnie od dużych możliwości obliczeniowych tych komputerów powodują, że masowe ich stosowanie w operatorskich urządzeniach treningowych staje się przedsięwzięciem praktycznie nierealnym, natomiast ich szerokie możliwości stosowania komputerów w przygotowaniu kwalifikowanych specjalistów pojawiło się dopiero wraz z masową produkcją mikrokomputerów personalnych. Mikrokomputery przy praktycznie takich samych możliwościach obliczeniowych, jak dawniejsze średnie komputery, są dużo bardziej przydatne ze względu na takie parametry, jak ciężar, gabaryty, cena, energochłonność i możliwość eksploatacji bez specjalnego personelu technicznego. Jeśli do tego jeszcze dodać elastyczność i łatwe dostosowywanie mikrokomputerów do różnorodnych zadań występujących w zawodowo-technicznym przygotowaniu kwalifikowanych specjalistów, to łatwo można przedstawić szeroki wachlarz zastosowań personalnych mikrokomputerów do nauk.

W odróżnieniu od dużych komputerów mikrokomputery personalne (średniej lub większej mocy) mogą być łatwo wbudowane w dowolne systemy, obiekty, urządzenia czyli wszędzie tam, gdzie wymagane jest sterowanie operatywne.

Operatorskie urządzenia treningowe należy zaliczyć właśnie do tej klasy urządzeń z wbudowanym mikrokomputerem. Warto tu jednak zauważyć, że problem polega nie tylko i nie jedynie na ulepszeniu i modernizacji znanych już operatorskich urządzeń treningowych, ale i na opracowywaniu nowych oddzielnych urządzeń i całych ich systemów, bazujących na mikrokomputerach personalnych profesjonalnych.

Szczególną właściwością tych mikrokomputerów, niezbędną przy stosowaniu ich w wyspecjalizowanych systemach obliczeniowych operatorskich urządzeń treningowych jest istnienie wielu rodzajów kanałów wymiany informacji, umożliwiających przyjmowanie różnych sygnałów i sterowanie pracą różnych urządzeń zewnętrznych, bloków, węzłów, urządzeń wejścia informacji i wskaźników cyfrowo-analogowych.

W najwyższym stopniu tym wymaganiom odpowiada opracowany w ZSRR personalny profesjonalny mikrokomputer Iskra 1030M. Koncepcja budowy serii Iskra 1030, jest oparta na doświadczeniach zgromadzonych przy korzystaniu z elektronicznych maszyn buchalteryjnych i programowanych maszyn elektroniczno-klawiaturowych, opracowanych w ZSRR w końcu lat siedemdziesiątych i w początku osiemdziesiątych. Komputery tej klasy były przeznaczone do rozwiązywania zadań techniczno-inżynierskich, ekonomicznych i administracyjno-zarządczych oraz do badań naukowych. Było to podyktowane przewidywanym postępem technicznym w obszarze wytwarzania sprzętowych i programowych środków mikrokomputerów personalnych i perspektywnym programem rozwoju i udoskonalenia serii Iskra 1030.

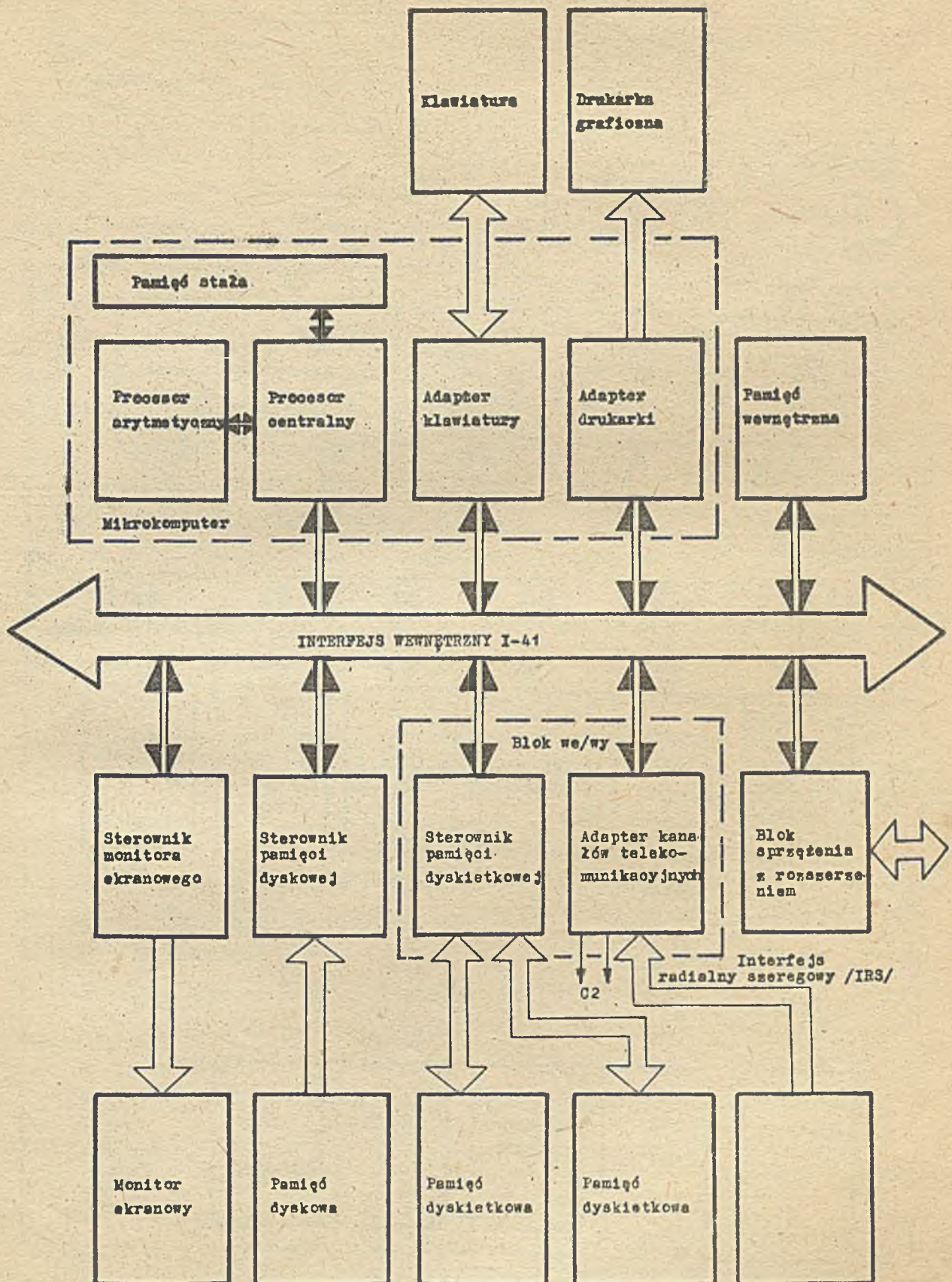
Opracowane w ZSRR specjalistyczne środki przetwarzania informacji dla Iskry 1030, pozwalają utworzyć dydaktyczny system zarządzania zespołem obliczeniowym, wyposażony w rozszerzające moduły, których schematy przedstawiono na rys. 2 i 3. Sam mikrokomputer Iskra 1030M jest wykonany w postaci zwartego modułu podstawowego o rozmiarach: 480x420x180 mm. Może być w nim rozmieszczone, oprócz bloku zasilania dwóch napędów dysków elastycznych do dyskietek 5.25" o pojemności 320 KB każdy i napędu dysku twardego, jeszcze sześć płyt (rys.2), stanowiących:

- mikrokomputer,
- pamięć operacyjną,
- kontroler kolorowego monitora graficznego,
- kontroler dysku,
- blok (płat) urządzeń we/wy,
- blok (płat) łączności z rozszerzeniami mikrokomputera Iskra 1030M.

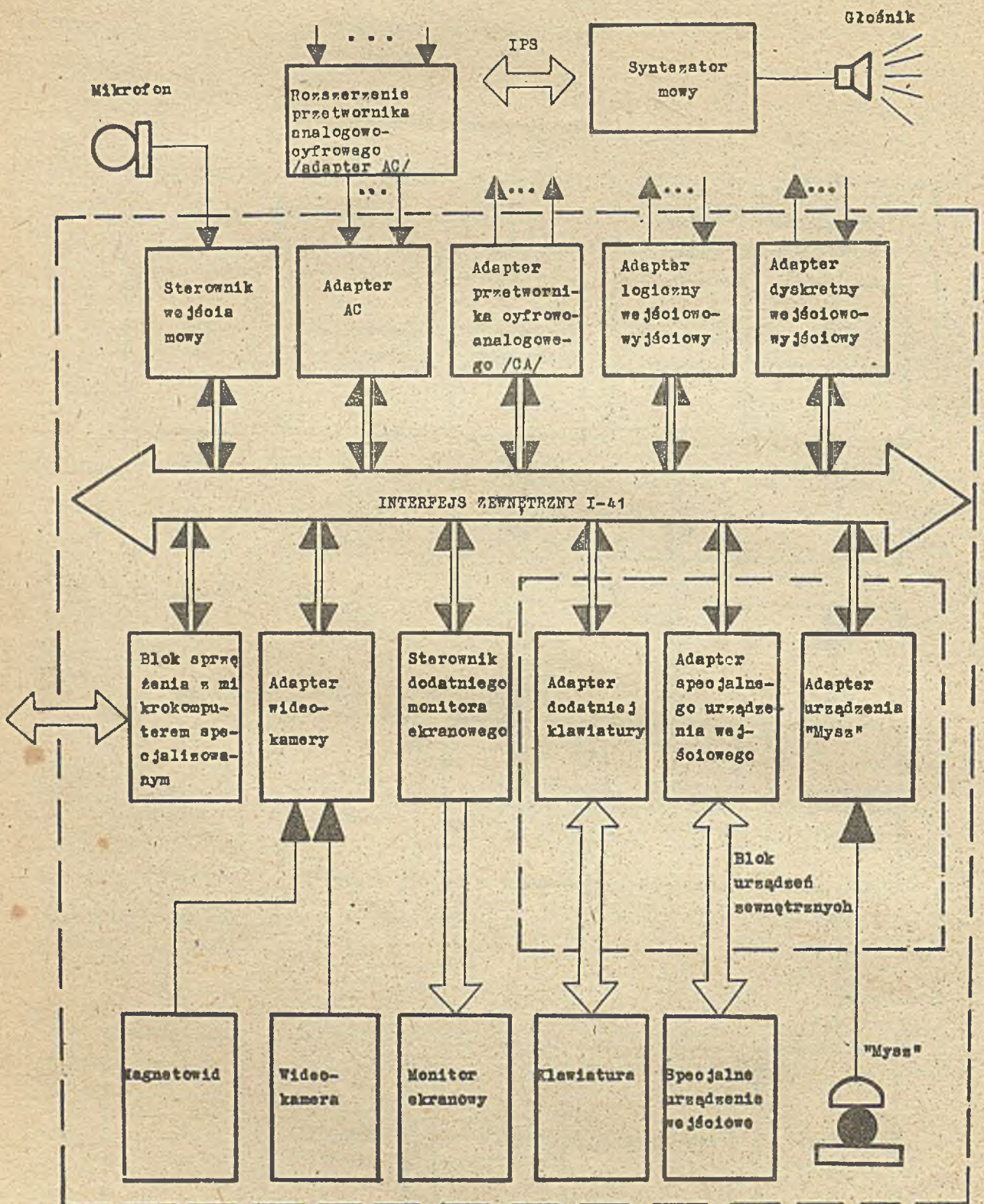
Konstrukcyjnie moduł rozszerzenia Iskry 1030M wykonany jest w wariantcie nastolnym, w tych samych rozmiarach co sam mikrokomputer z podłączeniem do wewnętrznego interfejsu I 41 za pomocą bloku łączności przez blok łączności rozszerzeniami (rys. 2 i 3).

Zgodnie z zasadami nauczania przyjmowanymi zazwyczaj w operatorskich urządzeniach treningowych, konieczny pełny zestaw wyspecjalizowanych środków przetwarzania informacji umieszczonych w module rozszerzenia, może być zrealizowany za pomocą następujących elementów:

- kontroler dodatkowego kolorowego monitora graficznego,
- adapter wideokamery,
- blok urządzeń zewnętrznych, zawierający w swoim zestawie adapter dodatkowej klawiatury, adapter wyspecjalizowanych urządzeń wejściowych (np. imitatory paneli sterujących mechanizmami, imitatory urządzeń załadowywania materiałów sypkich, imitator powrotu konwertera).



Rys. 2. Schemat logiczno-organizacyjny mikrokomputera specjalizowanego "I 3400"



Rys. 1. Schemat logiczny poszerzenia organizacyjnego mikrokomputera specjalizowanego "ISKRA 1030"

- imitator podnoszenia/opuszczania formy w przemyśle stalowym, imitatory obrabiarek sterowanych numerycznie, imitatory sterowane robotami przemysłowymi, manipulatorami itd.; do tego typu urządzeń zewnętrznych należy także adapter manipulatora typu "mysz",
- syntezytor mowy,
- kontroler wejścia mówionego,
- adapter przetwornika analogowo-cyfrowego (A/C),
- adapter przetwornika cyfrowo-analogowego (C/A),
- adapter we/wy sygnałów logicznych (Lwe/wy),
- adapter we/wy sygnałów dyskretnych (Dwe/wy).

Poniżej zamieszczono krótkie opisy wymienionych bloków mikrokomputera Iskra 1030M i środków przetwarzania informacji, jak również opisy funkcjonalnego ich wykorzystywania w operatorskim urządzeniu treningowym.

Podstawę mikrokomputera (rys.2) stanowi mikroprocesor KP1810 BM 86^{x/}, którego częstotliwość wynosi 8.7 MH, liczba poziomów - 8, pojemność pamięci operacyjnej 512 kB, pojemność pamięci stałej - 12 kB, możliwa liczba urządzeń bezpośredniego dostępu - do czterech. Do podstawowego modułu mikrokomputera podłącza się standardowy monitor graficzny o średnicy ekranu 31 cm i klawiaturę. Ponieważ klawiatura ta podłączona jest kablem elastycznym, może być ona w pewnym zakresie swobodnie przemieszczana.

Klawiatura zawiera następujące wydzielone pola: alfabetyczne, 10-cyfrowe, funkcjonalne - przy czym funkcje poszczególnych klawiszy określa użytkownik, oraz pole klawiszy sterujących kursorem na ekranie, wreszcie pole ściągania i ładowania systemu operacyjnego.

Naciśnięcie klawisza jest rejestrowane sygnałem dźwiękowym.

Adapter kanałów łączności umożliwia jednoczesną obsługę czterech kanałów łączności - dwa kanały na bazie interfejsu S2 i dwa kanały na bazie interfejsu IRPS.

Adapter wideokamery przeznaczony jest do wprowadzania w trybie cyfrowym do mikrokomputera Iskra 1030M standardowego obrazu telewizyjnego z magnetowidu lub z wideokamery. W tym wypadku objętość informacji graficznej jest nie mniejsza niż 256 - 256 punktów, przy 64 poziomach obrazu i długości ekspozycji nie więcej niż 5 s.

Kontroler wejścia mowy pozwala istotnie rozszerzyć dydaktyczne możliwości operatorskich urządzeń treningowych upraszczając procedurę współpracy ucznia z urządzeniem, co zwiększa efektywność treningu. Kontroler wejścia mowy (rys.3)

x/ - nazwy mikroprocesorów w brzmieniu oryginalnym

zbudowano na bazie procesora KI 801 BMIA o częstotliwości 4 MHz, pojemności pamięci operacyjnej 32 KB, przy wielkości rozpoznawanego przez urządzenie słownika do 100 słów, lub 56 kB przy wielkości słownika - 200 słów i pojemności pamięci stałej 4 kB dla słownika - 100 słów lub 8 kB dla słownika 200 słów. W skład kontrolera wejścia mowy wchodzi też preprocesor obróbki dźwiękowych sygnałów mowy. Czas reakcji na dźwiękowy sygnał mowy jest natychmiastowy. Niezawodność rozpoznawania słów nie mniejsza niż 96%.

Synteza mowy umożliwia wyjście w formie głosowej praktycznie w zakresie nieograniczonego słownictwa zgodnie z maszynowymi kodami obowiązującymi w mikrokomputerze Iskra 1030M.

Adapter logicznych sygnałów we/wy (Lwe/wy) jest przeznaczony do organizacji różnego rodzaju interfejsów i może być wykorzystywany do realizacji łączności mikrokomputera z przyrządami peryferyjnymi. Strukturalnie adapter Lwe/wy - jest realizowany na bazie trzech układów scalonych typu KP 580 BB 56 z elementami mocy KP 580 BA 86 na wyjściu. Adapter dyskretny sygnałów we/wy (Dwe/wy) umożliwia:

- przyjęcie sygnałów, nadchodzących np. z dyskretnych czujników urządzenia treningowego i wprowadzenie tych sygnałów do mikrokomputera,
- we/wy logicznych sygnałów poziomu TTL,
- wprowadzenie na mechanizmy wykonawcze urządzenia treningowego sygnałów sterowania dwupozycyjnego.

Adapter Dwe/wy realizuje dwubytową wymianę informacji z mikrokomputerem wspólnie z interfejsem I - 41. Przy liczbie kanałów wejścia dyskretnego - 8, maksymalne obciążenie prądowe wynosi - 0,2 mA, a przy liczbie kanałów wyjścia logicznego - 8 maksymalne obciążenie prądowe wynosi 15 mA. Natomiast liczba kanałów dyskretnego wyjścia - 8 wymaga dostępu dyskretnych sygnałów wyjścia - od 5 do 48 V, przy prądzie komutacji - do 200 mA.

Za pomocą adaptera przetwornika cyfrowo-analogowego - (C/A) może być realizowane wyprowadzanie na obiekty sterowania i na wskaźniki urządzenia treningowego sygnałów natężenia lub prądu w postaci odpowiednich funkcji schodkowych.

Wymiana informacji z mikrokomputerem jest realizowana w trybie bezpośredniego dostępu do pamięci komputera za pośrednictwem interfejsu I - 41. Formowanie sygnałów schodkowych realizuje się drogą cyfrowo-analogowego przetwarzania kodu informatycznego w natężenie, a następnie - przetwarzanie natężenia w prąd stabilizowany proporcjonalny do niego. Formowanie sygnałów czasowo ustalonych i zadawanie długości schodków - wyjściowych funkcji adaptera przetwornika C/A - realizuje się z wykorzystaniem schematów programowanego tajmera.

Adapter przetwornika C/A jest wyposażony w 16 kanałów wyjścia i pracuje według zasady wykorzystywania jednego programowo sterowanego przetwornika C/A, kolejno formułującego dla każdego kanału konieczny poziom sygnału analogowego, zapamiętywanego przez pamięć kanału analogowego.

Podtrzymywanie na wszystkich 16 kanałach wyjściowych zadanych przetwornikiem C/A poziomów sygnałów, realizuje się przez pracę adaptera przetwornika C/A w trybie regeneracji danych dla każdego kanału.

Adapter przetwornika C/A ma następujące charakterystyki techniczne:

- liczba kanałów wyjściowych - 16
- granice wyjściowego natężenia - ± 5 V
- rozszerzająca możliwość przestrojenia - 12 bitów, 11 bitów informacyjnych i jeden bit kontrolny
- czas ustalenia sygnału - poniżej 5 μ s
- możliwe obciążenie każdego kanału o wielkość prądu - nie mniej niż 20 mA.

