

PROJEKTOWANIE STRUKTURALNE PROCEDUR

"Moduł jest całkiem po prostu "czymś" co posiada zdefiniowane granice w stosunku do reszty systemu i dobrze zdefiniowane łącznie do komunikowania się z nią/../. Strukturalne programowanie/w najbardziej potocznym rozumieniu odnosi się do struktury logicznych przepływów w module."

/T. Gilb/

Przed projektowaniem strukturalnym procedur jako główne zadanie stawia się budowę procedur wielokrotnego użytku /powtarzalnych i równocześnie modyfikowalnych/ nadających się do stosowania w różnych pakietach problemowych. Najbardziej pożądane rozwiązanie technologiczne takich procedur polega na wykorzystaniu tzw. modułów wielodostępnych przeznaczonych do równoległego użycia w kilku różnych programach w warunkach pracy wieloprogramowej. Technologia ta w dużej mierze zależna jest od możliwości systemu operacyjnego komputera, który oprócz specjalnie zorganizowanego obszaru pamięci operacyjnej /dostępnego dla różnych programów/ powinien odtwarzać ślady różnych wywołań i stany procedury dla każdej jej realizacji.

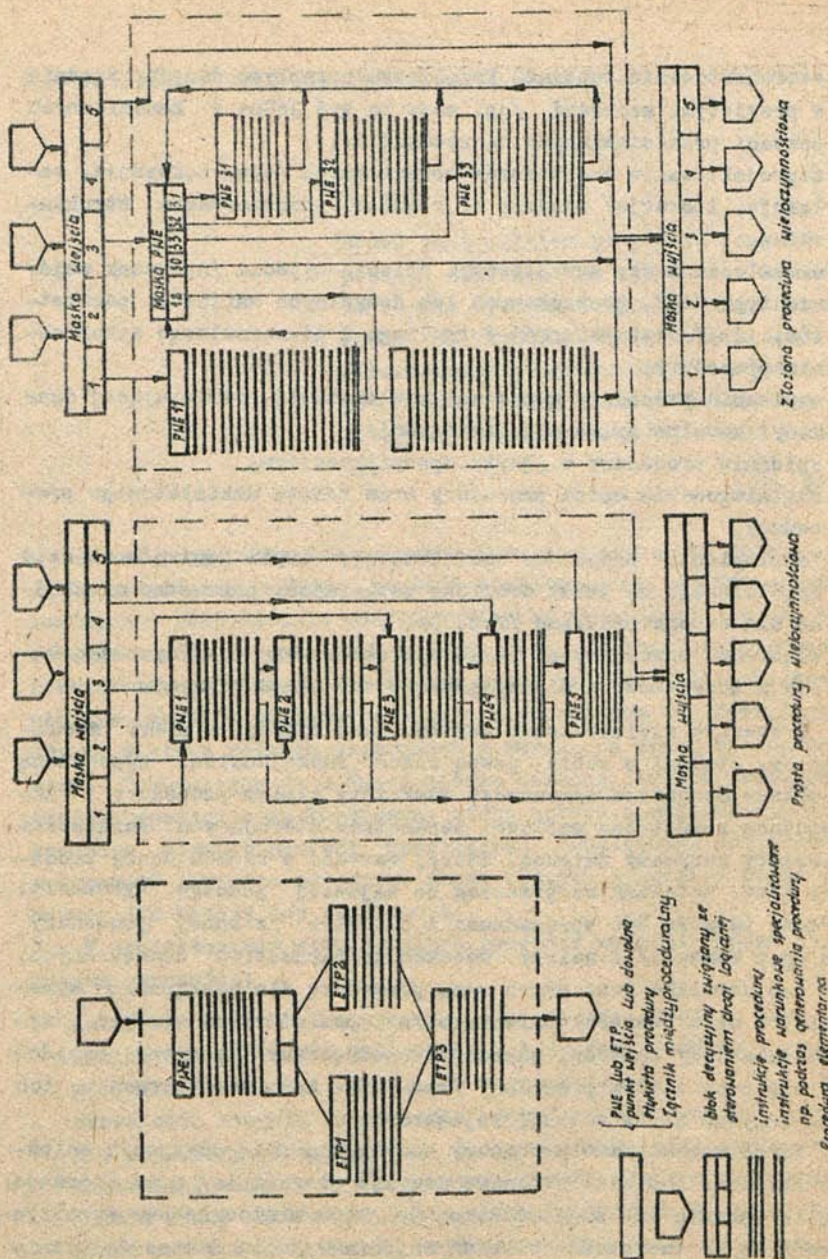
W projektowaniu strukturalnym procedur wyróżnić można następujące czynności:

- ustalenie zakresu funkcjonalnego procedury /takiego aby stanowiła ona względnie zamkniętą całość funkcjonalną z punktu widzenia realizacji określonej czynności/,
- określeniu stopnia uogólnienia procedury i parametrów strojenia procedury. Przykładowo, dla procedury obliczania cyfry kontrolnej mogą to być parametry: ilość cyfr symbolu, modulo, lokata cyfry kontrolnej, początkowa wartość wagi, krok wagi, ilość wag w cyklu, rodzaj wejścia /przedziały liczbowe lub zbiór/,

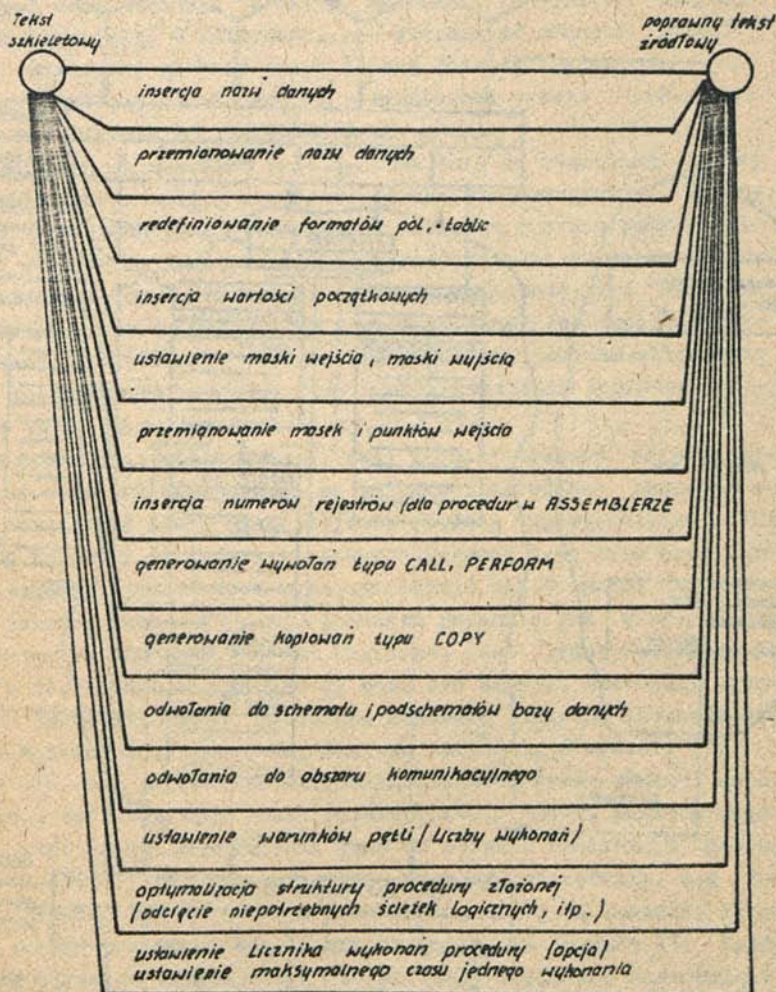
- zaprojektowaniu schematu logicznego przepływu danych zgodnie z przyjętymi wzorcami /np. może to być jedna z kanonicznych postaci przedstawionych na rysunku 8/,
- zaprojektowaniu mechanizmów wewnętrznych /typu sekwencja, selekcja, iteracja/ zgodnie z regułami programowania strukturalnego,
- określeniu łączy zewnętrznych wejścia-wyjścia /np. maszek wejścia typu CASE, początkowych lub domyślnych wartości parametrów, itp./ oraz warunków normalnego i nienormalnego zakończenia procedury,
- napisaniu procedury w postaci szkieletowej /zawierającej dane modyfikowalne zwane modyfikatorami/,
- opisanu procedury w języku specyfikacyjnym,
- skatalogowaniu opisu procedury oraz tekstu szkieletowego procedury,
- "odślepieniu" procedury szkieletowej, czyli zamianie tekstu uogólnionego na tekst źródłowy zawierający parametry niezbędne dla danego problemu /rys. 9/.
- włączeniu procedury do pakietu problemowego wykorzystując łączy międzybazowe i międzyjęzykowe zawarte w prasystemie.

