

- 1) przetwarzanie indywidualne,
- 2) przetwarzanie sekwencyjne¹, czyli partiowe.

Wprowadzone pojęcia rodzajów technik przetwarzania wymagają wyjaśnienia. Przykład zaczerpnijmy spośród prostych czynności administracyjnych, z jakimi spotykamy się na co dzień. Wyobraźmy sobie pracę urzędnika nanoszącego wielkości wpłat z przekazów pocztowych na jakąś kartotekę płatników. W zależności od tego, jaki sposób pracy wybierze urzędnik, będziemy mieli do czynienia bądź z techniką przetwarzania indywidualnego, bądź z techniką przetwarzania sekwencyjnego.

1.1. PRZETWARZANIE INDYWIDUALNE ORAZ SEKWENCYJNE

Przy technice indywidualnej urzędnik bierze po jednym przekazie ze sterty przekazów pocztowych przeznaczonych do przetwarzania. Następnie odszukuje w kartotece płatników kartę ewidencji wpłat o nazwisku, imieniu i adresie identycznym z nazwiskiem, imieniem i adresem z przekazu. Z kolei urzędnik dokonuje odpowiedniej adnotacji na karcie ewidencyjnej i ewentualnie przygotowuje zawiadomienie dla płatnika, że ostatnia rata należności została już spłacona. Po czym urzędnik bierze kolejny przekaz i powtarza wymienione czynności dopóty, dopóki nie zarejestruje wszystkich przekazów.

Przy technice sekwencyjnej urzędnik dzieli swą

¹ Przy bardziej złożonych procesach informacyjnych, dla uzyskania dostatecznie efektywnych rozwiązań, zachodzi konieczność stosowania technik mieszanych, gdzie część programu jest realizowana na podstawie techniki indywidualnej, a część — techniki sekwencyjnej.

pracę na dwa etapy. Pierwszy etap polega na uporządkowaniu sterty przekazów według nazwisk nadawców-płatników w porządku alfabetycznym, następnie w obrębie tych samych nazwisk segreguje według imion, a w obrębie tych samych nazwisk i imion — według adresów. Po zakończeniu porządkowania przekazów urzędnik przystępuje do drugiego etapu swojej pracy. Drugi etap polega na aktualizacji kartoteki i przygotowaniu zawiadomień dla płatników. Urzędnik bierze pierwszy przekaz ze sterty (uprzednio uporządkowanej) i przegląda kolejno kartotekę aż do napotkania karty ewidencyjnej o identycznym z danymi z przekazu — nazwiskiem, imieniem i adresem. Następnie urzędnik sporządza odpowiednią adnotację na karcie ewidencyjnej i ewentualnie przygotowuje zawiadomienie dla płatnika. Po czym pobiera następny przekaz i przegląda kartotekę począwszy od ostatniej karty, na której dokonał poprzedniej adnotacji wpisując odpowiednie dane. Ten zespół czynności urzędnik powtarza aż do wyczerpania wszystkich przekazów.

Omawiając obie techniki, założyliśmy, że kartoteka płatników jest uporządkowana alfabetycznie według nazwisk, imion i adresów. Dla sprawnego działania techniki przetwarzania indywidualnego w omawianym przykładzie, niezbędne jest zaopatrzenie kartoteki w wiele wskaźników (dla grup płatników o tych samych literach początkowych nazwisk), ułatwiających wyszukiwanie potrzebnych kart ewidencyjnych. Natomiast przy technice przetwarzania sekwencyjnego nie ma potrzeby stosowania rozwiązań ułatwiających wyszukiwanie właściwej karty. Dla jednego stosu przekazów (czyli partii dokumentów) kartoteka płatników jest przeglądana tylko jeden raz.

Już pobieżna analiza czasu wykonania pracy wykaże przewagę rozwiązania opartego na przetwarzaniu sekwencyjnym, pod warunkiem odpowiednio dużego zbioru przekazów i odpowiednio dużej ilości pozycji ewidencyjnych (kart) w kartotece płatników. Jednakże nie wszystkie procesy informacyjne mogą być organizowane w sposób sekwencyjny. Jeśli zmodyfikujemy nasz przykład i umieścimy urzędnika z kartoteką płatników przy okienku, do którego zgłaszają się interesanci dla dokonania wpłaty, rozwiązanie przy użyciu sekwencyjnej techniki przetwarzania będzie praktycznie bezużyteczne. Ze względu na całkowitą przypadkowość kolejności zgłaszania się interesantów, zmuszeni będziemy zastosować technikę przetwarzania indywidualnego.