Adapter przetwornika analogowo-cyfrowego (A/C) jest szybkim wielofunkcyjnym urządzeniem i umożliwia wymianę informacji z mikrokomputerem Iskra 1030M w trybie bezpośredniej wymiany informacji lub w trybie bezpośredniego dostępu do pamięci procesora i przekazywania danych zgodnie z interfejsem we/wy szyny systemowej. Za pomocą adaptera przetwornika A/C mogą być realizowane odczyty analogowych czujników urządzenia treningowego, przetwarzanie chwilowych wartości sygnałów natężenia w kod dwójkowy i przekazywanie tego kodu do mikrokomputera Iskra 1030M.

Współpraca adaptera przetwornika A/C z mikrokomputerem Iskra 1030M realizuje się zgodnie z Interfejsem szeregowym I-41 o strukturze magistralowej. W schemacie adaptera przetwornika A/C zastosowano następujące szyny interfejsu I-41:

- 16 bitów magistrali danych, przeznaczonych do zadawania trybów pracy adaptera i zliczania w mikrokomputerze rezultatów przetwarzania danych o stanie adaptera,
- dziewięć bitów magistrali adresowej, z których trzy pierwsze bity wykorzystuje się do kodowania i przekazywania do adaptera rozkazów interfejsowych, a sześć dalszych do kodowania fizycznego adresu urządzeń,
- pięć bitów magistrali sterowania, przy czym w czterech są zadawane do adaptera przetwornika A/C sygnały sterujące, a w ostatnim sygnale potwierdzenia gotowości adaptera przetwornika A/C.

Ponieważ w operatorskich urządzeniach treningowych, przy formułowaniu sytuacji problemowej i przy adaptacyjnym sterowaniu nauczaniem, wykorzystuje się sygnały i dane przychodzące z wielu czujników analogowych, urządzeń sterujących i urządzeń wprowadzania informacji takich, jak potencjometry, czujnik natężenia $\text{KHO}n\text{aKI}$ przyciski, przełączniki, to wydaje się celowe dokładniej opisać strukturę i funkcjonalne możliwości przetwornika analogowo-cyfrowego.

A oto ważniejsze jego charakterystyki:

- liczba bitów przekształcenia sygnałów - 12 (11 - informacyjnych i jeden kontrolny),
- maksymalna częstotliwość przetworzonego sygnału - 50 kHz,
- istnienie w strukturze adaptera A/C autonomicznego przetwornika C/A rozszerzających możliwość 12 bitowych sygnałów o tolerancji wyjściowych napiężeń $\pm 2,5 \text{ V}$, wskazany przetwornik ma wyjście na rozdzielacz użytkownika i może funkcjonować niezależnie od przetwornika A/C,
- istnienie w strukturze adaptera przetwornika A/C autonomicznego timera z trzema kanałami taktowania, umożliwiającego synchronizację procesów przetwarzania w szerokim przedziale czasowym okresów od 2 do 32000 μs , z zdolnością rozliczeniową dyskretyzacji - 0,5 μs ,
- kontrola wzorcowego i zerowego poziomu w celu zapewnienia cyfrowej korekty dokładnych parametrów przetwarzania,
- wyłączenie mikrokomputera po skończonym cyklu przetwarzania,
- dynamiczny zakres sygnałów wyjściowych: od $\pm 25 \text{ nV}$ do $\pm 10 \text{ V}$,
- istnienie trybu selekcji wejściowego sygnału analogowego na "gołą kość" realizowanego za pomocą przetwornika C/A i komparatora napięcia.

— Przy rozszerzeniu zadań dydaktycznych, przewidywanych dla tego lub innego urządzenia treningowego, mikrokomputer Iskra 1030M umożliwia dodatkowe dołączenie wyspecjalizowanych modułów. Dołączenie tych dodatkowych modułów opracowywania informacji powinno być wykonane przez wewnętrzny interfejs I-41 za pomocą standardowych interfejsów IRPS, styk S2 oraz interfejs narzędziowy IEEE - 488. W specjalnych sytuacjach dołączenie przyrządów nietypowych można wykonać za pomocą adaptera - LW we/wy. Jednak w tym ostatnim wypadku konieczne jest opracowanie własnego drajwera sterującego danym przyrządem.

Maksymalne oddalenie modułów wyspecjalizowanego urządzenia przetwarzania informacji i dodatkowych modułów określa się w zależności od parametrów użytego interfejsu, a więc dla I-41 wynosi ono - 2 m, dla IRPS - 500 m, dla interfejsu narzędziowego - 20 m, a dla styku -S2 zależy od typu używanego modemu.

Jednym z ważniejszych elementów wchodzących w skład dowolnego operatorskiego urządzenia treningowego jest jego oprogramowanie składające się z oprogramowania podstawowego, zwanego też systemowym, translatorów języków programowania, programów narzędziowych i użytkowych.

Podstawą oprogramowania systemowego operatorskiego urządzenia treningowego bazującego na mikrokomputerze Iskra 1030M jest modułowy system operacyjny ADOS przechowywany na dysku elastycznym (5 1/4"). System operacyjny ADOS składa się z zestawu modułów zmiennych i stałych.

Podstawowymi funkcjami i charakterystykami systemu ADOS są:

- zarządzanie zasobami systemu,
- język rozkazów,
- natychmiastowe lub grupowe wypełnianie rozkazów,
- programy wspomagające formatowanie i kopiowanie całych dyskietek i pojedynczych zbiorów,
- system zarządzania zbiorami, pośredniego i bezpośredniego dostępu,
- wywoływanie programów assemblerowych,
- wierszowy ekranowy redaktor tekstowy,
- automatyczne rozpoczynanie ładowania systemu.

Pod systemem operacyjnym ADOS pracują następujące języki programowania:

- Basic A - interpreter z operatorami graficznymi,
- mikroassembler (MAMS) z możliwością redagowania i uruchamiania programów,
- Pascal
- oraz język C.

Istnienie systemu operacyjnego ADOS i wypełnionych języków programowania, pozwala użytkownikom urządzenia treningowego stosować oprogramowanie opracowane dla mikrokomputerów z systemem operacyjnym MSDOS oraz opracowywać różne pakiety problemowo ukierunkowanych programów narzędziowych i użytkowych szerokiego zastosowania.

Oprogramowanie narzędziowe i użytkowe powinno być opracowywane w taki sposób, aby zapewniać użytkownikom maksymalny komfort nie tylko przy nauce za pomocą urządzenia treningowego, ale i na etapie opracowywania komputerowych programów dydaktycznych.

Szczególnie ważne jest aby nie wymagać od nich przy tej pracy specjalnych umiejętności, ani wiadomości wykraczających poza zakres ich zawodowych zainteresowań.

Literatura

1. Jaroszevszkaja M.W.: Personalny komputer "Iskra 1030M". Mikroprocessornyё sredstva i sistemy 1986 nr 4 s. 23-24.

dr Adam PAPST

Instytut Informatyki Akademii Ekonomicznej

Wrocław

System autorski AUTOOL

Wprowadzenie

Autoool jest systemem autorskim opartym na tutorialnym modelu nauczania /TIN/ przyjętym z systemu PLATO. Umożliwia on tworzenie materiału dydaktycznego z dowolnej dziedziny wiedzy bez znajomości programowania. W systemie AUTOOL zrealizowano zasadę rozdziału dwóch funkcji: tworzenie lekcji i ich wykonywanie. Oznacza to, że podczas tworzenia materiału dydaktycznego nie są formułowane przez autora algorytmy interpretacji materiału dydaktycznego. Gdyby program dydaktyczny był tworzony za pomocą normalnego języka programowania /np. Pascala/ to obie te funkcje musiałyby być w nim zawarte.

Rozdzielenie wymienionych funkcji ma następujące zalety:

- autor programu dydaktycznego nie musi umieć programować. W ten sposób bardzo istotnie poszerza się krąg potencjalnych autorów materiału dydaktycznego. Mogą się oni skoncentrować wyłącznie na tworzeniu części dydaktycznej lekcji abstrahując od algorytmów prezentacji i wykonywania materiału dydaktycznego.
- Autor programu dydaktycznego tworzy dane, które można o wiele łatwiej konserwować niż programy. Przy potrzebie aktualizacji należy tylko wymienić odpowiednie informacje o rekordzie zamiast poprawiać skomplikowany program. Ponadto łatwiejsza jest modyfikacja materiału dydaktycznego napisanego przez jednego autora przez innych autorów. Poza tym lekcja składająca się tylko z danych potrzebuje o wiele mniej pamięci niż odpowiadający jej zawartością dydaktyczną program napisany za pomocą jednego z języków programowania.
- Program wykonywany /Ekskutor/ może być użyty dla każdej lekcji z dowolną zawartością dydaktyczną. Koszty tego programu wykonawczego rozkładają się w ten sposób na setki lub tysiące lekcji.

Uważam, że posiadanie tej klasy systemu autorskiego jest niezbędnym warunkiem dla rozwoju zastosowań mikrokomputerów w dydaktyce w Polsce. Tylko za pomocą takiego narzędzia można w miarę szybko stworzyć wiele programów dydaktycznych przez nauczycieli odpowiednich typów szkół, czyli przez osoby najbardziej kompetentne do tworzenia materiału dydaktycznego.

Komponenty systemu AUTOOL

- AUTOOL - edytor lekcji. Umożliwia autorowi tworzenie nowych lekcji lub przeróbkę już istniejących. Jest to prosty w obsłudze program edytorski, za pomocą którego autor przygotowuje

materiał dydaktyczny składający się z tekstu, grafiki, koloru, przerw, pytań i prostych animacji. Podczas swojej pracy autor jest wspomagany przez system odpowiednimi funkcjami pomocniczymi. Opracowany materiał dydaktyczny jest zapisywany na dyskietce.

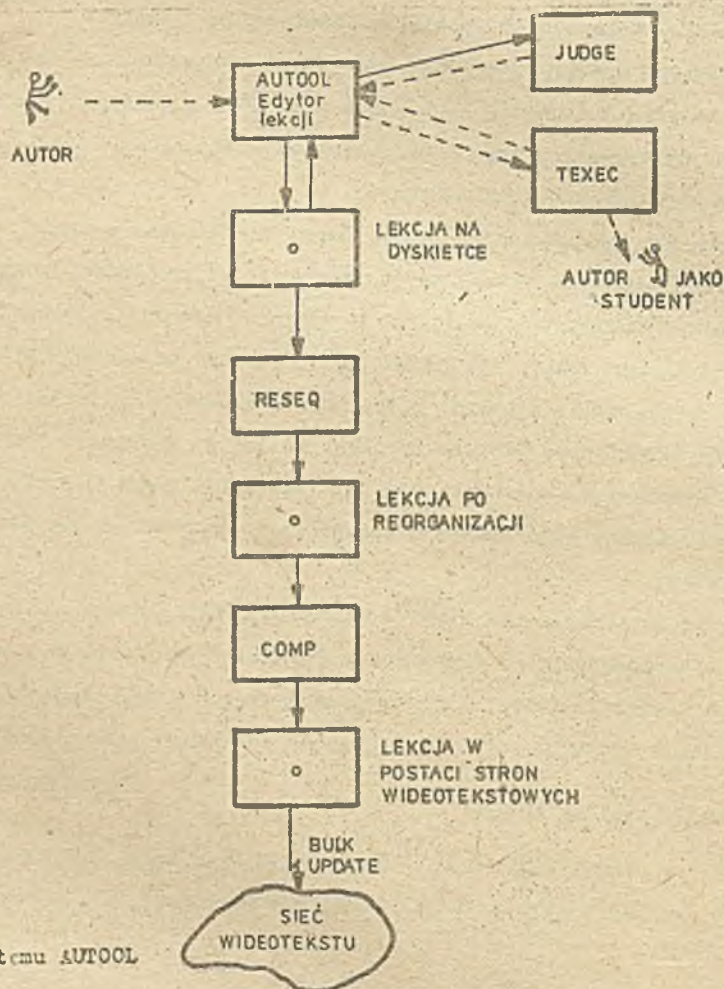
⊙ **TEEXEC** - program testujący. Stwarza autorowi możliwość przerabiania przygotowywanego materiału dydaktycznego z punktu widzenia uczącego się czyli testowanie materiału dydaktycznego. W trakcie pracy autor może dowolnie często wybierać między pracą w trybie edycji /AUTOOL/ i pracą w trybie testowania /TEEXEC/.

⊙ **COMP** - kompresor lekcji. Przygotowuje lekcje zapamiętane na dyskietce do ich rozpowszechniania za pomocą sieci wideotekstowej. Przede wszystkim program COMP wypełnia wymagania techniki wideotekstu zamieniając części lekcji na strony wideotekstowe, tak aby zminimalizować koszty przechowywania i czas ładowania lekcji.

⊙ **RESEQ** - reorganizator lekcji. Próbuje optymalizować kolejność zapamiętywania poszczególnych części lekcji, tak aby możliwie zgodna była logiczna i fizyczna kolejność ramek /frames/ tworzących jedną lekcję. Poza tym program RESEQ wykrywa "dziury" i pętle w lekcjach, których uczący się nie może opuścić.

⊙ **JUDGE** - program testujący odpowiedzi. Stwarza autorowi możliwość przetestowania zdefiniowanych przez niego lekcji pytań i sprawdzenia, jakie uczący uzyskałby odpowiedzi.

Na rys.1 przedstawiono zależności funkcjonalne między komponentami systemu AUTOOL.



Rys.1. Komponenty systemu AUTOOL

Lekcja w systemie AUTOOL

Każda lekcja w systemie AUTOOL składa się z pewnej liczby ramek /frames/. Ramka odpowiada dynamicznej zawartości jednego ekranu. Lekcja może składać się maksymalnie z 127 ramek, z których najwyżej 64 mogą być stworzone za pomocą systemu AUTOOL. Pozostałe ramki mogą być dowolnymi stronami wideotekstu. Do lekcji można dodatkowo dołączyć do 10 programów w języku Basic. Łącznie lekcja nie może przekraczać 32 bajtów.

Lekcja jest oznaczona przez nazwę kursu, która może być maksymalnie 10-znakowa, przy czym pierwszy znak nazwy musi być literą. Natomiast ramki w obrębie lekcji mają jednolicie skonstruowaną nazwę, która składa się z nazwy lekcji i dwuznakowego numerycznego przyrostka. System AUTOOL generuje te nazwy automatycznie podczas tworzenia lekcji poczynając od przyrostka 01. Każda lekcja ma też ramkę nagłówkową, która jest także generowana przez system i zawiera szczegółowe informacje o wszystkich ramkach danej lekcji.

W systemie AUTOOL można wyróżnić następujące typy ramek:

- N** - normalna ramka, która zawiera materiał dydaktyczny; liczba tych ramek w lekcji rośnie z ilością przekazywanej wiedzy;
- Z** - ramka podsumowująca, czyli ramka w przebiegu lekcji /nie konieczne ramka z najwyższym numerem/; zawiera ona podsumowanie i osiągnięte cele danej lekcji;
- I** - ramka indeksowa umożliwiająca autorowi rozgałęzienie w realizacji lekcji; wybór rozgałęzienia następuje przez podanie cyfry z przedziału 1-9 określającego liczbę punktów rozgałęzienia;
- P** - ramka zapytaniowa wyróżnia się tym, że zawiera co najmniej jedno pytanie; odpowiedź na zawarte pytania uzależnia dalszy przebieg nauki w ramach lekcji;
- G** - ramka graficzna, która może zawierać wszystkie dostępne w systemie AUTOOL typy obiektów z wyjątkiem dwóch, tj. indeksu i odpowiedzi. Służy ona głównie jako "tło", jeśli w danej lekcji powtarzają się podobne grafiki lub teksty. Ponieważ wiele ramek typu N może się odwoływać do tej samej ramki typu G, to wystarczy w wielu wypadkach raz stworzyć wielokrotnie stosowany tekst i grafikę i następnie wywołać go jako tło dla każdej z ramek. W ten sposób obniża się nakład pracy autora i jednocześnie zajętość pamięci danej lekcji, ponieważ ramka graficzna pomimo wielokrotnego stosowania jest tylko jednorazowo zapamiętywana. Każda ramka typu N może odwoływać tylko do jednej ramki typu G, która jest w trakcie przeróbki lekcji wyświetlana bezpośrednio przed ramką N;
- H** - ramka pomocy, która daje uczącemu się wskazówki w formie dodatkowych objaśnień. Podobnie jak ramce typu G, każdej ramce typu N można przyporządkować jedną ramkę typu H, z tym, że ramka typu H ukazywana jest uczącemu się tylko na jego życzenie, aby zapewnić mu pewną indywidualność w trakcie procesu nauczania. Po wyświetleniu ramki typu H następuje powrót do tego miejsca lekcji, w którym została ona przerwana;
- S** - strona wideotekstu. System AUTOOL umożliwia wkomponowanie w lekcję stron wideotekstu, które są traktowane identycznie jak strony typu G, czyli jako tło dla określonej ramki. W odróżnieniu od ramki typu G można jednej ramce typu N przyporządkować do 10 stron wideotekstu.

W trakcie przetwarzania lekcji uczący się prowadzony jest od ramki do ramki. W tym celu każda

da ramka typu N ma zdefiniowanego następcę i przewodnika. System AUTOOL standardowo przyjmuje, że przewodnikiem jest ramka o bezpośrednio niższym numerze a następcą ramka z bezpośrednio wyższym numerem. Struktura ta może być jednakże inaczej zdefiniowana przez autora, tak że posuwanie się w przód i w tył może być realizowane zgodnie z indywidualnymi życzeniami autora materiału dydaktycznego.

Po przerobieniu ramki typu N uczący odpowiada na pytanie, czy chce iść dalej czy powrócić do poprzedniej ramki; co umożliwia oczywiście powrót o kilka ramek do tyłu.

Dla ramek indeksowych muszą zostać zdefiniowane ramki odpowiadające początkom poszczególnych rozgałęzień, natomiast dla ramek zapytaniowych należy zdefiniować miejsca w lekcji, do których należy odesłać uczącego się po udzieleniu odpowiedzi poprawnej i po udzieleniu odpowiedzi błędnej. Natomiast ramki typu G nie mają zdefiniowanego ani poprzednika, ani następcy, bo nie są związane tylko z jedną ramką typu N, zaś ramka typu H ma tylko poprzednika. W tabeli 1 przedstawiono dopuszczalne połączenia pomiędzy poszczególnymi typami ramek.

Tabela 1. Dopuszczalne połączenia pomiędzy ramkami

Typ ramki	Następca	Poprzednik	Ramka H	Ramka G	Ramka B	Rozgałęzienie
N	tak	tak	tak	tak	tak	nie
I	nie	tak	tak	tak	tak	tak
F	nie	tak	tak	tak	tak	tak
G	nie	nie	nie	nie	nie	nie
H	nie	tak	nie	tak	tak	nie
B	nie	nie	nie	nie	nie	nie
Z	nie	tak	tak	tak	tak	nie

Dla lepszej prezentacji struktury lekcji w systemie AUTOOL posłużymy się przykładem zaczerpniętym z opracowania J. Garrat i in. Przykładowa lekcja składa się z 13 ramek; powiązania między tymi ramkami zawiera tabela 2, zaś strukturę sieciową tej lekcji przedstawia rys.2.

Przedstawiony przykład ma oczywiście bardzo uproszczoną strukturę, z reguły liczba ramek oscyluje wokół 64. Na ogół liczba rozgałęzień w lekcji wynosi około 8, natomiast w ramach każdego rozgałęzienia występuje zwykle kilka ramek. Opracowanie ogólnego schematu lekcji, takiego jak na rys.2, jest jednym z etapów prac nad stworzeniem lekcji. Schemat taki, wraz z projektami poszczególnych ramek, stanowi niejako projekt wstępny lekcji, który podczas procesu edycji może ulec pewnym zmianom spowodowanym uwarunkowaniami systemu AUTOOL.

