

C  
45333  
Biblioteka Główna  
Politechnika Warszawska

CENTRUM OBLICZENIOWE  
POLSKIEJ AKADEMII NAUK

MIECZYŚLAW WARMUS

# GIER-ALGOL

WARSZAWA 1966  
PAŃSTWOWE WYDAWNICTWO NAUKOWE

CENTRUM OBLICZENIOWE  
POLSKIEJ AKADEMII NAUK

MIECZYŚŁAW WARMUS

# GIER-ALGOL

WARSZAWA 1966  
PAŃSTWOWE WYDAWNICTWO NAUKOWE

C.45333

Printed in Poland

Państwowe Wydawnictwo Naukowe  
Oddział w Łodzi 1966

Wydanie I. Nakład 1200 + 100 egz. Ark. wyd. 13,25. Ark. druk. 16,75. Papier  
offsetowy kl. III, 80 g, 70 x 100. Oddano do druku 16. IV. 1966 r. Druk  
ukończono w maju 1966 r. Zam. nr 40. N-9. Cena zł 42,-

Zakład Graficzny PWN  
Łódź, ul. Gdańska 162

## PRZEDMOWA

Niniejszy podręcznik GIER-ALGOLu powstał ze skryptu pod podobnym tytułem, wydanego w 1965 r. przez Centrum Obliczeniowe Polskiej Akademii Nauk i Państwowe Wydawnictwo Naukowe. W porównaniu ze skryptem niniejsze opracowanie zawiera znacznie więcej zadań i przykładów, wykład został w wielu miejscach uzupełniony lub poprawiony, ponadto usunięto różnego typu drobne usterki i niedopatrzenia.

Wspomniany skrypt był pierwszym oryginalnym polskim opracowaniem języka ALGOL, jakie ukazało się na rynku księgarskim. Wkrótce po nim wydano jeszcze dwie książki:

1. Stefan Paszkowski; Język ALGOL-60, Państwowe Wydawnictwo Naukowe, Warszawa 1965.

2. Jan Madey; ALGOL-60 - GIERALGOL III, Wydawnictwa Uniwersytetu Warszawskiego, 1965.

Obie wspomniane publikacje różnią się od tego podręcznika bardzo wyraźnie zarówno układem treści, jak i metodą wykładu. Posługują się one m.in. tzw. metajęzykiem, który dla niematematyków jest najczęściej trudny. Dlatego też wspomniane wyżej książki są przeznaczone głównie dla czytelników oswajanych z matematycznym sposobem myślenia. Niniejsze opracowanie ma na celu uprzystępnienie programowania w języku ALGOL szerokim rzeszom czytelników o wykształceniu technicznym, ekonomicznym, przyrodniczym czy humanistycznym, poprzez maksymalne uproszczenie wykładu, niewnikanie w subtelności natury logicznej, nie mające znaczenia dla praktyki obliczeniowej, wręcz staranny dobór przykładów i zadań.

Książka niniejsza obejmuje pełny wykład języka ALGOL-60, a ponadto cały dodatkowy materiał potrzebny do praktycznego programowania w języku GIER-ALGOL na maszynach typu GIER. Sporządzenie translatora tego języka dla maszyn typu URAL-2 rozszerza znacznie krąg jego użytkowników.

Tak jak poprzedni skrypt, książka niniejsza oparta jest na dwu podstawowych publikacjach:

1. J.W. Backus, F.L. Bauer i inni: Revised Report on the Algorithmic Language ALGOL-60, Regnecentralen, Copenhagen 1962.

2. H. Christensen, G. Ehrling i inni: A Manual of GIER ALGOL III, Regnecentralen, Copenhagen 1964.

Język GIER-ALGOL jest jedną z najobszerniejszych reprezentacji języka ALGOL-60. Do istotnych różnic należą:

- w języku GIER-ALGOL nie ma deklaracji own array,
- liczby naturalne nie mogą być w języku GIER-ALGOL etykietami,
- wszystkie parametry formalne w procedurach GIER-ALGOLu muszą być specyfikowane,
- w procedurach GIER-ALGOLu nie można wywoływać tablic przez wartość,
- GIER-ALGOL zawiera około 30 procedur standardowych (głównie procedury wejścia i wyjścia), których nie ma w ALGOLu, ponieważ ALGOL-60 pozostawił wprowadzanie danych i wyprowadzanie wyników jako sprawy otwarte.