Procedura wielokrotnego użytku, jako typowy element "strukturalny, stanowi w sobie pewną całość funkcjonalną, wyposażoną we własny mechanizm wykonawczy zamknięty ciałem procedury i akceptującą zewnętrzne wartości parametrów sterujących. Parametrem procedury nazywamy zmienną, której wartość w ramach danej procedury jest ustalana na przeciąg co najmniej jednego wykonania. Wartość ta może być wprowadzana z zewnątrz /z innej procedury lub np. z konsoli/ i należy wówczas do parametrów zewnętrznych, lub też ustalana jest przez samą procedurę dla następnego wykonania. W celu uniezależnienia stanu procedury od miejsca wywołania adresy wartości parametrów zewnętrznych powinny znajdować się poza ciałem procedury /rys. 10/, zaś przekazywanie ich może odbywać się w notacji rejestrowej.

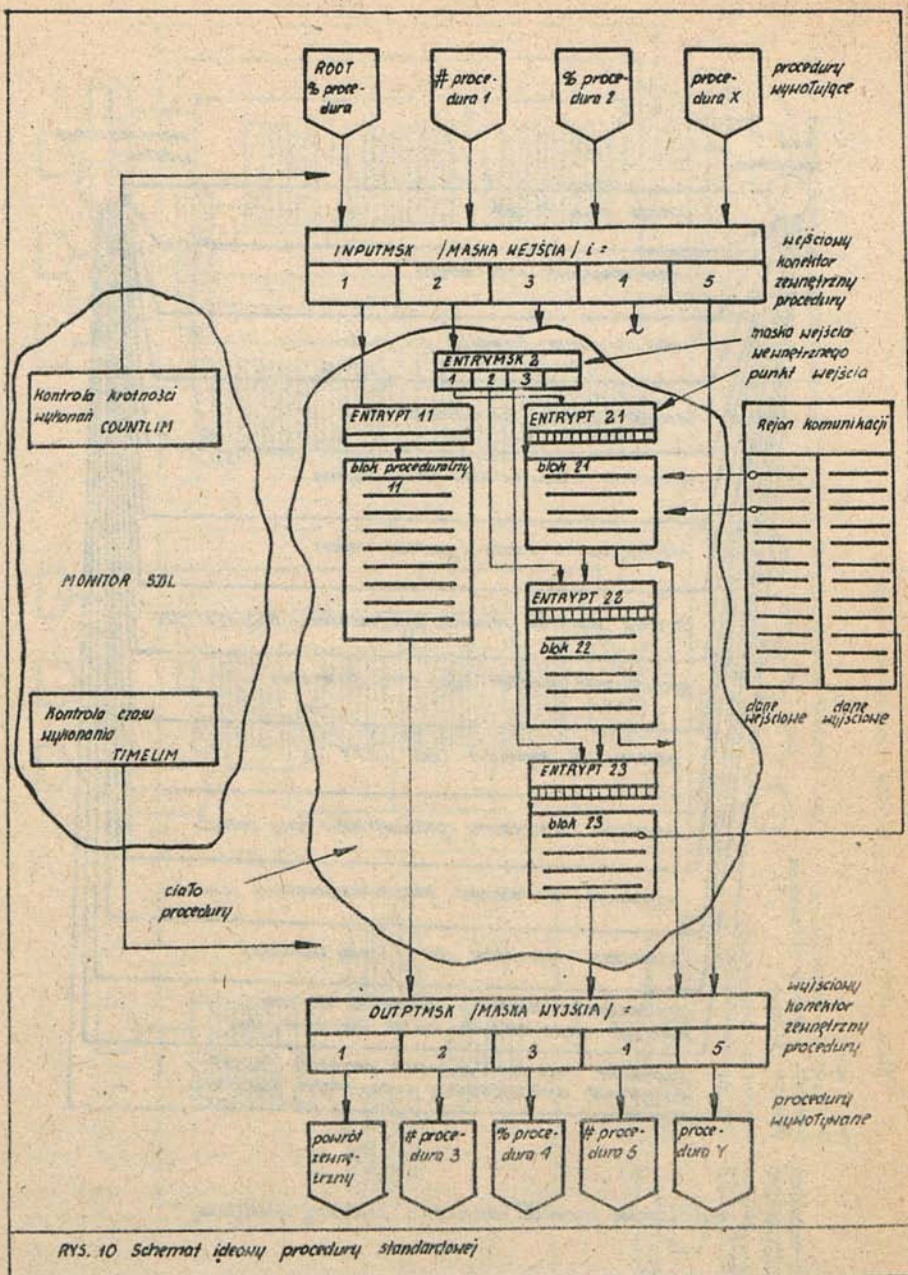
Wśród mechanizmów procedury szczególną rolę odgrywają selekcje /binarne lub wieloczołonowe decyzje sterowania/ oraz iteracje /pętle/ proste lub zagnieżdżone. Selekcje wieloczołonowe wykorzystywane są do tworzenia "bloków" wejścia-wyjścia. Zakres działania