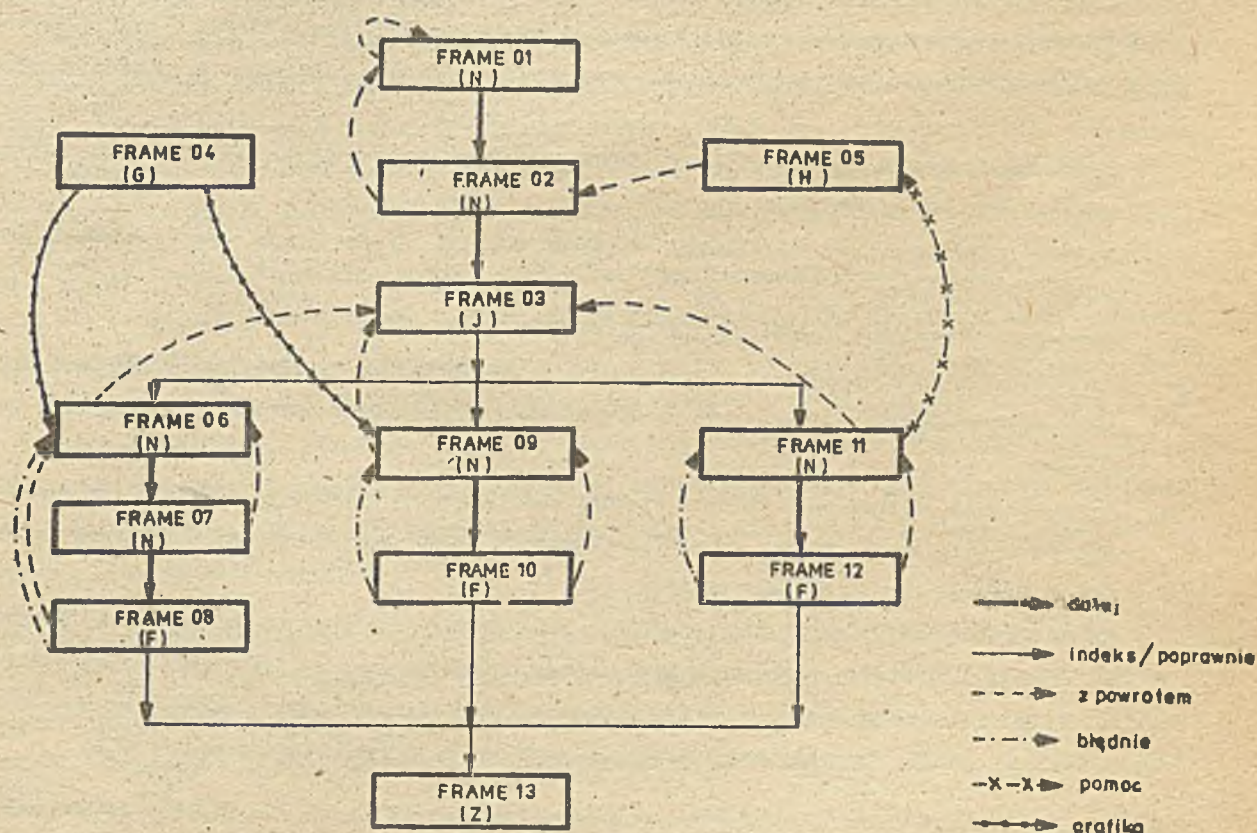
Obiekty tworzące ramkę w systemie AUTOOL

W systemie AUTOOL stosowane są następujące klasy obiektów:

graficzne	kasowanie
tekstowy	rozgałęzienie
"przerwa"	ruch
"kolor"	oceny odpowiedzi

Tabela 2. Powiązania między ramkami lekcji przykładowej

Numer ramki	Typ	Następca	Poprzednik	Ramka H	Ramka G	Rozgałęzienie
01	N	02	01	-	-	-
02	N	03	01	-	-	-
03	J	-	02	-	-	06,09,11
04	G	-	-	-	-	-
05	H	-	02	-	-	-
06	N	07	03	-	04	-
07	N	08	06	-	-	-
08	F	--	06	-	-	13 /poprawna odp✓ 06 /błędna odp✓
09	N	10	03	-	04	-
10	F	-	09	-	-	13 /poprawna odp✓ 09 /błędna odp✓
11	N	12	03	05	-	-
12	F	-	11	-	-	13 /poprawnie/ 11 /błędnie/
13	Z	-	-	-	-	-



Rys. 2. Struktura sieciowa lekcji przykładowej

program w języku BASIC"

strona widołektu"

DRCS" /zestaw znaków definiowanych dynamicznie/

Pozzczególnym klasom obiektów przyporządkowane są odpowiednie zestawy atrybutów, którym podczas definiowania ramki przydzielone są wartości. Większość obiektów jest pozycjonowana, tzn. podczas definiowania musi być określone miejsce obiektu na ekranie. Obiekty, które nie są pozycjonowane to kolor, DRCS, przerwa, strona widołektu i program w języku BASIC.

Liczba obiektów w ramce może wynosić maksymalnie 80, ponadto liczba punktów definiowania obiektów nie może przekroczyć wartości 256, zaś liczba wierszy tekstu wynosi maksymalnie 80. Ponadto obiekty definiowane w jednej ramce nie mogą zajmować więcej niż 1500 bajtów.

⊙ Obiekty graficzne można podzielić ogólnie na dwie grupy: liniowe i powierzchniowe. Do obiektów liniowych zalicza się linię prostą, łuk i krzywą. Natomiast obiekty powierzchniowe to prostokąt, koło, wycinek kołowy, odcinek kołowy i wielokąt. Oprócz tych dwóch grup obiektów graficznych stosowane są ponadto dwa typy obiektów graficznych: wektor i znacznik.

Obiekty powierzchniowe mogą być w różny sposób wypełniane i zakreskowane. Natomiast obiekty liniowe mają dwa rodzaje atrybutów: grubość linii i rodzaj linii. Wszystkie obiekty graficzne są tworzone bezpośrednio na ekranie w ten sposób, że autor podaje wartości i punkty definiowania za pomocą kursora graficznego. Liczba punktów definiowania jest różna w zależności od typu obiektu graficznego.

Jeśli obiekt składa się z więcej niż dwóch punktów definiowania to przebieg jego definiowania przedstawia się następująco:

- pierwszy punkt definiowania obiektu odpowiada pozycji kursora graficznego w momencie wybrania kodu obiektu,
- wprowadzenie wartości atrybutów dla obiektu,
- wszystkie punkty definiowania między drugim i przedostatnim są określane przez naciśnięcie dowolnego klawisza. Punkty te oznaczone są przez system w postaci małych punktów na ekranie,
- ostatni punkt definiowania obiektu określa się za pomocą kursora graficznego i wybrania specjalnego klawisza. System rysuje zdefiniowany obiekt z odpowiednimi atrybutami i w kolorze ostatnio zdefiniowanego obiektu kolor.

Część obiektów graficznych charakteryzuje się stałą liczbą punktów definiowania. W takim wypadku system AUTOOL zna poprawną liczbę punktów definiowania i rysuje obiekt nawet jeśli nie zostanie odpowiednio zakończone jego definiowanie. Również próba zbyt wczesnego zakończenia definiowania obiektu /za mało zdefiniowanych punktów/ jest ignorowana przez AUTOOL.

⊙ Obiekt tekstowy może być umieszczony w dowolnym miejscu ekranu /24 wiersze po 40 znaków/ i może składać się z więcej niż jednego wiersza tekstu. Dla obiektu tekstowego można zdefiniować następujące rodzaje atrybutów:

- wielkość znaków,
- repertuar znaków,
- różne sposoby mietotania,

podkreślanie części tekstu

różne kolory i warstwy kolorów w ramach części tekstu.

Po wybraniu kodu obiektu tekst, pojawia się na ekranie kursor znakowy i rozpoczyna się przebieg wprowadzenia obiektu tekstowego. Standardowe atrybuty obiektu tekstowego to normalna wielkość, brak podkreślenia, kolor znaków i ich tła, jak w ostatnio zdefiniowanym obiekcie kolor, migotanie wyłączone, normalny repertuar znaków. Te standardowe wartości atrybutów mogą być przez autora zmienione odpowiednio do potrzeb prezentacji materiału dydaktycznego.

● Obiekt "kolor" definiowany jest na trzech poziomach:

kolor pierwszoplanowy - kolor, którym pisany jest tekst i rysowana jest grafika; dopuszczalnych jest 16 kolorów;

kolor tła znaków - kolor, który jest tłem tylko dla znaków obiektu tekstowego. Można stosować osiem stonowanych kolorów;

kolor ekranu - kolor, którym wypełniony jest cały ekran. Tutaj także można stosować jeden z ośmiu stonowanych kolorów.

Ważność zdefiniowanych kolorów jest zachowana do momentu zdefiniowania nowego obiektu kolor lub do końca zdefiniowanej ramki.

Przy tworzeniu nowej ramki standardowe wartości kolorów są: biały kolor pierwszoplanowy, przezroczyste tło znaków i czarny kolor ekranu. Z reguły pierwszym obiektem ramki jest obiekt kolor, chyba że autor chce pracować w tonacji czarno-białej.

Autor może w dowolnym momencie zdefiniować obiekt kolor, z tym że zmiana jednego z poziomów koloru nie wymaga ponownego definiowania niezmienionych kolorów.

● Obiekt "przerwa" pozwala autorowi zdefiniować punkt czasowy w ramce, kiedy system powinien się zatrzymać podczas prezentacji materiału dydaktycznego. System AUTOOL ma dwie możliwości definiowania przerwy:

przerwa ograniczona czasowo /autor podaje jej długość w 1/100 częściach sekundy/.

przerwa użytkownika trwająca tak długo dopóki użytkownik naciśnie dowolny klawisz.

Stosowanie obiektu "przerwa" pozwala autorowi na stopniowe udostępnianie materiału dydaktycznego, a uczącemu się na indywidualne sterowanie tempem przerabiania materiału /jeśli autor stosuje przerwę użytkownika/.

● Obiekt "kasowania" pozwala autorowi na wymazywanie części ekranu podczas prezentacji lekcji. Pozwala to m.in. na kasowanie materiału dydaktycznego przed zadaniem pytania lub na umieszczenie w jednej ramce więcej informacji optycznej niż jest miejsca na ekranie monitora. Obiekt "kasowanie" wymazuje prostokątny obszar na ekranie, który definiowany jest przez podanie dwóch punktów.

● Obiekt "rozgałęzienie" pozwala autorowi na przekazanie informacji o umieszczeniu indeksu w odpowiednim miejscu na ekranie i jednocześnie podanie liczby rozgałęzień. Liczba możliwych rozgałęzień wynosi od dwóch do dziewięciu. W ramce może występować tylko jeden obiekt rozgałęzienia. Przed wprowadzeniem obiektu "rozgałęzienie" do ramki należy opisać na ekranie objaśnienie.

nia dla poszczególnych wartości rozgałęzień. Odwrótna kolejność może spowodować nałożenie się obiektu tekstowego objaśniającego poszczególne wartości rozgałęzień i obiektu rozgałęzienia. W wyniku takiego działania może ulec częściowemu skasowaniu lista możliwości wyboru dla przetwarzającego lekcję i tym samym utrudnić sterowanie wykonywania lekcji.

● Obiekt "ruch" umożliwia autorowi materiału dydaktycznego poruszanie tekstu i grafiki na ekranie. System AUTOOL daje autorowi do dyspozycji dwa sposoby definiowania drogi obiektu. W pierwszym wypadku autor wskazuje wszystkie punkty drogi, po której będzie się obiekt poruszał. W drugim wypadku autor podaje punkt początkowy i punkt końcowy oraz liczbę kroków, zaś system automatycznie dzieli drogę na jednakowe odcinki. Obydwa te sposoby definiowania drogi obiektu mogą być dowolnie kombinowane dla tego samego obiektu.

Niezależnie od wymienionych sposobów definiowania drogi obiektu system AUTOOL umożliwia stosowanie dwóch rodzajów ruchu obiektu:

wędrowanie, przy którym obiekt jest rysowany w każdym punkcie na drodze ruchu i po pewnym czasie kasowany, następnie jest rysowany w punkcie końcowym drogi ruchu obiektu i nie podlega kasowaniu,

kopiowanie, tym się różni od wędrowania, że obiekt jest rysowany w każdym punkcie drogi ruchu i nie jest kasowany. Za pomocą kopiowania można osiągnąć m.in. następujące efekty: dynamiczny wzrost, proste kopiowanie obiektów /np. tabela/, grafiki symetryczne /np. szpulę/.

● Obiekt oceny odpowiedzi służy do definiowania logiki odpowiedzi. AUTOOL umożliwia stosowanie dwóch typów odpowiedzi:

odpowiedź wyborem /do czterech możliwości odpowiedzi/,

odpowiedź tekstowa.

Obszar dla odpowiedzi na ekranie zajmuje sześć ostatnich wierszy, które dzielone są następująco:

pierwsze dwa są zarezerwowane na wprowadzenie odpowiedzi,

wiersz trzeci służy do wyprowadzania komunikatów systemowych,

ostatnie trzy wiersze przeznaczone są na formułowanie komentarzy po udzieleniu odpowiedzi na pytanie.

Czyli na formułowanie pytania w postaci tekstu i /lub grafiki autor dysponuje szesnastoma wierszami, co umożliwia dość dowolne formułowanie pytań.

W wypadku odpowiedzi na zasadzie wyboru jednej z czterech możliwości autor wprowadza najpierw pytania, następnie definiuje możliwości odpowiedzi, które są kolejno ponumerowane. Dla każdego z wariantów autor określa czy jest poprawny, czy nie, przy czym więcej niż jedna odpowiedź może być prawdziwa. Jeśli odpowiedź na pytanie ma postać dowolnego tekstu to może ona składać się z jednego lub kilku słów. Autor ma do dyspozycji 100 znaków na zdefiniowanie pożądanej odpowiedzi wzorcowej. Syntaktyka systemu AUTOOL składa się z różnego typu nawiasów, w których umieszczone są słowa obligatoryjne w odpowiedzi, synonimy, słowa dowolne, słowa dozwolone, słowa mogące wystąpić w dowolnej kolejności i słowa zabronione.

⊙ Obiekt "program w języku BASIC" pozwala autorowi na poszerzenie możliwości systemu AUTOOL o symulację przebiegów, którymi uczący się może sterować przez wprowadzenie danych oraz o realizację obliczeń wprowadzonych przez uczącego się na podstawie danych. Taki program może być większy niż 8000 bajtów, zaś liczba programów nie może być większa niż dziewięć. Należy także pamiętać, że lekcja wraz z dołączonymi programami nie może być większa niż 32 Kbajtów.

⊙ Obiekt "strona wideotekstu" umożliwia dołączenie do ramki do dziesięciu stron wideotekstu, które nie są tworzone za pomocą systemu AUTOOL, czyli pochodzą z systemu wideotekstu lub zostały specjalnie stworzone za pomocą edytora wideotekstu. Do drugiej ramki można dołączyć jeden obiekt "strona wideotekstu", który musi być pierwszym obiektem w ramce. W ten sposób autor lekcji może skorzystać z oferty informacyjnej zawartej w systemie wideotekstu i znacznie utrzymać materiał dydaktyczny.

⊙ Ostatni z obiektów, jaki może być użyty przez autora lekcji w systemie AUTOOL, to obiekt DRCS /Dynamical Redefinable Character Sets/, zawierający zestawy znaków dowolnie definiowanych przez użytkownika. Zestawy DRCS są wykorzystywane przez system AUTOOL, lecz tworzone są za pomocą innych narzędzi np. edytora CEPT mikrokomputera MUPID. System AUTOOL dopuszcza stosowanie do czterech zestawów znaków DRCS w jednej lekcji. Zestawy DRCS zawierają takie znaki, które nie mieszczą się w standardzie CEPT, np. znaki przegłosu, symbole matematyczne, znaki drogowe, symbole notacji technicznej.

Przedstawione elementy składowe ramek oraz typy ramek i możliwości ich powiązania wyraźnie charakteryzują możliwości prezentacji materiału dydaktycznego, jakie zawarte są w systemie autorskim AUTOOL. System AUTOOL jest bardzo interesującą propozycją umożliwiającą nauczycielom i innym autorom na w miarę proste opracowanie lekcji bez umiejętności programowania.

Obecnie autorzy systemu AUTOOL, Instytut Informatyki Uniwersytetu Technicznego w Grazu i firma Control Data Corporation, kończą pracę nad stworzeniem nowej wersji systemu dysponującego m.in. biblioteką obiektów graficznych, bardziej rozbudowaną logiką oceny odpowiedzi itp.

Literatura

- Garrat J., Huber F., Huemer H.: AUTOOL - ein Btx - orientiertes Autorensystem. Report 206. Technische Universität Graz 1985.
- Kaiser D., Maurer H.: AUTOOL - a new System for computer assisted instruction. Report 216. Technische Universität Graz 1986.
- Kaiser D., Maurer H.: How to develop a Costoc course. Report 229. Technische Universität Graz 1986.
- Makedon F., Maurer H., Ohmann Th.: Presentation Type CAI in Computer Science Education at University Level. Report 236. Technische Universität Graz 1987.
- Huber F., Maurer H.: Extended Ideas on Editors for Presentation Type CAI. Report 240. Technische Universität Graz 1987.
- Maurer H., Stubenrauch R.: A General Lesson Specification System. Report 241. Technische Universität Graz 1987.
- Papst A.: Projekt COSTOC - Wspomagane komputerem nauczanie informatyki na poziomie szkoły wyższej. Techniki Komputerowe 1988 nr 1.