Pozostałe różnice dotyczą szczegółów mniej istotnych. W tekście książki różnice między GIER-ALGOlem a ALGOlem-60 są szczegółowo wymieniane.

Składam podziękowanie Państwowemu Wydawnictwu Naukowemu za staranne wydanie niniejszej książki.

Warszawa, wrzesień 1965 r.

*Mieczysław Warmus*

## O. WSTĘP

### O.1. Wiadomości wstępne o programowaniu

Rozwiązywanie problemów obliczeniowych polega na:

- sformułowaniu matematycznym problemu,
- ustaleniu metod numerycznych i wybraniu metody najdogodniejszej dla danego wykonawcy,
- programowaniu obliczeń,
- wykonaniu obliczeń,
- kontroli wyników.

Niniejsze opracowanie poświęcone jest programowaniu obliczeń dla maszyn cyfrowych.

Programowanie takie obejmuje:

- ustalenie algorytmu czyli następstwa działań rachunkowych,
- ustalenie sposobu wprowadzania danych i wyprowadzania wyników,
- ustalenie metody kontroli zarówno dla samych obliczeń, jak i dla wprowadzania danych i wyprowadzania wyników,
- kodowanie programu czyli zapis programu w języku zrozumiałym dla maszyny,
- kontrolę programu.

Przez program rozumiemy tutaj zakodowany algorytm wraz z zakodowanym sposobem wprowadzania danych i wyprowadzania wyników oraz z zakodowaną metodą kontroli zarówno obliczeń, jak wprowadzania danych i wyprowadzania wyników.

Elektronowe maszyny cyfrowe są tak skonstruowane, że "rozumieją" przekazywane im informacje w pewnym specjalnym języku, zwanym kodem wewnętrznym maszyny. Kody wewnętrzne dla różnych typów maszyn są zazwyczaj różne. Dlatego też program napisany w kodzie wewnętrznym dla jednej maszyny jest na ogół nieprzydatny dla maszyn innych typów.

Programowanie w kodzie wewnętrznym maszyny jest pracochłonne. Dlatego użytkownicy programują zazwyczaj w języku dla nich wygodniejszym, który nazywa się wtedy kodem zewnętrznym albo językiem zewnętrznym, a gotowe programy są tłumaczone na kod wewnętrzny maszyny. Jeśli kod zewnętrzny jest tak skonstruowany, że informacje w nim zapisane mogą być przez maszynę odczytane, a następnie przy pomocy specjalnego programu automatycznie w maszynie przetłumaczone na kod wewnętrzny, to taki kod zewnętrzny nazywa się autokodem, a program tłumaczący go - t r a n s l a t o r e m. Jeden i ten sam język zewnętrzny może być autokodem dla wielu typów maszyn; wystarcza do tego skonstruowanie odpowiednich translatorów (dla każdego typu maszyn oddzielnie, ponieważ translatory są konstruowane w kodzie wewnętrznym i ich budowa zależy od typu maszyny). Ogólnie dążymy do jak najbardziej uniwersalnych i powszechnych języków zewnętrznych, wspólnych dla możliwie wielu typów maszyn.

Stosowanie autokodów skraca znacznie pracę programistów, ale na ogół przedłuża czas pracy maszyny i mniej ekonomicznie wykorzystuje jej pamięć. Dlatego zdarza się nieraz - zwłaszcza w przypadku maszyn o ograniczonych możliwościach - że wolimy programować w kodzie wewnętrznym, pomimo istnienia autokodu, aby w pełni wykorzystać możliwości maszyny. Natomiast z reguły używamy autokodów, gdy chcemy wyniki obliczeń otrzymać szybko, a ze względu na charakter zagadnienia nie zależy nam na pełnym wykorzystaniu możliwości maszyny. Na wybór kodu bez wątpienia ma wpływ przewidywana wielokrotność użycia programu.