Rys. 8. Kanałowa postać procedur wielokrotnościowych



RYS.9 Przykładowe czynności "odstępiania" procedury szkieletowej



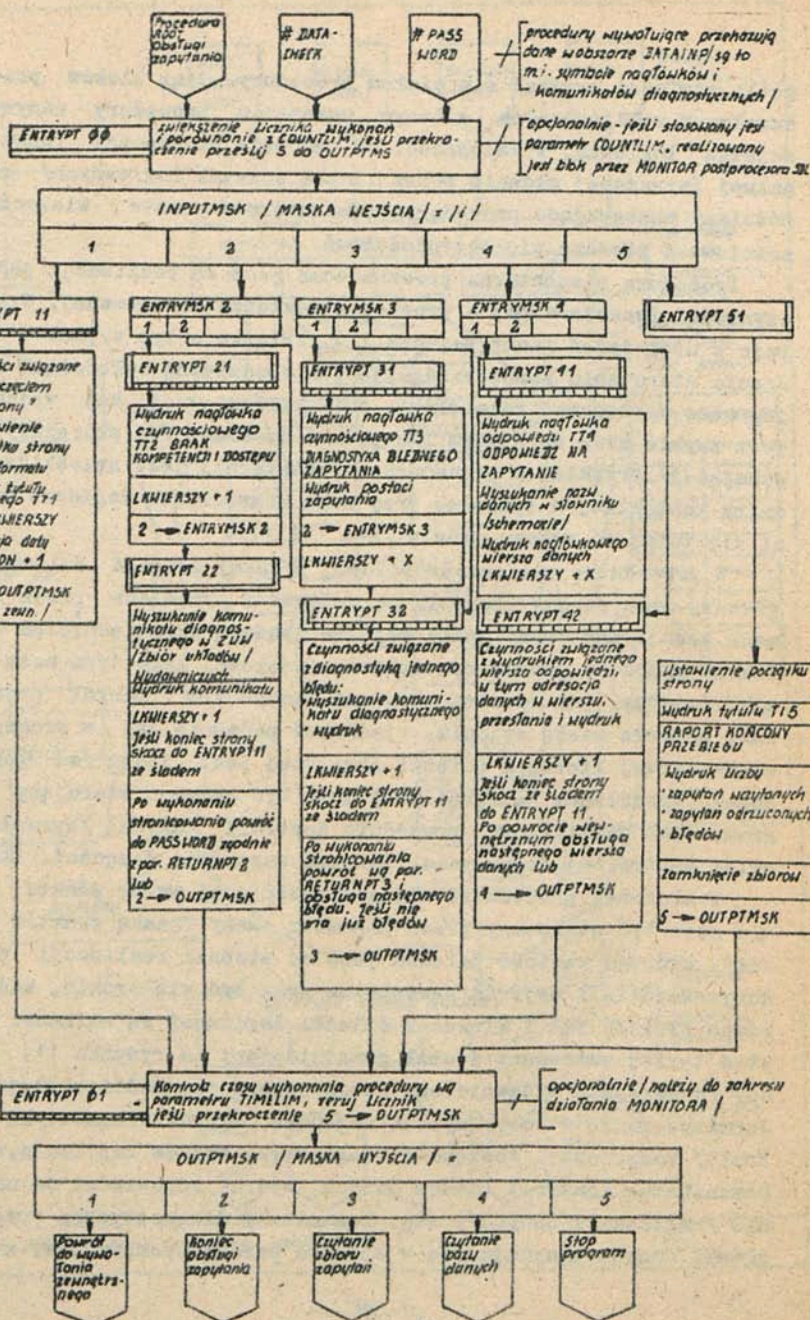
RYS. 10 Schemat ideowy procedury standardowej

pętli powinien zamykać się ciałem procedury. Ciąg bloków proceduralnych realizowanych w danym wykonaniu procedury nazywamy ścieżką logiczną. W zależności od stopnia złożoności funkcjonalnej /wyrażanej głównie przez liczbę ścieżek logicznych/ różniamy następujące procedury: elementarne, proste wieloczynnościowe i złożone wieloczynnościowe.