Ludmiła Wasilewna LIBIZOVA
WNII Proftekhobrazowanija

Inteligentne komputerowe systemy wspomagające nauczanie

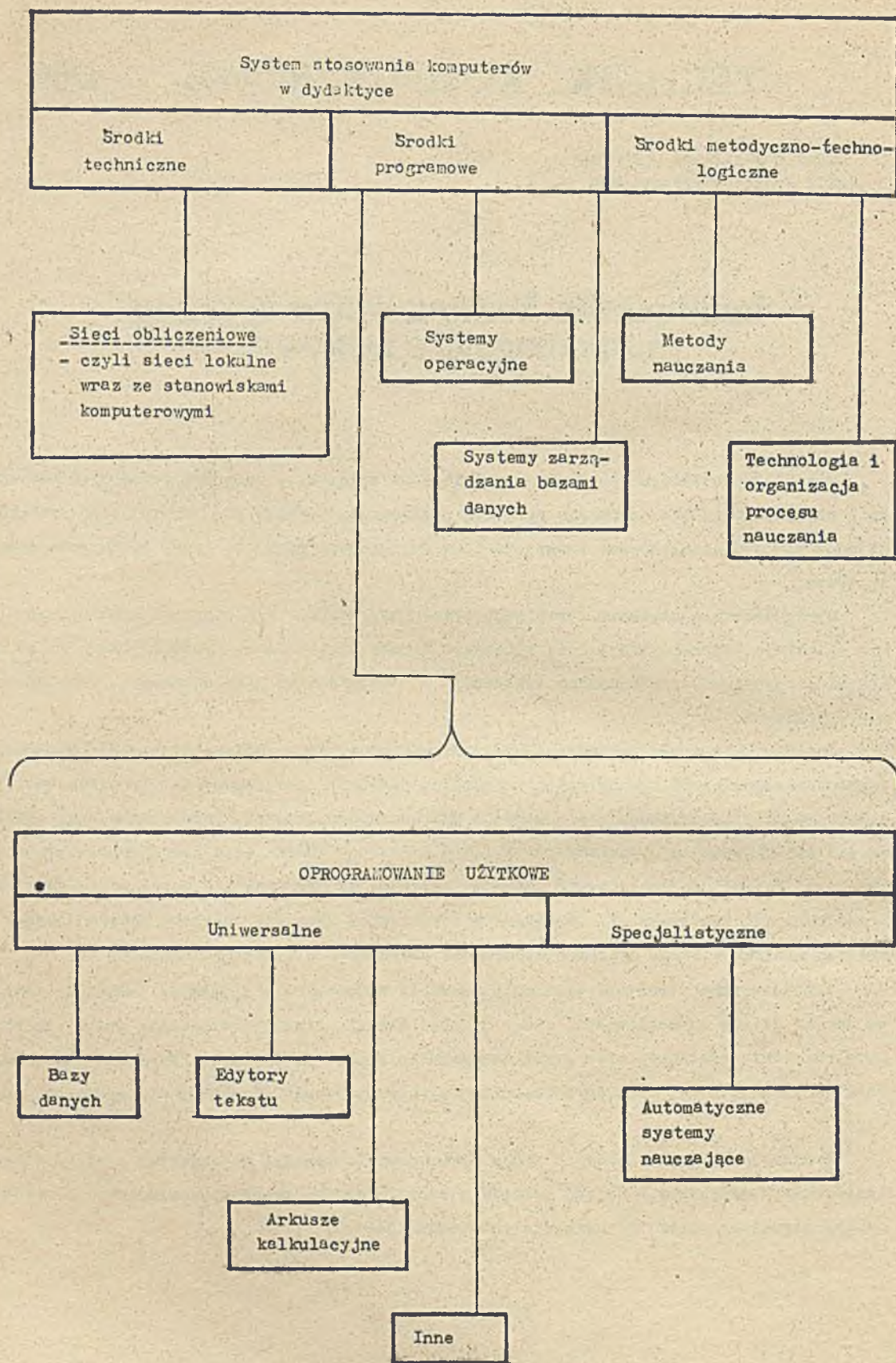
Rozwój komputerowych środków nauczania jest związany z rozwojem oprogramowania systemowego i użytkowego wykorzystywanego przy ich stosowaniu. Opracowaniem systemowego oprogramowania zajmują się wyspecjalizowani programiści, a dostarczane jest ono przez producenta wraz ze sprzętem.

Oprogramowanie użytkowe opracowuje często użytkownik, ale ciągle liczba opracowań daleko nie zaspakaja potrzeb. Radykalnie zwiększyć liczbę opracowanych programów mogą tylko specjalne środki narzędziowe, przeznaczone wyłącznie do wytwarzania ukierunkowanych problemowo programów użytkowych.

Obecnie więc przed profesjonalnymi programistami stoi zadanie dostarczenia programowych środków narzędziowych stanowiących integralny składnik oprogramowania. Oprogramowanie to pozwalałoby na pisanie użytkowych programów w wystarczającej ilości. Rozwiązanie tego problemu może okazać się kluczem do odpowiedzi na pytanie o sposoby zwiększenia ilości oprogramowania dydaktycznego. Pozwoli to opracowywać programy użytkowe szybko, więc i w odpowiednio dużej liczbie. Przed rozpoczęciem dyskusji o środkach narzędziowych istnieje potrzeba jednoznacznego określenia ich miejsca w całym systemie stosowania komputerów w dydaktyce /rys.1/.

Przedstawiona struktura systemu stosowania komputerów w dydaktyce pokazuje, że narzędziowe środki zależą od możliwości, które z kolei określane są przez wybrane środki techniczne i oprogramowanie systemowe oraz przez standardowe pomoce metodyczne. Związczą dwa pierwsze składniki nakładają podstawowe ograniczenia przy opracowywaniu struktury oprogramowania narzędziowego.

W niniejszej pracy dokonano próby sprecyzowania wymagań dotyczących systemu stosowania komputerów w dydaktyce i na tej podstawie zaproponowania pewnego podejścia do opracowania optymalnej struktury automatycznych systemów nauczających.



Wymagania stawiane programom dydaktycznym

Automatyczne systemy nauczające pojawiły się wprawdzie niedawno, ale w krótkim czasie zdobyły znaczenie jako programy narzędzowe. Stały się one konieczne jako narzędzie przy pisaniu dydaktycznych programów dla poszczególnych przedmiotów nauczania i dla specjalistycznych dyscyplin dydaktycznych na wszystkich poziomach nauczania, poczynając od szkoły podstawowej a na wyższych szkołach kończąc.

Wymagania co do struktury automatycznych systemów nauczających wynikają z wymagań dotyczących programów nauczania, które będą realizowane za pomocą tych systemów. Nas interesują tylko programy dydaktyczne. Przede wszystkim wypada rozpatrzyć jak wykorzystuje się komputer jako środek nauczania.

Przykładami zastosowań komputera w dydaktyce mogą być:

- demonstracja praw i zjawisk,
- modelowanie procesów,
- ćwiczenia w celu nabycia określonych umiejętności i nawyków,
- praca z użytkowymi programami przetwarzającymi wyrażenia arytmetyczne czy tekstowe, bloki informacyjne, wyniki pomiarów laboratoryjnych lub prac praktycznych itp.,
- kontrola wiedzy, umiejętności i nawyków,
- opracowanie wyników nauczania z uwzględnieniem uwarunkowań psychologicznych i poziomu przygotowania ucznia,
- wypracowanie strategii optymalnego nauczania indywidualnego,
- nauczanie samodzielne [1].

Konieczne jest przy tym aby programy dydaktyczne odpowiadały następującym zasadom:

- realizowały wychowawczy charakter nauczania,
- zapewniały odpowiedni poziom wiedzy,
- kształtowały świadomość i aktywność ucznia w procesie nauczania,
- zapewniały właściwe formy poglądowego prezentowania wiadomości,
- umożliwiały indywidualizację nauczania.

Dydaktyczne programy powinny również cechować się:

- komunikatywnością,
- ciekawą wizualizacją materiału,
- dość dużą dynamiką działania,
- naturalnością dialogu,
- prostotą obsługi,
- możliwością wykorzystywania programu, lub jego części w procesie nauczania.

Kończąc listę wymagań dotyczących programów dydaktycznych warto jeszcze wskazać dodatkowo na konieczność uwzględniania takich czynników jak:

- wynikowy poziom nauczania,
- formułowanie tylko takich celów, osiągnięcie których jest możliwe w trakcie pracy programem,
- zwiększenie motywów uczenia się,

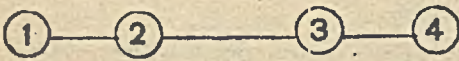
- właściwa organizacja pracy i czynności,
- wyłączenie elementów kontrolująco-korygujących [3].

Powyższe wymagania dotyczące programów dydaktycznych odnoszą się do merytorycznej części programów i form przedstawiania nauczanego materiału na ekranie monitora, a nie dotyczą typów algorytmów, które można wykorzystywać przy opracowywaniu dowolnego programu.

Typy algorytmów

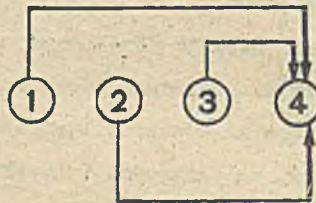
Jakie typy algorytmów są realizowane w niektórych programach opracowanych za pomocą automatycznego systemu nauczania? Jednym z najbardziej znanych automatycznych systemów nauczania jest system ASTRA/MIKRO i WUZ/MIKRO-1 pozwalający realizować przedstawione bliżej typy algorytmów.

1/ Algorytm kolejnego przygotowania

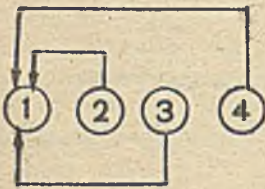


Uwaga: 1 ÷ 4 są to numery ukazujących się kolejno obrazów na ekranie monitora

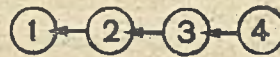
2/ Algorytm przygotowania równoległego



3/ Algorytm równoległego korygowania



4/ Algorytm kolejnego korygowania



5/ Algorytm przenoszenia



gdzie: A, B - są to poziomy programów, określające złożoność zadania.

6/ Algorytm analityczny



7/ Algorytm syntetyzujący; czyli łączący wszystkie przedstawione algorytmy

Analiza możliwości i wymagań

Przedstawiony dotychczas materiał, obejmujący wymagania dotyczące programów dydaktycznych i możliwe typy algorytmów, na podstawie których mogą być tworzone programy dydaktyczne, pozwala na sformułowanie odpowiedzi na pytanie: czy wszystkie wymagania można realizować za pomocą przedstawionych typów algorytmów?

Analiza problemu pokazała, że właściwie tak, przy jednym jednak warunku, mianowicie jeśli

zachowa się jednolity proces nauczania i jeśli rzecz jasna wszystkie programy dydaktyczne zostaną wykorzystane w działalności wykładowcy.

Sposób pracy i działalności u każdego wykładowcy jest indywidualny, zależny od jego osobowości i twórczych możliwości. Przy czym ze względu na istotę działalności dydaktycznych można rozróżnić:

- | | |
|---------------------------------|--|
| ① nauczanie programowe | ④ kierowanie wypełnianiem zadań |
| ② wybór zadań i wiadomości | ⑤ analiza wiadomości nauczzonego |
| ③ formowanie zadań i wiadomości | ⑥ przetwarzanie danych o pracy nauczanego [2]. |

Należy tu podkreślić, że podstawowe zadanie sprowadza się do zapewnienia jak najkrótszej drogi dojścia i przyswojenia sobie przez ucznia określonej wiedzy. Natomiast tak bardzo ważną cechą programu dydaktycznego, jaką jest naturalność dialogu z trudem można osiągnąć wykorzystując automatyczne systemy nauczające.

W znanych automatycznych systemach nauczających odpowiedzi ucznia są sztywno konstruowane, tzn. odpowiedź może być dana tylko w formie, którą proponuje komputer; jakakolwiek zmiana w konstrukcji odpowiedzi prowadzi do uznania jej za niepoprawną, co z punktu widzenia metodyki nauczania jest całkowicie niesłuszne. Jest to nie tylko bardzo niewygodne dla ucznia, ale upraszcza nadmiernie proces sprawdzania wiedzy, sprowadza ów proces do prymitywnych odpowiedzi na sformułowania typu "wybierz poprawny numer odpowiedzi", "odpowiedz poprzez określony wybór słów" itd. Niezbędne jest stworzenie takiej struktury automatycznych systemów nauczających, która pozwalałaby na swobodne konstruowanie odpowiedzi. Istnieje więc konieczność opracowania programowych środków pozwalających prowadzić "swobodny" dialog z komputerem. W tym celu powinien być stworzony logiczny moduł analizy odpowiedzi udzielanej przez ucznia. Proponuje się do tego celu wykorzystać pewne koncepcje opracowane dla systemów ekspertowych, odpowiednio adaptując je przy tworzeniu systemów nauczających nowego typu. Byłby to inteligentny system samodoskonalący i samouczący, mogący gromadzić doświadczenia uczących się.

Etapy opracowania inteligentnego komputerowego systemu wspomagającego nauczanie

We wstępnej fazie badań nad inteligentnymi komputerowymi systemami wspomagającymi nauczanie wyróżniono następujące etapy opracowywania takich systemów:

1. Klasyfikacja rodzajów gromadzonej informacji /ze względu na objętość przechowywanej informacji, złożoność algorytmów jej przetwarzania itp. /
2. Podział funkcji, przypisywanych poszczególnym modułom systemu zgodnie z zasadami dekompozycji struktury /wykorzystywany na tym etapie aparat formalny, jest zgodny z zasadami schematów logicznych algorytmów/. W ramach tego etapu należy wykonać:
 - 2.1. ocenę trzech możliwych sposobów tworzenia oprogramowania informacyjnego /indywidualne opracowanie, wykorzystanie jakiegoś uniwersalnego systemu np. zarządzania bazami danych, zautomatyzowane konstruowanie specjalizowanego systemu zarządzania informacjami /bazy wiedzy/, itp./,
 - 2.2. analizę architektury systemu, a w jej ramach stworzyć:
 - 2.2.1. informacyjny model obiektu, oraz przeprowadzić
 - 2.2.2. integrację podstruktur danych,

- 2.3 model funkcjonalny całego systemu,
- 2.4 model struktury danych, obejmujący model
 - 2.4.1 reprezentacji danych w wewnętrznej pamięci komputera,
 - 2.4.2 reprezentacji danych w zewnętrznej pamięci komputera.
- 3. Model danej dziedziny przedmiotowej, obejmuje
 - 3.1 formalny model sterowania pracą programu,
 - 3.2 koncepcję oceniania przez ekspertów opracowywanej struktury programu użytkowego.
- 4. Szczegółowe badania wzajemnego oddziaływania tworzonego systemu i środowiska zewnętrznego, czyli zachowanie się użytkowników systemu, w trakcie pracy z systemem.
- 5. Analiza poszczególnych ważniejszych charakterystyk systemu i bardziej złożonych problemów organizacji systemu.
- 6. Językowe środki formalizacji wiedzy.
- 7. Forma przedstawienia wiedzy.
- 8. Struktura inteligentnego komputerowego systemu wspomagania nauczania.
- 9. Wybór technicznych środków do realizacji wybranej struktury.

Przedstawiona kolejność opracowania inteligentnego komputerowego systemu wspomagającego nauczanie nie pretenduje do najlepszej w pełnym tego słowa znaczeniu, pozwoli jednak chociaż zbliżyć się w pewnym stopniu do optymalnej struktury takiego systemu.

Literatura

- [1] Golib M.: Struktura AOC. Informatyka i Kształcenie 1987 nr 3 s. 11-19.
- [2] Krótkie metodyczne rekomendacje dotyczące opracowywania programów nauczających i AOC. Materiały wewnętrzne WNII Proftechobrazowanie 1981.
- [3] Kuzniecowa A., Sergiejewa I.: Program nauczający i dydaktyka. Informatyka i Kształcenie 1986 nr 2 s. 87-90.
- [4] Maszbiec E.: Komputeryzacja nauczania: problemy i perspektywy. Moskwa 1986.

Margarita Borisowna LIFBIEDIEWA

WNII Proftechobrazowania

Leningrad

Dokumentacja oprogramowania dydaktycznego - wymagania ogólne

Wprowadzenie

Ogólne wymagania dotyczące dokumentacji oprogramowania dydaktycznego /dalej określane-go OD/ zostały opracowane zgodnie z roboczym planem problemu 1.2.7 KPPNT. W wymaganiach tych określono zasady przygotowywania, opracowywania i przyjmowania dokumentacji OD. Zbiór tych wymagań został przedstawiony i przyjęty do eksperymentalnego stosowania na roboczym spotkaniu przedstawicieli organizacji realizujących tematykę 1.2.7 we wrześniu 1987 r. w Warnie. Przy opracowywaniu tych wymagań korzystano z tzw. standardów RWPG: ST RWPG 1626-79, 2088-80, 1627-79, 3746-82, 2092-80, 2089-80.

Ogólne założenia

Do zbioru dokumentacji OD zalicza się: zadania techniczne, scenariusz pedagogiczny wraz z algorytmem nauczania, instrukcję programisty, wskazówki metodyczne dla nauczyciela - użytkownika danego OD, instrukcję dla ucznia - użytkownika danego OD, tekst danego OD i jego opis. Zestawienie zawartości tych dokumentów zawiera tabela 1. Wymienione dokumenty opracowywane są w różnych fazach procesu wytwarzania konkretnego OD. Zestawienie tych faz z odpowiadającymi im dokumentami zawiera tabela 2.

Tab.1. Zawartość poszczególnych dokumentów OD

Lp.	Nazwa i symbol	Zawartość dokumentacji
1	Zadania techniczne TZ	Określenie zakresu stosowania OD, techniczne i pedagogiczne wymagania, etapy, terminy opracowania danego OD
2	Scenariusz pedagogiczny PS	Opis dialogu ucznia z komputerem, wzory poszczególnych kadrów, krótkie zalecenia do pracy z programem
3	Algorytm nauczania AO	Omówienie zasad przechodzenia przez poszczególne kadry

4	Instrukcja programisty IP	Opis podstawowych wymagań dla procesu programowania
5	Wskazówki metodyczne dla nauczyciela MN	Opis metodyki wykorzystywania opracowanego OD na lekcji, łącznie z innymi metodami i środkami /pomocami/
6	Instrukcja ucznia IU	Opis podstawowych zasad pracy ucznia z danym OD
7	Tekst OD T OD	Wydruk tekstu danego OD w języku programowania opatrzone komentarzem
8	Opis OD O OD	Opis funkcjonalnego przeznaczenia struktury logicznej /np. schemat blokowy/ opis danych i wyników

Tab.2. Zestawienie faz i etapów wytwarzania OD oraz poszczególnych dokumentów

Lp.	Faza wytwarzania	Etap wytwarzania	Dokument
1	Opracowanie zadania technicznego	Stwierdzenie konieczności wytworzenia danego OD	TZ
		Opracowanie i zatwierdzenie zadania technicznego	
2	Realizacja projektu technicznego	Opracowanie scenariusza pedagogicznego	PS
		Opracowanie algorytmu nauczania	
		Opracowanie instrukcji programisty	
3	Realizacja projektu roboczego	Opracowanie OD	T OD
			O OD
4	Eksperymentalne aprobowanie i wdrażanie	Napisanie wskazówek metodycznych dla nauczyciela	MN
		Przygotowanie instrukcji dla ucznia	IU
		aprobowanie	Akt wynikowy
		wdrożenie	Akt wdrożenia

Wszystkie wymienione dokumenty związane z danym OD opracowuje się według ogólnie przyjętych zasad, przy czym w zależności od potrzeb dopuszcza się pewne modyfikacje. Na konieczność tych modyfikacji należy zwrócić uwagę w zadaniu technicznym. Jednak nawet w zmodyfikowanych postaciach poszczególnych dokumentów powinny zostać wprowadzone te elementy, które uznano za niezbędne. Szczególną uwagę należy przykładać do dokumentów przeznaczonych dla

użytkowników /metodyczne wskazówki dla nauczyciela i instrukcja ucznia/.

Opracowanie dokumentacji OD

Każdy dokument związany z OD składa się z następujących stałych części: część tytułowa, część informacyjna, wykaz zmian. Część tytułowa zawiera stronę zatwierdzenia i stronę tytułową. Strona zatwierdzenia /przy opracowywaniu dokumentów w ramach KPNT/ zawiera:

- nazwę problemu KPNT - tzw. szyfr /cyfrowy/, nazwę i szyfr zadania według szczegółowego planu współpracy,
- stanowiska i podpisy osób uzgadniających i zatwierdzających dokument,
- pełną nazwę danego OD, przedmiotu nauczania i tematu, nazwę danego dokumentu,
- przeznaczenie dokumentu i jego numer,
- rok opracowania dokumentu,
- symbol dokumentu.

Strona tytułowa zawiera te same informacje, z tą tylko różnicą, że zamiast podpisów osób uzgadniających i zatwierdzających umieszcza się przez całą szerokość strony słowo ZATWIERDZONO.