Przykładem języka zewnętrznego może być tradycyjny język matematyczny. Ma on jednak pewne właściwości, które bardzo utrudniałyby uczynienie zeń autokodu dla jakiejś maszyny. Można by tu przede wszystkim wymienić:

- wielopoziomowy zapis (ułamki z kreską ułamkową, ułamki tzw. piętrowe, wykładniki potęg, które ponadto same jeszcze mogą być potęgami, indeksy przy zmiennych),
- występowanie znaków, których odczytanie automatyczne byłoby trudne (symbole pierwiastków, nawiasy różnej wielkości i różnych kształtów).

Ponieważ jednak jesteśmy bardzo przyzwyczajeni do tradycyjnego języka matematycznego, staramy się konstruować autokody tak, by były w miarę możliwości podobne do języka tradycyjnego.

Język zewnętrzny powinien być stosowany przez jak najszersze kręgi użytkowników, ponieważ tylko wtedy może być językiem komunikatywnym, językiem, w którym można pisać publikacje, przekazywać programy z jednych maszyn na drugie.

Języków zewnętrznych skonstruowano już wiele. Jednym z najbardziej rozpowszechnionych jest np. język FORTRAN, będący autokodem dla maszyn amerykańskich IBM, zajmujących pod względem liczebności pierwsze miejsce w świecie. Język ten jednak prawie że nie wyszedł poza krąg użytkowników maszyn IBM, a nie znalazł żadnego szerszego zastosowania w krajach socjalistycznych.

Językiem, który zyskał najwięcej zwolenników, zarówno w krajach naszego obozu, jak i w krajach kapitalistycznych, jest język ALGOL-60, którego nazwa pochodzi od pierwszych liter wyrazów angielskich ALGOrithmic Language (tzn. język algorytmiczny) z dodaniem roku, w którym ten język powstał (1960). Rozpowszechnienie tego języka wzrasta z roku na rok, tak że nawet koncern IBM był zmuszony wprowadzić translatory ALGOLu dla swych maszyn.

## 0.2. Wiadomości wstępne o ALGOLu

Język ALGOL-60 powstał w wyniku kooperacji kilkunastoosobowej międzynarodowej grupy specjalistów z zakresu eksploatacji maszyn cyfrowych. W skład tej grupy wchodził przede wszystkim specjalista duński i zachodnioniemiecki, ale do sprecyzowania ostatecznej wersji ALGOLu przyczyniło się bardzo wielu innych.

Wersja ta pod nazwą ALGOLu-60 została m.in. zaakceptowana oficjalnie przez IFIP (International Federation for Information Processing) i została opublikowana pod tytułem: "Re-



vised Report on the Algorithmic Language ALGOL-60" (Regnecentralen, Copenhagen, 1962, jak również w czasopiśmie "Numerische Mathematik", 4. Band, 5. (Schluss-/Heft, 1963). W tej publikacji język ALGOL-60 jest opisany w tzw. metajęzyku i choć stanowi ona stale podstawę do innych opracowań, jest za trudna dla wstępnego zapoznania się z ALGOlem, zwłaszcza dla niematematyków.

W języku ALGOL posługujemy się wyrazami angielskimi. Jest to uzasadnione ze względu na międzynarodowy charakter ALGOLu i krótkość stosowanych wyrazów angielskich, a nie stanowi większych trudności dla osób nie znających angielskiego, ponieważ liczba używanych w ALGOLu wyrazów angielskich jest niewielka (kilkadziesiąt).

Przez język ALGOL-60 rozumiemy w niniejszym opracowaniu język wzorcowy, służący za podstawę dla tzw. konkretnych reprezentacji tego języka, tworzonych dla konkretnych maszyn i uwzględniających indywidualne cechy maszyn. Przez język GIER-ALGOL rozumiemy konkretną reprezentację języka ALGOL-60 skonstruowaną dla maszyn duńskich GIER. Różnice między ALGOlem-60 a GIER-ALGOlem będą w niniejszym opracowaniu dokładnie omawiane. Gdy będzie mowa po prostu o ALGOLu, będzie to oznaczać, że mówimy o rzeczach wspólnych dla języków ALGOL-60 i GIER-ALGOL. Gdy będzie mowa o GIER-ALGOLu, będzie to oznaczać, że mówimy o rzeczach charakterystycznych dla języka GIER-ALGOL, które w języku ALGOL-60 bądź tu nie występują, bądź występują ale z pewnymi różnicami.