Jeśli jednak skomplikujemy dodatkowo zmodyfikowany przykład, uzupełniając podstawowe zadanie urzędnika (obsługa interesantów przy okienku) obowiązkiem opracowywania miesięcznych sprawozdań o wielkości wpłat i ilości płatników, którzy nie zakończyli jeszcze należnych w danym roku spłat, okaże się, że właściwe będzie opracowywanie sprawozdań okresowych przy użyciu techniki przetwarzania sekwencyjnego. Tak więc, doszliśmy do prostego przykładu uzasadnionego zastosowania mieszanej techniki przetwarzania.

1.2. FORMALIZACJA OPISU CZYNNOŚCI

Przy automatyzowaniu procesów informacyjnych zachodzi konieczność jednoznacznego opisania każdej z czynności występujących w ramach danego procesu

informacyjnego. W odróżnieniu od człowieka, komputerowi nie można wydać polecenia (jeśli pojawi się sytuacja nietypowa) „postąp, jak uważasz za stosowne zgodnie z posiadanym doświadczeniem...”. Teoretycznie, możliwe jest ułożenie programu, który nabierałby doświadczenia i sam organizowałby proces podejmowania decyzji, ale ze względu na wielką pracochłonność zarówno programowania, jak i wykonania tego rodzaju złożonych czynności przez komputer (przy obecnym poziomie techniki cyfrowej), jest to raczej nieopłacalne.

1.3. TECHNOLOGIA PROCESU INFORMACYJNEGO

Przywykliśmy używać terminu *technologia* w odniesieniu do procesów materialnych. Rozróżniamy przy tym dwa znaczenia terminu *technologia*: 1) *technologia produkcji* i 2) *technologia obróbki wiórowej*. Przez *technologię produkcji* wyrobu rozumiemy zbiór jednoznacznie określonych operacji, wykonywanych w strumieniu materiałowo-zasileniowym w pewnej kolejności i prowadzących do uzyskania danego wyrobu. Przez *technologię obróbki wiórowej* (cieplnej, plastycznej lub jakiegokolwiek innej) rozumiemy zasady projektowania procesu wytwórczego opartego na wykorzystaniu danej metody (np. obróbki wiórowej).

W odniesieniu do zautomatyzowanych procesów informacyjnych możemy używać terminu *technologia* w podobnych dwu znaczeniach, jak dla procesów materialnych. W tym miejscu konieczna jest pewna dygresja.

Używając pojęcia *technologia* w odniesieniu do wytwórczych procesów materialnych mamy na myśli takie

procesy, w których nastąpiła daleko posunięta standaryzacja, czy — jak kto woli — typizacja operacji występujących czy to w danym procesie, czy w danej klasie procesów, w zależności od rozumienia słowa technologia. Na obecnym etapie rozwoju elektronicznego przetwarzania danych mamy daleko posuniętą typizację elementów procesów przetwarzania i dlatego — zdaniem autorów — używanie terminu technologia w odniesieniu do zautomatyzowanych procesów informacyjnych jest w pełni uzasadnione.

Posługując się pewnymi analogiami, postaramy się scharakteryzować zautomatyzowane procesy informacyjne i uzasadnić w ten sposób użycie tego terminu. Współczesny komputer (wraz z urządzeniami peryferyjnymi) traktujemy jako wieloczynnościową obrabiarkę zdolną (w zależności od oprzyrządowania) do wykonywania różnorodnych operacji. Warto podkreślić, że słabą stroną tej analogii jest daleko dalej posunięta uniwersalność komputera niż uniwersalność współczesnych obrabiarek wieloczynnościowych. Odpowiednikiem jednego zestawu oprzyrządowania decydującego o rodzaju wykonywanych grup operacji jest program komputera. Odpowiednikiem programu sterowania wieloczynnościowej obrabiarki jest zestaw parametrów dla programu komputera, decydujący o wariancie działania danego programu. Analogonem materiału obrabianego na obrabiarce wieloczynnościowej są wejściowe zbiory danych. Analogonem wyrobu (czy półfabrykatu), uzyskiwanego w wyniku procesu obróbkowego realizowanego za pomocą obrabiarki wieloczynnościowej, są zbiory danych wynikowych, utworzone w rezultacie działania programów z odpowiednimi parametrami na zbiorach danych wejściowych.