Ponadto na stronie tytułowej podaje się liczbę stron danego dokumentu.

Część informacyjna składa się ze streszczenia i właściwej zawartości. W streszczeniu należy podać informacje o przeznaczeniu dokumentu i krótkie sformułowanie jego podstawowych części. W zawartości dokumentu określa się strukturę dokumentu. Właściwa zawartość dokumentu może dzielić się na rozdziały w zależności od specyfiki dokumentu. Wykaz zmian zawiera notatki o zasadniczych zmianach wprowadzonych do dokumentu.

Ogólne zasady wprowadzania zmian

Zmianą wprowadzoną do dokumentu jest każde wprowadzenie do niego nowej informacji i każde usunięcie z niego dawnej informacji.

Przyczynami wprowadzania zmian mogą być:

- usunięcie ujawnionych błędów w samym OD lub w dokumencie,
- rozbudowa lub ulepszenie OD.

Zmiany wprowadza się oddzielnie do każdego z istniejących dokumentów. Każdą zmianę odnotowuje się w wykazie zmian według wzoru 1.

Wzór 1

WYKAZ ZMIAN										
N ^o	Numery stron				Ogólna liczba stron dokumentu	Numer dokumentu	Nowy numer dokumentu	Nowy numer dokumentu i data	Podpis	Data
	zmienionych	zamienionych	nowych	anulowanych						

Wszystkie zmiany uzgadnia się z opracowującymi /korzystającymi/ z danego dokumentu.

Zawartość i tryb opracowania TZ

Zadanie techniczne zalicza się do tych dokumentów, w których określa się przeznaczenie i zakres zastosowania oraz techniczne i pedagogiczne wymagania opracowanego OD, tu również ustala się fazy procesu wytwarzania danego OD.

Opracowanie TZ jest pierwszą fazą procesu wytwarzania OD. Zadanie techniczne opracowuje się na podstawie wstępnych wymagań dotyczących danych OD i jest podstawowym dokumentem określającym cały przebieg procesu wytwarzania oprogramowania.

Właściwa zawartość TZ powinna więc składać się z następujących rozdziałów:

- nazwa i zakres zastosowania danego OD,
- podstawa opracowania,
- przeznaczenie prac,
- wymagania techniczne i pedagogiczne,
- fazy procesu wytwarzania danego OD,
- sposób sprawdzania i odbioru.

W rozdziale "Nazwa i zakres zastosowania" podaje się nazwę przedmiotu zastosowania, rozdział i temat oraz określenie zawodu, dla którego opracowuje się dane OD. Ponadto podaje się zakres jego stosowania oraz ogólną charakterystykę sprzętu komputerowego, na którym dane OD będzie działało.

W rozdziale "Podstawa opracowania" podaje się pełną nazwę dokumentów, na podstawie których opracowuje się dane OD.

W rozdziale "Przeznaczenie pracy" określa się cel stosowania danego OD w procesie nauczania, jego specyfikę funkcjonalną i warunki eksploatacji - użytkowania. Przytacza się istniejące analogiczne produkty programowe lub stwierdza się, że takie OD jest opracowywane po raz pierwszy.

W rozdziale "Wymagania techniczne i pedagogiczne" określa się:

- wymagania techniczne dla danego OD i niezbędne eksploatacyjne charakterystyki sprzętu komputerowego,
- wymagania pedagogiczne /dydaktyczne/ stawiane opracowywanemu OD zgodnie z celami i zadaniami nauczania,
- wymagania metodyczne stawiane danemu OD wynikające z racjonalnej metodyki stosowania danego OD na lekcji, z organizacji pracy nauczyciela i ucznia.

W rozdziale "Sposób sprawdzania i odbioru" wymienia się dokumenty podlegające uzgadnianiu i zatwierdzaniu na poszczególnych etapach procesu wytwarzania OD, określa się też warunki wykonania badań i atestowania oraz warunki odbioru. W załącznikach do TZ, można umieścić:

- wykaz prac naukowo-badawczych i ewentualnie innych będących podstawą do opracowywanego OD lub
- inne źródła pracy.

Zawartość i forma scenariusza pedagogicznego

W scenariuszu pedagogicznym PS, będącym w istocie technicznym projektem opracowywanego OD jest opisywany dialog ucznia z komputerem, prowadzony w procesie nauczania, a więc zawartość poszczególnych kadrów i krótka informacja o pracy z danym OD.

PS jest opracowywany w drugiej fazie procesu wytwarzania OD. PS opracowuje się na podstawie TZ i jest to roboczy materiał, z którym pracuje programista w trakcie opracowywania projektu roboczego OD.

Równolegle z PS opracowuje się algorytm nauczania AO, w którym określa się kierunki wszystkich przejść od kadru do kadru, które należy wykonać w OD. Podobnie równocześnie z PS, opracowuje się instrukcje programisty IP, w której określa się wskazówki odnoszące się do samego procesu pisania danego OD. Wskazane jest, aby scenariusz pedagogiczny i algorytm nauczania połączyć w jeden dokument - nazwany ogólnie "Scenariuszem pedagogicznym". Dokument ten powinien zawierać następujące rozdziały:

- wstęp,
- opis scenariusza,
- algorytm nauczania,
- źródła wykorzystywane przy opracowywaniu PS.

W zależności od specyfiki PS, poszczególne rozdziały można łączyć, a także można wprowadzić inne nowe rozdziały. We "Wstępie" podaje się, na podstawie jakiego zadania technicznego opracowano dany PS oraz jakie są jego podstawowe cele.

PS w zależności od owych celów, zwykle powinien obejmować następujące typy kadrów: kadry początkowe /winieta/ oraz kadry instruktażowe /In/, informacyjne /If/ i pomocnicze /P/, kontrolne /K/, kadry powtórzeniowe /R/ - /repliki/, kadry pomocnicze - podpowiadające /Pp/ i inne.

W zależności od sposobu przedstawiania informacji kadry można dzielić na statyczne i dynamiczne. Kadry początkowe /winieta/ służą do opisu nazwy OD i przedstawienia jego autorów. Na winietę, ogólnie rzecz biorąc, składają się trzy kadry, tj. tytułowy, emblemat i zawierający nazwę przedmiotu.

W zakres podstawowych napisów w kadrze tytułowym zazwyczaj wchodzi: nazwa ministerstwa i instytucji zamawiającej, nazwa instytucji opracowującej dane OD oraz nazwa danego OD, nazwiska autorów, miejsce opracowania, rok i miasto.

Drugi kadr winiety powinien prezentować przyjęty emblemat opracowujących. Emblemat może też być opuszczony. Trzeci kadr winiety zawiera nazwę zawodu, dla którego jest opracowane OD, nazwę przedmiotu, rozdział, temat.

Kadry instruktażowe /In/ zawierają opis podstawowych zasad pracy OD. Obok istnienia kadrów instruktażowych /In/ w samym OD, przygotowuje się specjalny dokument programowy; jest to instrukcja dla ucznia, w której podaje się szczegółowo zasady pracy OD.

Kadry informacyjne /If/ i pomocnicze /P/ zawierają teoretyczne wiadomości z zakresu przedmiotów związanych z danym przedmiotem. Specjalne miejsce wśród kadrów informacyjnych i pomocniczych zawierają kadry demonstracyjne, które ilustrują przebieg przyrodniczych i

przemysłowych procesów i zjawisk, modelują pracę aparatury i stanowisk, pokazują przebieg doświadczeń.

Kadry kontrolne /K/ zawierają pytania i zadania pomagające w przeprowadzaniu kontroli wiedzy /wiadomości uczniów/.

Kadry podpowiadające /Pp/ zawierają wiadomości pomagające uczniom pracować z OD, odpowiadać na pytania, wypełniać /rozwiązywać/ zadania itp.

Kadry dynamiczne tym różnią się od statycznych, że stosuje się w nich dynamiczne przedstawianie wiadomości /informacji/, np. sterowane mnożenie. W porównaniu ze statycznymi, kadry dynamiczne są bardziej informujące, i dlatego zaleca się stosować je w OD jak najszerszej.

Rozdział "Algorytmy nauczania" zawiera opis algorytmu ze wskazaniem kierunków wszystkich przejść wewnętrznych OD i opis wszystkich znaczeń użytych w schemacie blokowym algorytmu.

Rozdział "Źródła wykorzystywane przy opracowywaniu scenariusza pedagogicznego" zawiera wykaz podręczników oraz innych pomocy naukowych i metodycznych, literatury popularno-naukowej, dydaktycznych środków nauczania, na których opierali się autorzy przy opracowywaniu scenariusza pedagogicznego.

Dla ukształtowania różnych kadrów celowe jest posługiwanie się specjalnymi wzorami, np. dla informacyjnych, instruktażowych, pomocniczych i innych typów kadrów, mających tylko jedno wyjście, wykorzystuje się wzór 2.

Wzór 2 .

Typ i numer kadru:
Kierunek przejścia:

W rubryce "Typ i numer kadru" wskazuje się, jaki kadr będzie opisany w ramce: informacyjny, pomocniczy, instruktażowy, powtórzeniowy i jaki jest jego numer w kolejności śledzenia tzn. pracy OD.

W rubryce "Kierunek przejścia" wskazuje się, do jakiego kadru przechodzi się po skończeniu pracy z opisanym kadrem. Do opisu kadrów kontrolnych korzysta się z wzoru 3 i 4.

Wzór 3

Typ i numer kadru: ...
+ Kierunek przejść -

Wzór 4

Typ i numer kadru:		
-2	Kierunek przejść	+
-3		-1

Wzór 3 wykorzystuje się wówczas, gdy jest możliwa tylko jedna reakcja na prawidłową odpowiedź ucznia, wzór 4 wtedy, jeżeli możliwe są różne reakcje w zależności od typu nieprawidłowości w odpowiedzi.

Wzory 3 i 4 mogą być przydatne dla opisu kadrów informacyjnych jeżeli są w nich zawarte elementy rozwiązywania problemów, a uczniowie w czasie pracy powinni odpowiadać na pytania lub rozwiązywać zadania.

Wzór 5 pokazuje sposób zapisu scenariusza w postaci tabeli.

Wzór 5

N nn	Uczestnik dialogu	Zawartość dialogu	Typ i nr kadru	Kierunek przejścia

W rubryce "Nnn" podaje się numer zapisu. W rubryce "Uczestnik dialogu" podaje się do kogo /ucznia, czy komputera/ odnosi się dany zapis. W rubryce "Zawartość dialogu" przytacza się treść, która będzie odtwarzana na ekranie monitora lub którą wprowadza uczeń z klawiatury.

W rubryce "Kierunek przejścia" wskazuje się, gdzie dokonuje się przejście po zakończeniu prac z opisanym kadrem.

Zawartość i forma instrukcji programisty

Instrukcja programisty zawiera opis podstawowych wymagań dla realizacji procesu programowania na podstawie scenariusza pedagogicznego. Instrukcja ta powinna pomóc programiście zrealizować w OD wszystkie koncepcje autora założone w scenariuszu.

Instrukcję programisty opracowuje się równolegle z opracowywaniem scenariusza pedagogicznego. Scenariusz pedagogiczny /PS/ i instrukcja programisty /IP/ mogą stanowić jeden dokument. Dla jednego scenariusza pedagogicznego można napisać parę instrukcji programisty z uwzględnieniem właściwości konkretnego systemu komputerowego. Instrukcję programisty przygotowują autorzy scenariusza pedagogicznego i jest przez nich podpisywana. Instrukcja programisty zazwyczaj zawiera wprowadzenie i wymagania dotyczące programowania.

W rozdziale "Wprowadzenie" wskazuje się, do jakiego scenariusza pedagogicznego odnosi się ta instrukcja, na jaki typ komputerów powinno się ukierunkować pracę.

Rozdział "Wymagania dla programowania" składa się z dwóch podrozdziałów. Pierwszy zawiera ogólne wymagania dla programowania, drugi wymagania co do programowania poszczególnych kadrów scenariusza. W pierwszym podrozdziale należy odpowiedzieć na następujące pytania:

- jaki powinien być maksymalny czas pracy danego OD?
- jakie zestawy kolorów należy zastosować?
- jakie sposoby przejścia pomiędzy kadrami powinny być zrealizowane w OD? /przez funkcję czasu, przez naciśnięcie odpowiednich klawiszy, przez naciśnięcie dowolnego klawisza itd./.
- jakie powinno być muzyczne tło OD?
- jakie powinno się w danym OD zbierać dane statystyczne o nauczaniu?

W drugim podrozdziale wymagania dla programowania oddzielnych kadrów OD formułuje się według następującego schematu:

- czas prezentacji kadru uczniowi,
- użyte kolory,
- sposoby budowy kadru /etapowo, łącznie, itd/,
- sposób wydzielania ważnej informacji w kadrze /oscylacja, migotanie, kolor, podświetlanie/,
- zasady budowy kadru /liczba znaczeniowych pól dla różnych informacji itp./;
- prezentacja informacji w postaci graficznej w sposób dynamiczny.

W podrozdziale "Wymagania dotyczące programowania oddzielnych kadrów" można wykorzystać tabelaryczną formę zapisu /wzór 6/.

Instrukcję programisty w ogólnej formie należy opracować zgodnie z ogólnymi wymaganiami dotyczącymi formułowania dokumentów OD.

Nr kadru	Typ kadru	Wymagania do programowania

Zawartość i formułowanie tekstu OD

Tekst OD jest dokumentem zawierającym zapis w języku programowania z niezbędnymi komentarzami.

Tekst OD formułuje się zgodnie z ogólnymi wymaganiami dotyczącymi formułowania dokumentów programowych. Część informacyjna /adnotacja i streszczenie/ nie są obowiązujące. Jeżeli wprowadza się adnotację należy w niej opisać funkcję OD. Tekst OD ogólnie zawiera jeden rozdział - zapis programu. W tym rozdziale może być przytoczony zapis OD w języku programowania lub wydruk z komputera. Należy przewidzieć w tekście OD możliwie dużą liczbę komentarzy, aby tekst był łatwo zrozumiały.

Zawartość i forma opisu OD

W dokumencie zwanym "Opis OD" należy podać funkcjonalne przeznaczenie i logiczną strukturę oprogramowania. Opis OD opracowuje się równolegle z tekstem oprogramowania i mogą być one połączone w jednym dokumencie.

Opis OD powinien zawierać następujące podstawowe rozdziały:

- uwagi ogólne,
- przeznaczenie funkcjonalne,
- opis struktury logicznej,
- wykorzystane środki techniczne,
- uruchomienie /ściąganie/.

W zależności od specyfiki OD dopuszcza się wprowadzenie uzupełniających rozdziałów lub łączenie poszczególnych rozdziałów w jeden. W rozdziale "Uwagi ogólne" należy wskazać:

- przeznaczenie i nazwę OD,
- programowanie podstawowe niezbędne dla funkcjonowania OD;
- język programowania, w którym napisano OD.

W rozdziale "Przeznaczenie funkcjonalne" należy scharakteryzować przeznaczenie OD i podać informacje o funkcjonalnych ograniczeniach przy stosowaniu tego oprogramowania.

W rozdziale "Opis struktury logicznej" powinny być wskazane struktura OD z opisem funkcji części składowych oraz związki OD z innym oprogramowaniem.

W rozdziale "Wykorzystane środki techniczne" należy wskazać typy komputerów, które wykorzystuje się przy pracy OD.

W rozdziale "Ściąganie i uruchamianie" należy wskazać:

- sposób wywołania OD z odpowiedniego nośnika danych;
- wiadomości o wykorzystaniu pamięci operacyjnej, objętość OD,
- punkty wejściowe do OD.

Dopuszcza się ilustrowanie treści rozdziałów wyjaśniającymi przykładami, tablicami, schematami, grafikami.

Zawartość i forma metodycznych zaleceń dla użytkownika-nauczyciela

Zalecenia metodyczne dla nauczyciela są dokumentem, który zawiera opis metod wykorzystywania danego OD.

Zalecenia metodyczne należy opracować po opracowaniu projektu roboczego OD. Zalecenia metodyczne dla nauczyciela powinny uwzględniać specyfikę narodową każdego kraju i specyfikę procesu nauczania.

Zazwyczaj zalecenia metodyczne powinny zawierać następujące rozdziały:

- wprowadzenie,
- wskazówki dotyczące korzystania z OD,
- opis środków technicznych i dydaktycznych, które wykorzystuje się równocześnie w OD,
- załączniki.

W rozdziale "Wprowadzenie" podaje się, do jakiego OD odnoszą się zalecenia i jaki jest podstawowy cel tych zaleceń.

W rozdziale "Wskazówki dotyczące korzystania z OD" podaje się:

- na jakiej lekcji i przy jakim temacie celowe jest wykorzystanie danego OD, jakie dydaktyczne zadania ono rozwiązuje /wyłożenie nowego materiału, kontrola wiadomości, utrwalenie wiadomości, nawyków itp./;
- w jakim momencie lekcji należy zastosować OD;
- jak przygotować uczniów do pracy z OD;
- jakie pomoce dydaktyczne /podręczniki, zbiory zadań, poradniki itd./ wskazane jest wykorzystać na lekcji równocześnie z OD.

W rozdziale "Opis środków technicznych i dydaktycznych" wymienia się pomoce typu technicznego, którymi wskazane jest posługiwać się łącznie z OD: rzutniki, przezroczka, filmy oraz pomoce typu dydaktycznego np. plansze, zadania, zeszyty ćwiczeniowe.

W rozdziale "Załączniki" przytacza się przykłady kart zadaniowych, zeszytów ćwiczeniowych, pomocniczych konspektów, które mogą być wykorzystane na lekcji.

Zawartość i forma instrukcji użytkownika - ucznia.

W instrukcji dla ucznia podaje się podstawowe zasady pracy z OD. Instrukcję należy opracowywać równocześnie z opracowaniem metodycznych zaleceń dla nauczyciela.

Instrukcję opracowują autorzy OD i powinna ona uwzględniać specyfikę każdego kraju. Zazwyczaj w Instrukcji powinny występować trzy rozdziały: wstęp, zasady pracy z OD i załączniki. We "Wstępie" podaje się nazwę OD i cel jej stosowania na lekcji. W rozdziale "Zasady pracy z OD" podaje się te zasady w formie zrozumiałej dla ucznia. W "Załącznikach" przedstawia się zadania dla ucznia związane z pracą z OD, plansze zadaniowe, warianty samodzielnej pracy ucznia z OD, wskazówki dotyczące zapisów w zeszytach, zadania domowe itp.

mgr Andrzej RAFF

Instytut Maszyn Matematycznych

Ocena oprogramowania dydaktycznego - różne podejścia

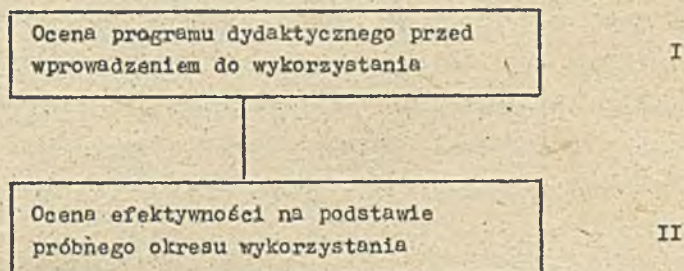
Wstęp

Brak właściwego oprogramowania uniemożliwia efektywne korzystanie z komputera. To banalne stwierdzenie jest konieczne dla następnego, równie oczywistego, mianowicie niezbędne są metody pozwalające ocenić czy dane oprogramowanie jest właściwe.