O p r a c o w a n i e n i n i e j s z e s t a n o w i w y k ł a d GIER-ALGOLu, którego translator posiadają w Polsce maszyny typu GIER i URAL-2. Zakładamy, że urządzeniami wejścia i wyjścia są: flexowriter (czytaj fleksorajter; urządzenie podobne do dalekopisu) czytający i perforujący taśmę papierową 8-kanalową oraz maszyna do pisania, podłączona bezpośrednio do maszyny cyfrowej. O korzystaniu z tych urządzeń będzie mowa w końcowych rozdziałach.

### 0.3. Wiadomości wstępne o programach pisanych w ALGOLu

Podstawowym elementem programu napisanego w języku ALGOL jest i n s t r u k c j a (statement; czytaj: stejtment), która określa, jakie operacje ma wykonać maszyna. Najczęściej spotykanym typem instrukcji jest i n s t r u k c j a p o d s t a w i e n i a (assignment statement; czytaj: esajjment), którą zapisujemy następująco:

nazwa zmiennej: =      wyrażenie

np.

a:= x + y;

Instrukcja podstawienia oznacza, że wartość napisanego wyrażenia należy obliczyć i podstawić na zmienną wymienioną po lewej stronie znaku :=. Należy tu podkreślić, że := uważamy za jeden znak. Koniec instrukcji sygnalizuje na ogół średnik. W y r a ż e n i a c h (expressions; czytaj: ekspreszjns) mamy do czynienia z operacjami wykonywanymi na l i c z b a c h (numbers; czytaj: nambers) i z m i e n n y c h (variables; czytaj: weriebłs).



W y r a z e n i a mogą być a r y t m e t y c z n e (arithmetic expressions; czytaj: aritmetik ekspreszyns), jak również logiczne czyli b o o l e o w s k i e (czytaj: bułowskie; po angielsku Boolean expressions; czytaj: bułan ekspreszyns). Analogicznie, zmienne mogą być liczbowe, jak również booleowskie. Aby maszyna mogła rozpoznać typ zmiennej, wprowadzamy ją za pomocą tzw. d e k l a r a c j i t y p u (type declaration; czytaj: tajp deklarejszyn). Istnieją w ALGOLu trzy typy zmiennych: real (rzeczywiste; czytaj: rijal), integer (całkowite; czytaj: intidża) i Boolean (booleowskie; czytaj: bułan). Zmienne całkowite (integer) wyodrębniamy, ponieważ działania na liczbach całkowitych są w maszynie wykonywane bezbłędnie, tzn. nie są obciążone błędami zaokrągleń, a ponadto szybciej niż działania typu real.

Instrukcje piszemy kolejno i łączymy w tzw. b l o k i (blocks; czytaj: bloks), stanowiące określone części programu. Blok rozpoczyna się wyrazem begin (tzn. zacznij; czytaj: bigin), a kończy wyrazem end (tzn. koniec; czytaj: end). Bloki mogą występować jeden w drugim. Całość programu jest też blokiem.

A oto prościutki przykład programu napisanego w języku ALGOL:

```
begin real a,b,c,x;

a:= 1;

b:= 2x+3;

c:= (a+6.1)/(b+5)-0.15;

a:= 1.2x;

b:= x:= c:= a+b+c;

end;
```

W przykładzie powyższym symbol x oznacza mnożenie, a symbol / dzielenie.

W wyniku obliczeń według powyższego programu otrzymujemy

$$a = 1.2, \quad b = 6.76, \quad c = 6.76, \quad x = 6.76,$$

co z łatwością sprawdzamy.

Na tym kończymy rozdział wstępny i przechodzimy do systematycznego wykładu języka GIER-ALGOL.