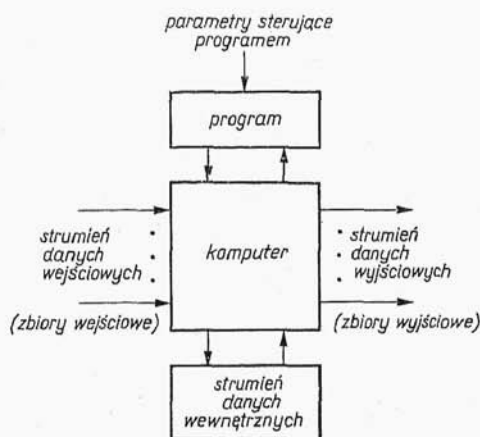
Mówiąc o technologii obróbki wiórowej na obrabiar-
kach wieloczynnościowych, mamy przede wszystkim na
myśli:

- 1) standaryzację (typizację) oprzyrządowania,
- 2) standaryzację (wymiarów, właściwości mechanicz-
nych itd.) materiałów,
- 3) standaryzację półfabrykatów.

Przy technologii zautomatyzowanych systemów infor-
macyjnych, operując wprowadzonymi analogonami,
możemy przetłumaczyć wymienione zagadnienia na ję-
zyk zautomatyzowanych systemów informacyjnych.
W ten sposób otrzymamy:

- 1) standaryzację oprogramowania,
- 2) standaryzację struktury zbiorów wejściowych,
- 3) standaryzację struktury zbiorów wyjściowych.

Te trzy zagadnienia standaryzacyjne możemy roz-
szerzyć o standaryzację parametrów sterujących dzia-
łaniem oprogramowania typowego (patrz rys. 1).



Rys. 1. Schemat procesu przetwarzania

W dalszych rozdziałach książki przedstawimy wymienione problemy standaryzacji dla przetwarzania sekwencyjnego, realizowanego na komputerach wyposażonych w urządzenia pamięci zewnętrznej na taśmach magnetycznych.

PROGRAMOWANIE ZADAŃ PRZETWARZANIA DANYCH

2

Punktem wyjścia do dalszych rozważań jest przedstawienie zarysu specyfiki programowania zadań przetwarzania danych. W toku wykonywania jakiegoś programu przetwarzania danych przez komputer przepływają dwa rodzaje strumieni informacyjnych:

- 1) *strumień rozkazów wykonywanego programu,*
- 2) *strumienie danych przetwarzanych przez komputer płynące z urządzeń zewnętrznych wejściowych (w szczególności pamięci zewnętrznych działających dla danego programu lub jego fragmentu jako urządzenie wejściowe) przez jednostkę centralną (dokładniej mówiąc poprzez pamięć operacyjną i rejestry) do urządzeń zewnętrznych wyjściowych (w szczególności pamięci zewnętrznej działającej dla danego programu lub jego fragmentu jako urządzenie wyjściowe).*

Strumień rozkazów wykonywanego programu jest kształtowany przez sieć działań programu, strumień danych przetwarzanych i parametry sterujące działaniem programu. Na wstępie omówimy typowe elementy sieci działań występujące w programach, następnie pokażemy (na dość elementarnym przykładzie) sieć działań programu jako całość.

Każdą z sieci działań możemy rozłożyć na pewne elementarne części składowe. Każda z części składać się może z jednej lub więcej instrukcji komputera. Wszystkie elementarne czynności wchodzące w skład sieci działań programu podzielimy na dwie klasy: 1) testy i 2) operatory. *Testy* są to elementarne czynności decydujące o wyborze drogi w sieci programu w czasie wykonywania programu przez komputer, *operatory* zaś, to wszelkie pozostałe czynności. Jak już powiedzieliśmy, zarówno testom, jak i operatorom odpowiadają sekwencje instrukcji komputera. W szczególnym przypadku dany test czy dany operator może być realizowany za pomocą jednej instrukcji komputera. Wprowadzimy z kolei dwa pomocnicze skróty:

1) *wejście testu* (operatora), zamiast: pierwsza instrukcja komputera składająca się na realizację danego testu (operatora);

2) *wyjście testu* (operatora), zamiast: instrukcja komputera składająca się na realizację danego testu (operatora) przekazująca sterowanie na zewnątrz danego testu (operatora).