Biorąc pod uwagę skalę światową można mówić o zalewie wszelkiego rodzaju oprogramowania. Co roku pojawiają się setki różnej jakości produktów programowych. Często dziełem przypadku, reklamy czy nawet mody jest to, jakiego program /tj. jakiego typu, lub wersji/ używamy, po jakim sięgamy, co przy braku metod oceny jest nieuniknione.

Szczególnej wagi nabiera ten problem w odniesieniu do programów z dziedziny dydaktyki i nauczania. Metodami oceny oprogramowania dydaktycznego programy trafiające do szkoły powinny być weryfikowane.

Chcąc w pełni ocenić wartość komputerowych programów dydaktycznych należałoby stosować ocenę dwustopniową /rys.1/.



Rys.1. Dwustopniowy model oceny programów o przeznaczeniu szkolnym

Ocena określana jako I w dwustopniowym modelu oceny jest obecnie często stosowana, a zwłaszcza szeroko dyskutowana w tych krajach, w których wykorzystuje się już komputery w szkolnictwie. Zestaw kryteriów, według jakich oceniane są programy dydaktyczne jest różny, ale zawsze ostateczną konkluzją każdej metody jest określenie, które programy nie powinny trafić do szkoły, a które mogą być wykorzystywane w procesie nauczania. Tak więc ocena typu I pozwala wyeliminować z rynku programy słabe, kształtujące błędne nawyki, źle pomyślane od strony dydaktycz-

nej, a zwłaszcza programy zawierające błędy i nieścisłości merytoryczne.

Drugą istotną zaletą wyraźnego działania ocen typu I jest zwrócenie uwagi potencjalnych autorów programów na liczne aspekty często dotychczas pomijane, przy pisaniu programów dydaktycznych. Można tu wymienić podstawową wadę oprogramowania dydaktycznego mianowicie, niewłaściwe potraktowanie komputera jako środka nauczania. Otóż najpoważniejszym błędem jest sprowadzenie komputera do roli automatycznego podręcznika /książki/ umożliwiającego jedynie czytanie kolejnych stron. Na marginesie warto wspomnieć, że oprócz względów dydaktycznych jest to szkodliwe również ze względów zdrowotnych.

Innym błędem jest niewykorzystywanie możliwości interaktywnej pracy z komputerem. Dialog, jaki w trakcie pracy ucznia z komputerem, powinien się odbywać, powinien wpływać aktywizująco na percepcję ucznia. Bardzo też ważna jest programowa możliwość modelowania określonych zjawisk, np. przez zmianę parametrów i ich wartości, dzięki czemu uczeń niejako sam projektuje eksperyment lub ćwiczenie, oraz śledzi jego przebieg z dokładnością z jaką nigdy nie mógłby obserwować w trakcie eksperymentu technicznego.

Znajomość kryteriów oceny przez zespoły piszące programy dla szkolnictwa pozwoli na odpowiednie projektowanie scenariuszy tych programów uwzględniających owe wymagania. Z doświadczeń np. brytyjskich wynika, że jedynie niecałe 20% dostępnego na rynku oprogramowania przeznaczonego dla szkół nadaje się naprawdę do wykorzystania. Pozostałe 80% to produkty bardzo wątpliwej jakości merytorycznej i dydaktycznej 9.

Jeszcze słabiej oceniany jest przez specjalistów francuski rynek oprogramowania dydaktycznego.

Ocena oznaczona jako II, w zaproponowanym dwustopniowym modelu ocen, jest trudna do stosowania; wprowadzie niektórzy teoretycy uznają potrzebę jej przeprowadzania, ale praktycznie nie jest wykonywana. Brak dotychczas nawet zgodności co do możliwości jej stosowania. Decydującą barierą jest tu konieczność przeprowadzania długotrwałych, wiarygodnych i rzetelnych eksperymentów na reprezentatywnej próbie z całej populacji uczniów /użytkowników/. I właśnie tak sformułowany zakres oceny budzi najwięcej zastrzeżeń co do jej wykonalności. A niespełnienie wymienionych warunków powoduje, że badania stają się bezzasadne. Wreszcie jeszcze jedno spostrzeżenie: ocena oprogramowania dydaktycznego dokonywana na podstawie próbnej eksploatacji swoją praktycznością nie może stać w sprzeczności z wartością efektów, jakie można osiągnąć ze stosowania danego oprogramowania.

Eksperymenty, które mają stać się podstawą oceny typu II wymagają, nim zaczną być stosowane, określenia wielu elementów, m.in.

- wybór cech, które powinny być oceniane, propozycja tych miar oraz wartości kryterialne /graniczne/,
- sposób pomiaru poszczególnych cech,
- wypracowanie metod analizy wyników pomiaru każdej z cech,
- propozycja ogólnego modelu odniesienia i kwalifikacji oprogramowania,
- zasady wyboru grup młodzieży objętych eksperymentem,
- organizacja eksperymentu zapewniająca niezmiennność warunków w odpowiednio długim przedziale czasu /np. semestr lub rok/.

Wszystkie te problemy wymagają dopiero rozwiązania, a nie jest to możliwe bez wnikliwych studiów teoretycznych lub nawet badań eksperymentalnych. Bez wątpienia należy prowadzić prace w tym kierunku, ale na razie trudno przewidzieć ich wyniki i efekty. A nawet po opracowaniu efektywnych metod oceny II typu, uzyskanie odpowiedzi na pytanie, czy dany program komputerowy istotnie podnosi efektywność zdobywania i utrwalania wiedzy lub nawyków, może okazać się bardzo trudne. Ponieważ jednak proces dydaktyczny jest najlepszym sprawdzianem dla każdego narzędzia dydaktycznego, a więc i dla komputerowych programów dydaktycznych, dokonywanie ocen tych programów w procesie dydaktycznym wydaje się mimo wszystko najwłaściwszym i najwartościowszym sposobem oceny. Dlatego pomimo trudności powinno się poszukiwać takich metod, tzn. prowadzić stosowne prace badawcze.

W dalszej części niniejszego artykułu przedstawimy przykłady sposobów oceny I typu.

Różny sposób rozumienia i podejścia do zagadnień oceny programów spowodował opracowywanie i funkcjonowanie na świecie wielu metod, w kraju dyskutowanych jest również kilka sposobów oceny.. Do najbardziej obecnie znanych należy pierwsza z niżej omawianych metod opracowana w "Ośrodku Pomocy Naukowych", a mająca swoje źródła w materiałach francuskich. Tej metody oceny użyto w wydawanym przez Fundację Edukacji Narodowej katalogu programów dydaktycznych. Jednak według wielu specjalistów problem sposobu oceny oprogramowania dydaktycznego nadal pozostaje otwarty.

Ocena edukacyjnych programów komputerowych opracowana w Ośrodku Pomocy Naukowych

Ten sposób oceniania oprogramowania dydaktycznego składa się z oceny punktowej i opisowej. Ocena opisowa wyjaśnia i uzasadnia, które cechy zdecydowały w trakcie analizy o wyniku oceny punktowej. Wykonanie oceny punktowej wymaga ustosunkowania się do wielu szczegółowych kryteriów.

Kryteria ujęte w trzy grupy zwane przez autorów charakterystykami:

- charakterystyka techniczno-użytkowa,
- charakterystyka przedmiotowa,
- charakterystyka komputerowa.

Charakterystyka techniczno-użytkowa

/skala od 0 do 3 pkt./

Największy nacisk w tej grupie położono na poprawność merytoryczną i trafność doboru treści, a także na walory pedagogiczne: interaktywność, motywacje itd. Tak więc oceniane oprogramowanie należy przeanalizować ze względu na treści nauczania, a w tym należy uwzględnić zwłaszcza poprawność merytoryczną. Ponadto należy uwzględnić relację: program a użytkownik, w w niej tzw. oddziaływanie psychologiczne ocenianego oprogramowania czyli jego

- interaktywność oraz siłę
- motywację do nauki wywołowaną ocenianym programem i przejawiającą się jako jasność i aktywność formułowania zadań i metod realizacji itd. a również wpływ programu na koncentrację uwagi ucznia czyli w jakiej mierze fizyczna obecność komputera pomaga lub przeszkadza w nauczaniu itd.

Odrębnego omówienia wymaga odpowiednie przygotowanie uczniów i nauczycieli, komfort użyt-

kownika komputerów oraz istnienie i jakość materiałów metodycznych dla nauczycieli i uczniów.

Po przeanalizowaniu danego oprogramowania z tych wszystkich punktów widzenia przypisuje mu oceniający punkty z założonego przedziału, tj. 0 - 3.

Charakterystyka przedmiotowa /skala 0 - 6 pkt./

Dla ustalenia wartości przedmiotowej przyjęto uproszczoną wersję teksonomii Blooma. Kryteriami oceny są tu cele nauczania. Ogólnie w tej grupie ocenia się:

Cele dydaktyczne

- | | |
|------------------------|----------|
| a/ wiedza | - 3 pkt. |
| b/ zrozumienie | - 4 pkt. |
| c/ zastosowanie | - 5 pkt. |
| d/ wyższe umiejętności | - 6 pkt. |

Ocena powinna dawać odpowiedź na następujące pytania:

- czy program umożliwia realizację ogólnych celów kształcenia zawartych w programie nauczania i w jakim stopniu?
- czy cele i przewidywane efekty zostały jasno sformułowane?

Ponadto analizuje się, jaka została zastosowana strategia dydaktyczna, a mianowicie jakie jest miejsce komputera w procesie nauczania, tj. w jakich partiach materiału można wykorzystać program? oraz jaka jest sekwencja metodyczna tzn. włączenie programu w ciąg czynności dydaktycznych, powiązanie z pracami doświadczalnymi, prace uzupełniające bez komputera, itp.

Jako ostatnią w tej grupie analizuje się jakość metodyczną oprogramowania, tzn.

- czy metoda prezentacji materiału jest właściwa z punktu widzenia dydaktyki?

Charakterystyka komputerowa /skala 0 - 1 pkt./

Oceny podstawowej dokonuje się tu ze względu na funkcję, jaką spełnia komputer w procesie dydaktycznym. Ocena ta jest ułamkiem z przedziału 0,1

Funkcja komputerowa w procesie dydaktycznym

- czy komputer jest użyty jedynie jako sprawdzian wiadomości? /1/2/,
- czy komputer jest wykorzystany jako środek nauczania /2/3/,
- czy program uczy zastosowań komputera /komputer "obiektem" nauczania/? /1/.

Wykorzystanie komputera w sytuacji dydaktycznej.

Możliwość realizacji celów nauczania za pomocą innych środków dydaktycznych.

Ocenie podlega tu również kwestia czy program dobrze wykorzystuje możliwości komputera oraz czy komputer jest w tym wypadku lepszym środkiem dydaktycznym niż inne.

Uzyskana podstawowa ocena punktowa /1/2,2/3 lub 1 punkt/ może zatem być skorygowana /zmniejszona lub zwiększona/ w zależności od tych aspektów. Ta grupa kryteriów umożliwia praktyczne skorygowanie całej oceny z punktu widzenia funkcji, jaką pełni komputer w procesie nauczania.

Ocena punktowa łączna jest iloczynem trzech wyżej uzyskanych ocen /jest to zasadniczo skala 0 - 18 pkt./ i służy do porównania programów z tej samej dziedziny i tego samego rodzaju.

Program, w którym komputer występuje jako obiekt nauczania może osiągnąć maksymalnie

18 pkt., natomiast gdy komputer służy jako środek dydaktyczny 0 - 12 pkt. Programy, które testują wiadomości mogą uzyskać co najwyżej 9 pkt.

Można mieć wątpliwości, czy takie podejście jest słuszne. Może lepiej byłoby w każdej z tych grup osobno klasyfikować programy a nie porównywać ich między grupami.

Kwestionariusz wartościowania oprogramowania mikrokomputerowego, opracowany pod przewodnictwem: E. L. Klopfera - University of Pittsburgh

Kwestionariusz ten powinien pomóc w badaniu i dyskutowaniu zalet oprogramowania minikomputerowego, które mają być użyte w nauczaniu przedmiotów ścisłych. Kwestionariusz podaje praktyczny sposób postępowania i podstawowe kryteria oceny oprogramowania. Pozwala również na dołączenie tych kryteriów, które zdaniem nauczycieli powinny być uwzględnione.

W metodzie tej poddany jest ocenie cały komplet oprogramowania, w skład którego wchodzi:

- program komputerowy,
- towarzyszące materiały instruktażowe dla ucznia, nie występujące w programie,
- materiały informacyjne /wskazówki metodyczne/ dla nauczycieli i/lub dokumentacja programu.

Na ostatniej stronie kwestionariusza jest miejsce, gdzie można opisać składniki kompletu, jego zawartość i inne cechy charakterystyczne.

Cztery części tego kwestionariusza zwracają uwagę na cztery istotne aspekty oceniania kompletów oprogramowania mikrokomputerowego: zamierzenia dydaktyczne, poziom naukowy, wartość dydaktyczną i wartość techniczną. Każdy aspekt powinien być oceniany oddzielnie, a w wyniku podaje się wszystkie cztery oceny, dając tym samym charakterystykę całego kompletu oprogramowania /tabelka na ostatniej stronie/.

Każda część kwestionariusza zawiera podany w opisowy sposób zbiór kryteriów typowych dla danego aspektu oceny kompletu oprogramowania. Do oceny każdej części używana jest dwubiegunowa skala /z wartościami "+" i "-"/, zatem ocena kompletu może być w każdej części dodatnia, zero-owa lub ujemna. Autorzy metody założyli, że żadna z czterech ocen w charakterystyce programu nie powinna być ujemna, o ile program ma być zaakceptowany. Podobnie cały komplet oprogramowania powinien mieć oceny zdecydowanie dodatnie, zanim zostanie zaakceptowany do nauczania. Jednak dokładny poziom wymagany dla zaakceptowania kompletu oprogramowania do używania, powinien być zdaniem autorów metody określony na podstawie lokalnych warunków i zdrowego rozsądku oceniającego.

Część P /Policy Issues/: Zamierzenia dydaktyczne

Ta część zajmuje się najtrudniejszymi /i jak się wydaje najważniejszymi/ pytaniami, na które należy odpowiedzieć, rozważając jakiegokolwiek oprogramowanie dydaktyczne. Pytania te dotyczą stosowalności komputera jako środka dydaktycznego, zgodności z celami i metodami dydaktycznymi oraz innymi środkami dydaktycznymi, a ponadto efektywności kosztów, jakie należy ponieść na opracowanie danego programu /tzn. czasu i pieniędzy/ i efektywności nauczania za pomocą danego kompletu oprogramowania.

Wśród tych pytań znajdują się następujące:

- Czy specjalne możliwości komputera zostały wykorzystane dla osiągnięcia efektów dydak-

tycznych niełatwo osiągalnych innymi środkami?

- Czy program komputerowy dobrze wykorzystuje czas pracy ucznia przy komputerze?
- Czy oprogramowanie jest zgodne z celami i teoretycznymi podstawami szkolnego programu nauczania?
- Czy komputer zachęca uczniów do wzajemnej współpracy?
- Jakie są dowody na to, że uczniowie osiągają założone wyniki nauczania?

Inne sprawy o podobnym znaczeniu, powinny być dodane do kryteriów w tej części.

W czasie pewnej dyskusji w środowisku nauczycieli nad prezentowaną metodą, niektórzy z dyskutantów sugerowali, że jeśli pakiet oprogramowania w poważnym stopniu nie spełnia wymagań stawianych w tej części, nie musi być poddawany dalszym badaniom. Niemniej autorzy metody uważają, że zdecydować o tym powinni ci, którzy dokonują danej oceny.

Część S /Science Subject-Matter Standards/: poziom naukowy

Dobre nauczanie musi być oparte na solidnej wiedzy. W celu przekonania się, że pakiet oprogramowania spełnia te oczekiwania, w tej części zajmujemy się poprawnością zawartości naukowej, trafnością zastosowań osiągnięć naukowych, brakiem stereotypów i innymi kwestiami związanymi z uczciwym przedstawieniem wiedzy w nauczaniu.

W tej części jest wiele miejsca na dodanie /jeśli oceniający chce/ tych kryteriów, które są istotne w poszczególnych dziedzinach nauki.

Część I /Instructional Quality/: wartość dydaktyczna

Ta część jest związana z kwestią efektywności pedagogicznej, zastosowaniem dobrych zasad nauczania, zdolności adaptacji oprogramowania do indywidualnych możliwości uczniów, oszacowaniem osiągnięć uczniów i zaplanowaną rolą ucznia korzystającego z oprogramowania.

Część T /Technical Quality/: wartość techniczna

W centrum zainteresowania w tej części leży wartość techniczna zarówno programu komputerowego, jak i innych składników kompletu.

Oceniającego interesuje tu, jak funkcjonuje program, jak jest on opracowany pod względem operacyjnym i jak są opracowane materiały towarzyszące dla nauczyciela i ucznia. Dodatkowe kryteria mogą być tu przydatne, szczególnie wymagania sprzętowe lub inne dotyczące sprawności całego kompletu.

Dokonanie ocen

Każda część kwestionariusza zawiera zestaw dwubiegunowych skal. Każde dodatkowe kryteria powinny być skonstruowane w podobny sposób. Oceniający powinien starannie rozważyć opisy na obu krańcach każdej skali /w tłumaczeniu na polski oznaczane przez L - lewa i P - prawa/ i przyznać wartość z przedziału od - 3 do + 3, zgodnie z tym jak dobrze lewy lub prawy opis odpowiada oceniającemu kompletowi. Zaznacza się tylko jeden punkt na skali.

Jeśli nie można podjąć decyzji, zaznacza się punkt zerowy. Ocena każdej części otrzymuje się dodając przyznane punkty na każdej skali tej części. Ocena wypisuje się do tabelki na ostatniej stronie. Dolna część tabelki powinna zawierać minimalne wartości, jakie są potrzebne aby komplet mógł być zaakceptowany. Porównanie uzyskanych ocen z minimalnymi pozwala określić stopień przydatności ocenianego oprogramowania.

Każdą skalę ocen należy interpretować zgodnie z poniższym zestawieniem.

Skala ocen:	lewy opis	prawy opis
zdecydowanie prawdziwy	- 3	+3
częściowo prawdziwy	- 2	+2
prawdziwy w niewielkim stopniu	- 1	+1
żaden opis nie ma zastosowania	0	

Dla ilustracji proponowanego sposobu oceny przytoczono kilka charakterystycznych pytań, jakie zaproponowali autorzy metody w swoim angielskim oryginale. A więc

w części P- zamierzenia dydaktyczne - znajdują się takie pytania:

L: Program powoduje, że komputer staje się niczym innym, jak urządzeniem do przewracania kartek.

P: Program wykorzystuje specjalne możliwości komputera /np. animowaną grafikę, symulację/ dla osiągnięcia efektów dydaktycznych trudno osiągalnych innymi środkami.

L: Program stanowi stratę ograniczonego czasu pracy ucznia z komputerem.

P: Program dobrze wykorzystuje ograniczony czas pracy ucznia z komputerem.

L: Cele i zamierzone wyniki nauczania są niejasne.

P: Cele i zamierzone wyniki nauczania są dobrze określone.

L: Komplet stoi w sprzeczności lub jest nieistotny dla realizacji celów szkolnego programu nauczania.

P: Program jest zgodny z celami i założeniami teoretycznymi szkolnego programu nauczania.

L: Jeden uczeń ma pracować z komputerem, nie porozumiewając się z nikim innym.

P: Program zachęca do wzajemnej współpracy dwóch lub więcej uczniów.