Procedura elementarna przeznaczona jest do realizacji jednej czynności funkcjonalnej o prostej strukturze wewnętrznej. Występuje w niej jeden punkt wejścia i w związku z tym z punktu widzenia sterowania zewnętrznego jest niepodzielną całością. W ciele jej może występować rozwidlenie ścieżki logicznej, zaś wyjściem jest zwykle skok ze śladem lub powrót do śladu /do procedury wywołującej/. Przykładem procedury elementarnej jest np. obliczenie cyfry kontrolnej w symbolu, przy czym rozwidlenie logiczne odnosi się tutaj do wyboru modulo /np. 7 i 11/.

W procedurze wieloczynnościowej prostej długość ścieżki logicznej oraz punkty wejścia są zmienne i określane przez parametr zewnętrzny zwany maską wejścia /będącą odpowiednikiem CASE/. Zakres rzeczowy procedury wieloczynnościowej tego typu może być więc zmienny, jak również może ona łączyć się z innymi procedurami poprzez maskę wyjścia, jednakże podobnie jak w procedurze elementarnej kierunek ścieżki logicznej jest jeden /od góry do dołu/. Przykładem tej procedury może być segment sterujący programu lub kompleksowa procedura kontroli danych /sprawdzenie cyfry kontrolnej, typu pola, zakresu wartości, długości, itp./.

W złożonej procedurze wieloczynnościowej wybór punktu wejścia może być sterowany przez parametr zwany "maską punktów wejścia", którego wartość zależna jest od stopnia realizacji procedury. Wejścia i wyjścia zewnętrzne mogą być wielorakie, zaś zarówno długość jak i kierunek ścieżki logicznej są zmienne. Przykład takiej procedury został przedstawiony na rysunku 11. Dotyczy on agregatu wydawniczego obejmującego czynności następujące: formatyzowanie strony, wydruk nagłówka, wydruk diagnostyki /kontroli/ kompetencji dostępu i diagnostyki błędów zapytania, wydruk komunikatów kontroli błędów danych, wydruk odpowiedzi na zapytanie /nagłówka i danych/, itp. Komunikaty diagnostyczne oraz nagłówki danych znajdują się w zbiorze parametrycznym ZUW, skąd są



RY.15 Schemat konstrukcji czynnościowej procedury wydruku # REPORTIER raport wydawniczy

Tablica 3. Przykładowa lista parametrów konstrukcyjnych do strojenia procedur

Lp.	Umowna nazwa parametru	Wartość parametru	Operacje	Uwagi
1.	INPUTMSK	nazwa danej literał etykieta wyrażenia nr rejestru	przemianowanie nadanie wartości wskazanie adresu " źródła	maska wejścia /CASE/.
2.	OUTPUTMSK	nazwa danej literał etykieta wyrażenia etykieta procedury następnej nr rejestru	"	maska wyjścia
3.	ENTRYPTIK	"	"	maska punktów wejścia ścieżki logicznej "i"
4.	COUNTLIM	liczba całkowita	nadanie wartości	liczba wykonan /maks./
5.	TIMELIM	gg, mm, ss	"	dopuszczalny czas wykonania
6.	DATACOM	nazwa obszaru nr rejestru	wskazanie adresu	obszar danych globalnych do komunikacji pomiędzy procedurami
7.	LANG	ASS, COB, PLI, FOR...		język programowania
8.	LIB	SL, RL, CL ...		rodzaj biblioteki

przywoływane stosownie do potrzeb. Parametry sterujące są przekazywane przez procedury wywołujące. Opis tych parametrów zawiera tablica 3, która odnosi się do działu /należącego do języka umownego SDL/ przeznaczonego do opisu montażu procedur. Opis struktur danych w tym języku przedstawiliśmy w rozdziale 7.4. Projektowanie struktur danych.

Łączność parametryczna pomiędzy modułami organizowana może być poprzez wywołanie typu CALL, wbudowane funkcje adresowe np. w PL/1/, globalny obszar komunikacyjny do którego odwołują się wszystkie powiązane procedury /przy czym dla niektórych języków nieodzowne może być tutaj pośrednictwo podprocedury w języku typu assembler oraz zbiór zewnętrzny parametrów w pamięci dyskowej. Bloki zdań źródłowych mogą być sprowadzone z bibliotek za pomocą zwrotu COPY, przy czym w przypadku języka COBOL dotyczyć to może na przykład zwrotów opisu danych. Poza parametrami na poziomie danych i na poziomie sterowania przepływem logicznym wystąpić mogą przemianowania nazw stanowiących punkty wyjścia i nazwy procedur otwartych.