Dla zapewnienia jednoznaczności i dla uniknięcia nieporozumień narzucimy na testy i operatory pewne warunki ograniczające. Mówiąc inaczej, grupy sekwencji instrukcji, na które możemy podzielić dowolny program, będą wtedy i tylko wtedy uważane przez nas za testy i operatory, jeśli spełnią następujące warunki:

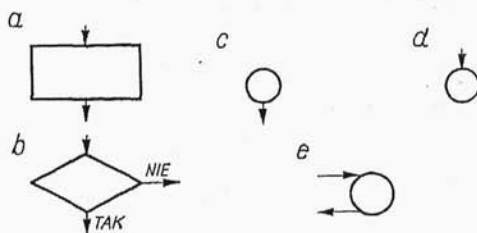
— *uporządkowania testu*; sterowanie z zewnątrz może być przekazane do testu jedynie przez wejście testu;

— *uporządkowania operatora*; sterowanie z zewnątrz może być przekazane do operatora jedynie przez wejście operatora; ponadto, operator może mieć tylko jedno wyjście;

— prostoty operatora; operator może wykonywać tylko jeden rodzaj czynności (albo czynności obliczeniowe, jak wyznaczanie wartości wyrażenia arytmetycznego, albo czynności logiczne, jak wyznaczanie wartości wyrażeń boolowskich, albo czynności manipulowania danymi, jak rozpakowanie, zapakowanie, redagowanie itp., albo czynności przenoszenia danych z jednego obszaru pamięci do innego, czy też z jednego typu urządzenia pamięciowego do innego).

Opracowując graficzne schematy sieci sterowania programu, oznaczamy operatory za pomocą prostokątów (patrz rys. 2a), zaś testy — za pomocą rombów (patrz rys. 2b). Połączenia pomiędzy kolejnymi operatorami i testami oznaczamy za pomocą strzałek. Dla zwiększenia przejrzystości schematów, zamiast strzałek łączących operatory i testy pomiędzy sobą, rysujemy często jedynie ich fragmenty początkowe i końcowe, zaopatrując je etykietami (patrz rys. 2c). Dalej operatory i testy będziemy nazywali funkcjami.

W każdym programie występują pewne typowe rodzaje powiązań pomiędzy funkcjami składającymi się na dany program. Są to tak zwane podsieci programu.



Rys. 2. Oznaczenia używane na schematach blokowych (sieci działań) programów: a — operator, b — test, c — początek podsieci (etykieta), d — koniec podsieci (przekazanie sterowania do etykiety), e — wywołanie podprogramu i powrót przez ślad

2.1. PODSIEĆ LINIOWA

Podsieć liniowa jest najprostszym typem podsieci. Składa się ona jedynie z operatorów. Wyjście pierwszego operatora jest połączone z wejściem drugiego operatora itd. Wyjście przedostatniego operatora jest połączone z wejściem ostatniego operatora (patrz rys: 3).



Rys. 3. Schemat blokowy fragmentu programu o podsieci liniowej

W toku pracy komputera każdy z operatorów wchodzących w skład podsieci liniowej jest wykonywany jeden raz. Teoretycznie, można wyobrazić sobie złożony program o sieci liniowej. Zaletą takiego programu byłaby maksymalna szybkość jego realizacji dla danego komputera. Podstawową jednak wadą byłyby jego rozmiary. Dlatego też dążymy do pisania programów zawierających w sobie części wielokrotnie wykonywane w toku jednej realizacji danego programu.

2.2. PODSIECI Z ROZWIDLENIAMI

Dla opisanía tych fragmentów programu, w których w różnych przedziałach (lub układach warunków) rozwiązania uzyskiwane są za pomocą różnych formuł (lub procedur postępowania), wykorzystujemy podsieci z rozwidleniami. Podsieć taka zawiera test (lub testy) spraw-