L: Jest niewiele lub nie ma wcale dowodów na to, że uczniowie osiągają założone wyniki nauczania.

P: Widać, że uczniowie osiągają oczekiwane wyniki nauczania.

L: Komplet jest sprzeczny z celami i materiałami metodycznymi aktualnego programu.

P: Komplet jest dobrze dopasowany do innych materiałów metodycznych używanych w danym programie nauczania.

L: Cena kompletu jest "astronomiczna" w porównaniu z tym, co on oferuje.

P: Cena kompletu jest rozsądna w porównaniu z jego wartościami dydaktycznymi.

W oryginalnym formularzu po tych ośmiu pytaniach zostawiono dwa dodatkowe wolne miejsca przeznaczone na pytania własne oceniającego.

W części S- poziom naukowy - znajdują się pytania:

- L: Komplet prezentuje tematy nieistotne dla edukacyjnych potrzeb ucznia.
 - P: Tematy zawarte w komplecie są bardzo znaczące dla wykształcenia ucznia.
 - L: Zawartość naukowa jest nieścisła.
 - P: Zawartość naukowa jest bezbłędna.
 - L: Program prezentuje i utrwala stereotypy rasowe, etniczne, czy płciowe.
 - P: Prezentacja jest wolna od niekorzystnych stereotypów.
 - L: Informacje stroniczne lub wypaczone są przedstawione jako informacje faktyczne.
 - P: Prezentowane informacje są dobrze wyrażone i reprezentowane.
 - L: Komplet zawiera przestarzałe poglądy naukowe.
 - P: Zawartość naukowa odpowiada współczesnej wiedzy.
 - L: Prezentacja treści naukowych jest pogmatwana.
 - P: Zawartość naukowa jest jasno przedstawiona
 - L: Komplet nie zawiera uwagi na proces badania naukowego.
 - P: Postępowanie badawcze jest istotnym elementem pracy ucznia.
- Na koniec przewidziano trzy dodatkowe wolne miejsca na pytania własne oceniającego.

W części I- jakość dydaktyczna - zaprezentowano również, jak w części P osiem pytań:

- L: Oprogramowanie i jego otoczenie zostawia uczniowi niewiele inicjatywy.
- P: Program oferuje wiele możliwości pracy nad tematem, stopni trudności i szybkości prezentacji.
- L: Uczeń jest pasywny podczas pracy z komputerem i robi niewiele więcej niż naciskanie od czasu do czasu dowolnego klawisza.
- P: Uczeń jest aktywnie zaangażowany we współpracę z programem komputerowym.
- L: Strategie dydaktyczne wykorzystywane w programie nie biorą pod uwagę osiągnięć naukowych pedagogiki czy psychologii.
- P: Strategie dydaktyczne kompletu są oparte na ważnych osiągnięciach naukowych pedagogiki czy psychologii.
- L: Program nie dostosowuje się łatwo do różnic w zdolnościach uczniów, uprzednio zdobytej wiedzy, czy stylu uczenia się.
- P: Program umożliwia dostosowanie się do indywidualnej zdolności uczniów.
- L: Komplet nie informuje uczniów o tym, czego będą się uczyć i co mają robić.
- P: Wskazówki w komplecie informują uczniów, do czego zmierzają /cele/ i co będą robić /metody postępowania/.
- L: Narzucona przez program organizacja pracy ucznia i zawarte w nim metody oceny są niezgodne z zasadami pedagogiki.
- P: Nauczanie za pomocą kompletu odbywa się w dobrej kolejności, ma cechy motywacyjne i właściwe metody oceny.
- L: Oprogramowanie zakłada, że wszyscy uczniowie osiągną ten sam poziom zaawansowania.
- P: Uczniowie pracujący z kompletem mogą odnosić sukcesy w osiąganiu różnych poziomów

zaawansowania.

L: Autor nie umożliwia wykorzystania innych środków nauczania w klasie.

P: Komplet współpracuje z dostępnymi środkami nauczania.

Na koniec zarezerwowano na pytania ocenianego dwa dodatkowe wolne miejsca.

Wreszcie w części T+ wartość techniczna zestawiono pytania:

L: Uczniowie potrzebują niedopuszczalnie dużej pomocy ze strony nauczyciela, aby z powodzeniem obsługiwać program.

P: Uczniowie mogą z łatwością obsługiwać program po krótkim okresie przygotowawczym.

L: Reakcje komputera na odpowiedzi uczniów są niewłaściwe i mylące.

P: Reakcje komputera na odpowiedzi uczniów są właściwe, informujące i na czasie.

L: Ilustracje graficzne są bezładne i "niewyglądzone".

P: Ilustracje graficzne są jasne.

L: Zachowanie komputera jest gruboskórne i obraźliwe.

P: Program jest przyjazny dla użytkownika.

L: Program ma niepoprawione błędy powodujące czasami zachowanie "sprzeczne" lub "pada".

P: Wszystkie możliwe kombinacje danych są przewidziane przez program, powodując, że jego działanie jest przewidywane i odpowiedzialne, godne zaufania.

L: Dokumentacja jest niekompletna, myląca, sprzeczna z obserwowanym działaniem programu.

P: Dokumentacja programu jest zwarta, jasna i zgodna z działaniem.

L: Materiały towarzyszące są źle zorganizowane, nieatrakcyjne i niewłaściwe.

P: Materiały towarzyszące są dobrze zaprojektowane, właściwe dla ucznia korzystającego z nich.

L: Wskazówki metodyczne dla nauczyciela są niekompletne, ubogie, pisane żargonem.

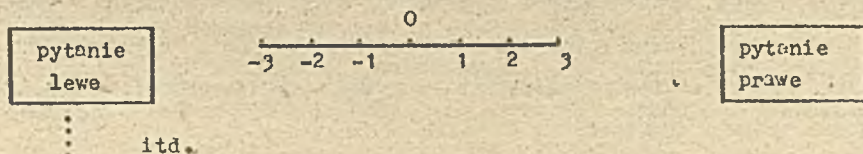
P: Wskazówki metodyczne są atrakcyjne, pełne i odpowiednie dla nauczyciela mającego niską wiedzę informatyczną.

L: Komplet jest mało odporny fizycznie.

P: Składniki kompletu są zaprojektowane tak, aby przetrzymały warunki pracy w klasie.

Dla oceniającego pozostawiono jedno dodatkowe miejsce.

W oryginalnym formularzu, w celu łatwiejszego dokonywania ocen każdemu z pytań nadano poniższą formę graficzną /rys.2/, Wydaje się, że istotnie jest to sposób bardzo ułatwiający dokonywanie oceny.



Ryc.2. Graficzna skala przyznawania ocen na poszczególne pytania

Jak wspomniano - na końcu formularza zamieszczono tabelę przeznaczoną do wpisywania ocen łącznych za każdą z grup i analizy ocen.

P	S	I	T	
				miejsce na wpisanie ocen
				oceny porównawcze

Na ostatniej stronie kwestionariusza oceniający wypełnia następujące rubryki:

Tytuł kompletu oprogramowania:

Wydawca lub rozpowszechnienie:

Oceniający:

Tabela:

Komentarze i rekomendacje:

Opis kompletu oprogramowania:

/tematyka, typ programu, poziom, materiały drukowane dla uczniów, przewodnik dla nauczyciela/

Wymagania sprzętowe:

Sposób oceniania programów dydaktycznych opracowanych w IMM

Gdy podejmowano w latach 1984 - 85 w Pracowni SR Instytutu Maszyn Matematycznych prace nad stosowaniem komputerów w dydaktyce, nie było w kraju żadnych metod oceny oprogramowania dydaktycznego. Opisane w poprzednich rozdziałach metody zostały wprowadzane w kraju dopiero w latach '87, 88. Dlatego też uznano, że opracowanie chociażby na własny użytek jakiejś metody jest sprawą nieodzowną. Przystępując do prac nad taką metodą stwierdzono, iż właściwie brak jest mierzalnych cech oprogramowania dydaktycznego. W istocie wszystkie wymienione w literaturze metody, czego przykładem są przytoczone w niniejszym referacie, są metodami subiektywnymi, opartymi głównie na intuicji, wiedzy i doświadczeniu oceniającego. Wobec powyższego za podstawę własnej metody przyjęto podejście, w którym stosuje się skalę dwuwartościową. TAK - nadaje się do stosowania, NIE - nie nadaje się do stosowania. Sposób ten nie pozwala na ocenę, który z programów jest lepszy, a który gorszy.

Ostateczny wybór z paru dopuszczalnych programów - o ile takie istnieją - pozostawia się nauczycielowi. Tak więc podstawowym założeniem przyjętej metody jest "zasada sita". - Oceniający dany program musi odpowiedzieć na wszystkie 15 pytań, i tylko odpowiedź TAK na wszystkie dopuszcza program do użycia w szkołach:

1. Czy program jest uruchomiony?
2. Czy program jest dostępny na nośniku maszynowym?
3. Czy program jest poprawny merytorycznie?
4. Czy polecenia są formułowane jasno i zwięźle?
5. Czy program jest poprawny językowo?

Program edukacyjny nie może zawierać błędów ortograficznych, stylistycznych, itp.

6. Czy sposób rozplanowania informacji na ekranie jest przejrzysty i interesujący?
7. Czy uczeń ma wpływ na szybkość przebiegu lekcji komputerowej?
8. Czy reakcja programu na błąd ucznia jest kulturalna - nie może być obraźliwa.
9. Czy dźwięk, o ile jest, nie pogarsza warunków pracy z programem?
10. Nie można używać dźwięku do sygnalizacji błędu ucznia, czy program spełnia ten warunek?
Dziecko nie może być dyskwalifikowane w oczach kolegów za błędną odpowiedź.
11. Czy prawidłowa odpowiedź jest dostatecznie nagradzana? Nagroda za prawidłową odpowiedź powinna być bardziej znacząca dla ucznia, niż krytyka błędu.
12. Czy program jest odporny za nieprawidłową reakcję ucznia? W programie musi być uwzględniona możliwość pomyłki osoby wykorzystującej go; np. przyciśnięcie niewłaściwego klawisza - nie-dopuszczalna jest tu niezrozumiała reakcja programu lub jego przerwanie.
13. Czy program nie godzi w interes kraju oraz czy nie obraża uczuć np. religijnych itp.
14. Czy program jest zgodny z aktualnym programem nauczania?
15. Czy jest dostateczna dokumentacja programu?

Widać więc, że istotnie każde z tych pytań ma charakter kwalifikacyjny. Przy paru z pytań podano komentarz wyjaśniający dlaczego odpowiedź negatywna dyskwalifikuje program. Uczyniono to dlatego, iż w początkowym okresie stosowania "metody DNM" dla niektórych oceniających owe pytania mogą nie wydawać się aż tak ważne aby dyskwalifikowały program.

Metodę tę zastosowano w latach '85-87 przy ocenie głównie oprogramowania mikrokomputerów Meritum, jakie "Zakład Zastosowania Mikrokomputerów Zrzeszenia MERA" otrzymywał z tych szkół, którym dostarczał owe komputery. Zakładano wtedy, że otrzymane ze szkół, a opracowywane przez uczniów i nauczycieli programy staną się dla "Zakładu Zastosowań" produktami do pewnego stopnia ekwiwalentnymi za przekazany szkołom sprzęt. Stąd też wyrasta potrzeba jakiejś, chociażby bardzo uproszczonej ale jednoznacznej metody oceny programów, i taką właśnie uproszczoną metodą - jest niewątpliwie "metoda DNM".

Uwagi końcowe

W referacie przedstawiono trzy różne podejścia do problemu oceny oprogramowania dydaktycznego. Wszystkie są to oceny typu I - czyli dokonywane przed dopuszczeniem danego oprogramowania do stosowania.

Każda z nich ma pewne zalety ale i poważne wady. Wszystkie są mniej lub bardziej subiektywne. Pozostaje więc zupełnie nie rozstrzygnięty problem wiarygodności dokonywanych ocen, czyli doboru oceniających, ich liczby, kwalifikacji, odpowiedzialności itp. W literaturze można spotkać także inne metody i jak na razie nie ma żadnej metody powszechnie przyjętej i zaakceptowanej.

Konieczne jest dalsze prowadzenie studiów i badań w tym zakresie, przy czym badania takie przynajmniej w kraju powinny być poprowadzone bardziej metodycznie niż dotychczas.

Literatura

1. Klopfer E. Microcomputer software equation instrument University of Pittsburg 1984.
2. Sepko-Guzicka E.,Guzicki W.: Uwagi o opiniowaniu komputerowych programów dydaktycznych 1987.
3. Katalog oprogramowania dydaktycznego - 1987.
4. Jaszejska E.,Dunin-Borkowski J.: Ocena oprogramowania dydaktycznego.
5. Bramble W.J.,Mason R.J. : Computer in school, 1984.
6. L.E. Klopfer: Intelligent Tutorial System in Science Education: The Coming Generation of Computer-Based Instructional Programs, 1986.
7. Edwardson R.: "Informatyka w szkole", materiały II krajowej konferencji. Wałbrzych 1986.
8. Materiały III krajowej konferencji. Wałbrzych 1987.
9. Raff A.: Własne materiały i sprawozdania z udziału w konferencjach i seminarjach.

NOWOŚCI TECHNICZNE

Nowe organizacje powołane do opracowania standardów

W przeszłości czas wprowadzania nowego wyrobu na rynek przez poszczególne przedsiębiorstwa nie był związany z wprowadzaniem wyrobów innych firm, decydowały tylko względy marketingu. Przy nowym cyklu standardów, istnieją czynniki techniczne, które motywują synchronizację wprowadzania pewnych klas wyrobów na skalę dotychczas nie spotykaną. Przykładem tego była konferencja i wystawa Enterprise w czerwcu 1988 roku, na której pokazywano po raz pierwszy wiele urządzeń, zaprojektowanych do współpracy z innymi.

Koordinacja ta powinna być kontynuowana i do tego celu służą nowo powołane organizacje użytkowników poszczególnych grup standardów, jak MAP (Manufacturing Automation Protocol - protokół automatyzacji wytwarzania), TOP (Technical and OFFICE Protocol - protokół zastosowań technicznych i biurowych), COS (Corporation for Open Systems - stowarzyszenie otwartych systemów) i seminaria realizatorów OSI w Narodowym Biurze Standardów (NBS OSI Implementators Workshops). Wspierane są one przez firmy, które inicjowały te standardy, jak General Motors dla MAP i Boeing dla TOP. Brak jest również informacji potrzebnej do koordynowania równoległych wysiłków standaryzacyjnych. Stara się to wypełnić swoimi usługami organizacja Omnicom.

W Europie program ESPRIT uwzględnia zarówno wymagania ciągłości opracowywania standardów wynikające z inwestycji badawczo-rozwojowych jak i koordynowania inwestycji społecznych potrzebnych w tej epoce "przewidywanych standardów".

Stowarzyszenie Komputerowe IEEE opublikowało pracę o strategii rozwoju techniki Informatycznej w różnych krajach (National Computer Policies), w której porównuje się działania w tym zakresie w USA i poza ich granicami.

Computer Age - World Trade nr 3/88

Konferencja na temat opracowywania standardu w sieci komputerowych

W marcu 1988 r. odbyła się w Arlington konferencja COMPSTAN'88 organizowana przez IEEE Computer Society, na której określono zmiany w polityce i planowaniu nowych standardów techniki Informatycznej. Przedstawiono tu liczne prace świadczące o nowym zrozumieniu w planowaniu i zarządzaniu standardami jako zasobem przedsiębiorstwa.

Problemy techniczne w zastosowaniu standardów sieciowych dyskutowane były na Sympozjum Sieci Komputerowych, które odbywało się w dniach 11-13 kwietnia 1988 r. w Sheraton National Hotel. Wygłoszone prace dotyczyły analizy i optymalizacji rozproszonych sieci, przydziału zasobów, sterowania przesyłaniem, trasowaniem i zrównoważeniem obciążenia, a także aspektami technicznego zarządzania protokołami OSI obejmując ich specyfikację, weryfikację i zabezpieczenie. Firma Bell przedstawiła opracowanie odnoszące się do zakładu opartego na Inteligentnej Cyfrowej Sieci Zintegrowanych Usług (Integrated Services Digital Network - ISDN).

25 kwietnia 1988 r. odbyła się w Narodowym Biurze Standardów dyskusja nad zastosowaniem nowego cyklu standardów do technik Informatycznych innych niż OSI. Omawiano profile normalizacyjne specyfikacji uogólnionego sprzęgu dla grup standardów dopełniających się technik Informatycznych, które łącznie pracują podtrzymując przenośne programy aplikacyjne. Sprzęg POSIX do systemów operacyjnych jest podstawą przy tworzeniu oprogramowania systemów dla ogólnych klas zastosowań.

Światłowodowy umożliwia zastosowanie PC jako stacji graficznej

SunRiver Corp. z Jackson w stanie Missouri rozpoczęła sprzedaż (po 899 dolarów) t.zw. PC LightCard wykorzystującą światłowod do CAD, wydawnictw i wyszukiwania obrazów przy pomocy szybkiego mikrokomputera Intel 80386 działającego jako procesor z mapą bitową. Karta ta dopasowana jest do łączówek zapasowych PC zarówno 8 jak i 16 bitowych. PC może wówczas pracować w DOS lub pod Unixem/Xenixem. Szybkość przesyłania danych wynosi 32 Mb/s.

Mini Micro Systems nr 10/88

Karta pozwalająca na szybkość 80 MFLOPS

Interstate Electronics z Anaheim w Kalifornii zapowiada wprowadzenie pod koniec 1989 roku zespołu kostek pozwalających na osiągnięcie 80 milionów operacji zmiennego przecinka na sekundę przy karcie wielkości jak w PC. Przewidywany koszt karty jest rzędu 3 tysięcy dolarów przy czym istnieje możliwość użycia czterech takich kart i osiągnięcie 320 MFLOPS za mniej niż 20 tys. dolarów.

Problemy rozwiązywania metodami o wysokim stopniu równoległości obejmują projektowanie wspomagane komputerowo, tworzenie obrazów, zastosowania inżynierskie i symulację, a także dynamikę płynów i przetwarzania sygnałów, a więc dziedzin, gdzie wykorzystywane są superkomputery.

Omawiany zestaw kostek opiera się na architekturze QUEN opracowanej na uniwersytecie Johna Hopkina, która wykorzystuje macierz procesorów, gdzie każda para sąsiednich procesorów dzieli pamięć o podwójnym wejściu. Informacja jest przesyłana przez pamięć, a jednostki pamięciowe zajmują się adresowaniem i sterują przesyłaniem danych.

Projekt stwarza szybki równoległy system przetwarzania z maksymalnym wykorzystaniem mocy obliczeniowej.

Pierwsze procesory oparte na architekturze QUEN budowane były na 32-bitowych układach cyfrowych przetwarzania sygnału firmy Analog Devices. Posiadały one stałoprzecinkową arytmetykę i były bardzo szybkie.

Byte nr 9/88

Rozwój kostek z arsenku galu

Dotychczas kostki z arsenku galu traktowane były jako egzotyczna lecz droga technika. Eksperci orzekli, że kostki te pojawią się w komputerach osobistych, gdy cena ich spadnie, a uzysk wzrośnie. Oba te warunki powoli zaczynają być spełniane. Firma Gazelle Microcircuits z Santa Clara w Kalifornii oferuje ostatnio GA22V10. Jest to wersja programowalnego układu logicznego 22V10 realizowane w GaAs. Kostkę tę stosuje się w układach buforowania pamięci systemów obliczeniowych o wysokiej wydajności opartych na mikroprocesorach 80386 lub 68020, a także w nowych komputerach o zredukowanej liczbie rozkazów (RISC). Służy ona jako sprzęg pomiędzy szybkim procesorem i wolniejszą pamięcią dynamiczną lub szybką pamięcią notatnikową.

GA22V10 wykorzystuje standardowe poziomy logiczne TTL na wszystkich wejściach i wyjściach i zasilany jest jednym napięciem 5V. Wewnątrz jednak wszystkie bramki realizowane są na GaAs, w wyniku czego opóźnienie propagacji nie przekracza 10 ns, co pozwala na pracę z częstotliwością co najmniej 90 MHz. Nie dąży się tu do uzyskania maksymalnej szybkości, raczej kładąc nacisk na niezawodność i uzysk.

Układ jest trzykrotnie droższy od układów krzemowych (55 dolarów) wobec 16 dolarów za krzemowe o dwukrotnie gorszej wydajności przy 100 sztukach, a więc jeszcze zbyt kosztowny dla większości zastosowań PC, lecz dla niektórych projektantów zalety GaAs - wysoka szybkość i mniejsza zależność od temperatury i napięcia - są istotne. Przewiduje się, że w ciągu 3-4 lat cena kostek GaAs spadnie do około 10 dolarów. Obecnie w kostce jest około 10 tysięcy bramek i wartość ta podwaja się co roku, a więc w ciągu 2-3 lat osiągnie się wartość potrzebną dla jednostki centralnej.

Wiadomo, że firma Texas Instruments łączy układy GaAs i tranzystory krzemowe w jednej kostce wykorzystując szybkość pierwszych oraz upakowanie i małą moc drugich. Stwarza to jeszcze bliższe perspektywy stosowania GaAs w komputerach osobistych.

Byte nr 9/88

Pokrywające się częściowo w czasie projekty opracowania standardów mają potrzebę pośrednich standardów dla pewnych składników profili tak, aby odpowiadały one standardom, które mogą być określone przy czasowej realizacji wyrobów. Czekanie na zakończenie procesu przyjęcia standardów dla jednego lub kilku składników profilu mogłoby narzucić niedopuszczalne koszty na użytkowników standardów i sprzedawców sprzętu podległego standardom. Dyskutowano też opracowanie tymczasowych standardów, zwłaszcza dla rządowych potrzeb przetwarzania informacji.

Computer Age - World Trade nr 3/88

Nowy mikrokomputer Hewlett-Packarda

Naburkowy komputer osobisty HP Vectra QS/16 wyposażony jest w mikrokomputer Intel 80386 z zegarem o częstotliwości 16 MHz. Standardowa wersja zawiera 1 MB pamięci operacyjnej i stację dysków elastycznych albo 1,2 MB o średnicy 5,25 cala, lub 1,44 MB o średnicy 3,5 cala, kontroler zarówno do dysków elastycznych, jak i sztywnych oraz jeden port szeregowy, jeden równoległy i możliwość zastosowania koprocatora arytmetycznego 80387. Pamięć operacyjna może być rozszerzona do 16 MB bezpośrednio na pakiecie procesora.

Model podstawowy z dyskiotką 1,2 MB kosztuje 3995 dolarów, a z dyskiem twardym 40 MB - 5095 dolarów. Najdroższa wersja (5495 dolarów) obejmuje dysk twardy 40 MB, kartę VGA i wybór stacji dysków elastycznych (1,2 MB 5,25 cala lub 1,44 MB 3,5 cala).

PC Tech Journal nr 12/88

Nowy model IBM PS/2 30

Mikrokomputer PS/2 Model 30 286 oparty jest na mikrokomputerze 80286 i jest dwukrotnie szybszy (10 MHz) od oryginalnego Modelu 30, a pojemność pamięci jest 25-krotnie większa (512 kB rozszerzone na pakiecie głównym i 16 MB z dodatkowymi pakietami). Posiada on kartę VGA, stację dyskiotek 3,5 cala o pojemności 1,44 MB i trzy łączówki dodatkowe typu AT. Model 30 286-E01 (bez dysku twardego) kosztuje 1995 dolarów, a model 30 286-E21 (z dyskiem 20MB i wbudowanym sterownikiem) 2595 dol.

PC Tech Journal nr 12/88

Przenośny mikrokomputer firmy Compaq

Spełniający wszystkie funkcje przenośny komputer osobisty SLT waży 7 kg i wyposażony jest w mikroprocesor 80C286 z zegarem o częstotliwości 12MHz. Standardowa wersja obejmuje kartę VGA i monitor o przekątnej 10 cali, stację dyskiotek o średnicy 3,5 cala, stację dysku twardego, pamięć operacyjną 640 kB (rozszerzaną do 3,64 MB) i dołączoną klawiaturę z przyciskami o pełnym wymiarze.

Wewnętrzna bateria zapewnia pracę przez 3,5 godziny, przy czym ładuje się ona w ciągu 1,5-3 godzin, gdy system nie jest używany, bądź też może być ładowana w ciągu 6-10 godzin przy dołączeniu systemu do adaptera prądu zmiennego. Model 20 (z dyskiem twardym o pojemności 20 MB) kosztuje 5399 dolarów, a Model 40 - 5999 dol.

PC Tech Journal nr 12/88

Wspólne opracowanie DEC i Tandy Corp.

Obie firmy zawarły umowę na sprzedaż przez DEC odpowiedników IBM PC firmy Tandy i które będą odpowiednio dostosowane do środowiska DEC, zwłaszcza jeśli chodzi o właściwości sieciowe. Komputery te opierać się będą na procesorach Intel 80286 i 80386.

Mini Micro Systems nr 10/88

Wymazywalny dyskowy system optyczny

Firma NeXT Inc. z Palo Alto w Kaliforni oferuje NeXT Computer System, którego twórcą jest były założyciel firmy Apple, Steven P. Jobs. Jest to pierwszy system wykorzystujący jako moduł pamięci zewnętrznej - stację wymazywalnych dysków optycznych o pojemności 256 Mbajtów. Zawiera on procesor Motorola 68030 i 8Mbajtów pamięci operacyjnej oraz kostkę Motorola z procesorem sygnału cyfrowego służącą za modem, a także do transmisji FAX, przetwarzania maclorzy, poczty dźwiękowej, syntezy mowy i szybkich obliczeń numerycznych.

System operacyjny NextStep oparty jest na systemie Unix, umożliwia graficzny sprzęg z użytkownikiem za pośrednictwem okien, a Application Kit (zestaw zastosowań) języka C ułatwia opracowywanie oprogramowania.

Mini Micro Systems nr 10/88

Nowy system dysków optycznych firmy Matsushita

System ten pozwala na zapis 2,6 GB informacji na każdej stronie 12-calowego dysku o jednostronnym zapisie. Odczyt i zapis odbywa się z szybkością 18 MB/s. W zastosowaniach audiowizualnych wystarczy to na pamiętanie 500 okresów statycznych i godzinę dźwięku stereofonicznego.

Byte nr 9/88

Pierwsze dyski Micropolis o średnicy 3 1/2 cala

Na pokazie Comdex w listopadzie 1988 r. w Las Vegas firma Micropolis, która była czołowym dostawcą dysków o średnicy 5 1/4 cala do stanowisk pracy, wystawiła serię 1700 stacji dyskowych o średnicy 3 1/2 cala. Mają one dwie sformatowane pojemności 112 Mbajtów i 202Mbajty, a czas dostępu wynosi 15 ms.

Kontroliery są tu wbudowane, a sprzęg jest typu albo IBM PC/AT lub SCSI. Cena przy co najmniej 1000 szt. wynosi dla pojemności 202 Mbajtów 935 dolarów. Dostawy są przewidywane w pierwszym kwartale 1989 roku.

Mini Micro Systems nr 10/88

Wytwórcy odpowiedników przyjmują standard IBM

IBM stosuje w swoich drukarkach specjalny język opisu strony o nazwie Intelligent Printer Data Stream (strumień danych inteligentnej drukarki). Przypuszcza się, że protokoły tego języka zostaną włączone do powstającego systemu SAA (Systems Applications Architecture - architektura zastosowania systemów). Od roku język ten przyjęła firma Interface Systems Inc. a obecnie dołączyły się do niej Decislon Data Computer Corp., I-Data A/S, Lexl Computer Systems i Memorex Corp.

Mini Micro Systems nr 10/88

Najnowsza drukarka laserowa HP

Rodzina drukarek LaserJet firmy Hewlett-Packard powiększyła się o model IID, który ma pojemnik na 400 arkuszy papieru, dwa pojemniki papieru, urządzenie do dwustronnego drukowania i zestaw do automatycznego drukowania 50 kopert. Drukarka ta dysponuje 24 wykrojami czcionek, z czego 14 jest wbudowane wewnątrz, a 10 zawartych jest w dołączonej kasie (o symbolu HP 92290S2). Pojemność pamięci wynosi 640 kB i może być rozszerzona do 4,6 MB. Cena wynosi 4295 dolarów, a specjalny pojemnik kopert 350 dol.

PC Tech Journal nr 12/88

System operacyjny do macierzy transputerów

Programiści z Cornell University opracowali kompletny system operacyjny Trylnm do produkowanego przez brytyjską firmę INMOS komputera w jednej kostce zwanego transputerem. System ten pozwala użytkownikom organizować rozwiązywanie problemów na komputerach równoległych składających się z macierzy transputerów.

Transputer projektowany był do łączenia go w macierze dla przyspieszenia przetwarzania złożonych problemów obliczeniowych, przez dzielenie ich na części, które rozwiązywane są jednocześnie. Twórcy systemów równoległych chcieliby łatwo programować komputery aby podzielić program, którego części mogłyby być skutecznie realizowane, a dane przesyłane były wewnątrz procesorów i pomiędzy nimi.

System Trylnm zarządza przetwarzaniem i komunikacją macierzy transputerów oraz dołączonymi do niej komputerami nadawców, a także skomputeryzowanych stanowisk pracy. Ponadto pozwala on programować transputery w takich popularnych językach, jak Fortran i C. Komendy jego są zgodne z systemem Unix.

Nowy system zastosowano po raz pierwszy w komputerach serii T firmy Floating Point Systems z Beaverton. Ośrodek Cornell posługiwał się siecią złożoną z 16 węzłów tych komputerów.

Istnieje duże zainteresowanie systemem Trylnm, zarówno w USA jak i w Europie. Transputery używane są do rozbudowanych stanowisk graficznych, gdzie wykorzystuje się równoległość do przyspieszenia obliczeń graficznych. Wprowadzenie jednak procesora zmiennego przecinka w jednej kostce powinno znacznie rozszerzyć zastosowanie transputerów, jako elementów składowych dużych maszyn.

Computer Age - World Trade nr 4/88

WordStar do Macintosha

Firma MicroPro International z San Rafael w Kalifornii oferuje za 495 dolarów Mac WordStar, który jest nowym opracowaniem, różnym od wersji znanej pod systemem MS-DOS, aczkolwiek ma wszystkie własności WordStar Professional, WordStar 2000 i Microsoft Word, a także wiele cech PageMakera firmy Aldus. Pozwala on nanosić tekst na rysunki, obracać tekst i przesuwając go, a także sprawdzać zasady gramatyczne. Opracowywane są wersje pracujące pod Unixem i OS/2.

Byte nr 9/88

Nowy system operacyjny dla 80386

Theos Software Corp. z Walnut Creek w Kalifornii posiada już handlową wersję systemu operacyjnego THEOS 382, który jest 32-bitowym, wielozadaniowym systemem dla wielu użytkowników komputerów zbudowanych na mikroprocesorze Intel 80386. Firma oferuje też nowy kompilator języka C, który pozwala na wykonywanie programów pisanych w tym języku pod systemami Unix i DOS bez żadnych zasadniczych zmian. Programy pisane dla poprzednich wersji ulegają ponownej kompilacji. System pozwala na pracę 12 zadań z oglądaniem ich na ekranie. System wieloużytkownikowy kosztuje 799 dolarów, dla pojedynczego użytkownika 399, a zestaw do opracowywania oprogramowania 1599 dol.

Mini Micro Systems nr 10/88

Rozwój techniki komputerowej we Włoszech

Co najmniej 135 przedsiębiorstw włoskich pracuje o dziedzinie sztucznej inteligencji. Włochy inwestują tu najwyższy procentowo udział dochodu narodowego. Od 1975 roku wytwarzane są tu roboty do linii montażowych i spośród pięciu rodzajów takich robotów obecnie wykorzystywanych - trzy są włoskie.

Opracowano też dwa prototypy superkomputerów o nazwie APE, z których pierwszy może wykonywać miliard operacji na sekundę i kosztuje 117 tys. dolarów, a drugi jest 4-razy szybszy i kosztuje 325 tys. Pomoc rządowa przy ich opracowaniu wyniosła 1,2 mln dolarów. Niski koszt (dla porównania komputer Cray kosztuje we Włoszech 12 mln dolarów) wynika z faktu, że są to komputery specjalistyczne.

Computer Age - World Trade nr 6/88

Dodatni bilans USA na początku 1988 r.

Styczeń 1988 r. zamknął się nadwyżką 123 mln dolarów według Departamentu Handlu USA w dziedzinie przemysłu komputerowego. Jest to znaczny wzrost po deficycie 222 mln w grudniu 1987 r. (w styczniu 1987 deficyt wyniósł 69 mln).

Wzrost zarówno eksport o 60%, jak i Import - o 42,5%, podczas gdy w całym roku 1987 eksport wzrósł o 21,3%, a Import o 26,7%.

Wzrost zanotowany w styczniu 1988 r. spowodowany był głównie sprzętem i częściami zapasowymi systemów przetwarzania danych, które wykazały wzrost o 63,3%.

Computer Age - World Trade nr 4/88

Zwiększa się rynek używanych mikrokomputerów.

Według danych Narodowego Stowarzyszenia Sprzedawców Komputerów w USA sprzedaż używanych mikrokomputerów osiągnęła w roku 1987 1,2 mld dolarów, a przewidywania na rok 1988 mówią o 1,8 mld. Zapasy tych komputerów stanowią poniżej 6% tej wartości; co wskazuje na znaczne jeszcze możliwości. Główny udział mają tu użytkownicy, którzy sprzedają sprzęt swym przyjacielom i współpracownikom. Przedsiębiorstwa przechodzące na nowocześniejszy sprzęt, proponują zwykle sprzedaż przestarzałych urządzeń swoim pracownikom przed ofertą na rynku. Wskaźnikiem tego zjawiska jest współczynnik wartości (Used Value Index - UVI), który wyraża stosunek średniej ceny sprzętu używanego do nowego. W pierwszym kwartale 1988 r. wzrósł on znowu osiągając najwyższą dotychczas wartość 67,8%, co oznacza tendencję do kupowania używanego sprzętu.

Dla firmy Compaq wynosi on 73,2%, co oznacza wzrost o 3,9% w stosunku do poprzedniego roku o 3,9%. Dla Apple odpowiednio wskaźniki wynosiły 67,2% oraz wzrost o 4,1%, natomiast dla IBM współczynnik ten zmalał o 0,3% osiągając 63,1%. Prawdopodobnie wzrosło on dla tej ostatniej firmy znowu gdy ustabilizuje się rynek PS/2.

Computer Age - World Trade nr 4/88

Wzrost europejskiego rynku usług informatycznych

Zachodnioeuropejski przemysł usług informatycznych obejmujący usługi przetwarzania i sieciowe, oprogramowanie i kompletne systemy osiągnął w 1987 r. 31 mld dolarów i wykazuje wzrost rzędu o 20% rocznie, co daje wartość około 76 mld w 1992 r. Największy udział ma tu Francja, aczkolwiek Włochy wykazują szybszy wzrost. Wśród poszczególnych działów zwiększa się udział oprogramowania i usług specjalistycznych, a także dla usług sieciowych podnoszących wartość (Value-Added Network - VAN).

Computer Age - World Trade nr 5/88

Dobre wyniki w handlu informatycznym USA w I kwartale 1988 r.

Dodatni bilans przemysłu informatycznego USA wzrósł w I kwartale 1988 o 134% w stosunku do roku ubiegłego osiągając 309 mln dolarów. Eksport wyniósł 6,72 mld wzrastając o 37,8%, a Import 6,4 mld (wzrost o 35,1%).

Wypożyczenie biurowe wykazało deficyt 806 mln dolarów, co oznacza wzrost o 42,8% w stosunku do roku poprzedniego. Import wzrósł tu o 29,4% do 1,23 mld (w całym roku 1987 wzrósł tylko o 9%), a eksport o 9,9% do 424 mln dolarów (w 1987 25,1%).

Również w sprzęcie telekomunikacyjnym deficyt wzrósł do 547 mln dolarów, o 13,4% w stosunku do tego samego okresu w poprzednim roku. Eksport wzrósł o 55,2% do 356 mln, a Import o 26,9% do 903 mln.

Nadwyżka w przemyśle komputerowym wyniosła 1,48 mld dolarów, co oznacza wzrost o 35,9% w stosunku do I kwartału 1987. Eksport tu wzrósł o 38,3% do 5,72 mld (w całym roku 1987 wzrósł o 21%). Import wzrósł o 38,9% do 4,24 mld dolarów.

Computer Age - World Trade nr 6/88

Spis treści
rocznika 1988

	nr
BABCZENKO W.: Elementy informatyki w średnich szkołach zawodowych	3-4
BAKOWSKI M.: Mikrokomputer MAZOVIA 1016. Interfejs szeregowy i równoległy	1-2
BOTT Cz., KOPRON A., KWIATKOWSKI J., DZIK K., KOSIEL W., WARDA A.: Napędy dysków elastycznych	5-6
BRZEZIŃSKA I. - zob. POPKO J.	
CHOLEWA W., MOCZULSKI W.: Reprezentacja wiedzy w diagnostycznych systemach doradczych	5-6
DZIK K. - zob. BOTT Cz.	
DZIK K.: Mikrokomputer MAZOVIA 1016. Konfiguracja podstawowa	1-2
GÓRECKI A.: PS/2 nowy standard minikomputerów personalnych	3-4
GRZEGOREK M.: Charakterystyka zintegrowanego systemu komputerowego wspomagania produkcji z użyciem stacji roboczej SIEMENS WS10	5-6
JAWORSKI R., PŁOWIEC K.: Ekran MAZOVII 10/16. Działanie kontrolera i opis BIOS-u	1-2
KOPRON A. - zob. BOTT Cz.	
KOSIEL W. - zob. BOTT Cz.	
KOWALSKI L., PANEK J.: CPBR - Technika komputerowa	3-4
KWIATKOWSKI J. - zob. BOTT Cz.	
MOCZULSKI W. - zob. CHOLEWA W.	
PANEK J. - zob. KOWALSKI L.	
PAPST A.: Projekt COSTCC-wspomagane komputerem nauczanie informatyki na poziomie szkoły wyższej	3-4
PŁOWIEC K. - zob. JAWORSKI R.	
POPKO J., BRZEZIŃSKA I.: Mikrokomputer MAZOVIA 1016. Opis funkcjonalny procesora	1-2
PYZEL D.: Program WYKRES - grafika użytkowa	5-6
RAFF A.: Sprawozdanie z konferencji naukowej "Komputer w kształceniu"	3-4
RYŻKO J.: Nowości techniczne	1-4
WARDA A. - zob. BOTT Cz.	
WOLSZCZAK S.: Minikomputerowy system półprzewodnikowej pamięci operacyjnej DRAM-4MB	3-4

Cena 760.-

ISSN 0239-8044