

Aplicația Pseudocod. Utilizare

BREVIAR TEORETIC

LIMBAJUL PSEUDOCOD

În cadrul Universității "Stefan cel Mare" Suceava se utilizează, începând din 1986, un metalimbaj numit în continuare PSEUDOCOD.

Un algoritm este reprezentat în acest metalimbaj printr-o listă de enunțuri, dintre care se deosebesc două tipuri de enunțuri:

- standard și
- nestandard.

ENUNȚURI STANDARD

În această categorie sunt cuprinse enunțurile care reprezintă cele patru *operații de bază* din cadrul oricărui limbaj de programare

- citirea,
- atribuirea,
- scrierea,
- oprirea,

precum și reprezentarea a două structuri de control

- decizională și
- repetitivă.

Tot în categoria enunțurilor standard se include și *declararea variabilelor*.

Variabila nr trebuie declarată imediat după start sub forma:

```
int <lista variabile>
```

```
float <lista variabile>
```

```
char <lista variabile>
```

```
<lista variabile> := <id variabilă> |
```

```
<lista variabile> <id variabilă>
```

```
int i, n, indice, vector[7];
float x[10], p, a, matrice[10][5];
```

Exemplu:

```
int nr
```

REPREZENTAREA OPERAȚIILOR DE BAZĂ

Citirea

Este o operație prin care calculatorul preia datele utilizatorului prin intermediul unui dispozitiv de intrare (DI), de exemplu tastatura calculatorului.

```
<citire> := citește <lista de intrare>

<lista de intrare> := <id variabilă> | <lista de intrare>,
                    <id variabilă>
```

Atribuirea

Această operație permite modificarea valorilor unei variabile în timpul execuției unui algoritm.

Prin <expresie> se acceptă orice tip de expresie aritmetică sau logică realizată cu constante, variabile, operatorii aritmetici +, -, *, /, ^ (ridicare la putere), parantezele (,) cât și funcțiile remarcabile cunoscute.

Sintaxa

```
<atribuire> :=<id variabila> = <expresie>

d = (b*b - 4*a*c)^0.5
x1= (-b - d)/(2*a)
x2= (-b + d)/2/a
i = i+1
```

Scrierea

Prin această operație se pot face cunoscute utilizatorului o serie de rezultate ca urmare a execuției algoritmului. Aceste rezultate sunt presupuse ca fiind afișate la dispozitivul de ieșire (DE) standard care poate fi ecranul, imprimanta ș.a.

Sintaxa

```
<scrierea> := scrie <lista de iesire>
<lista de iesire> := <element> | <lista de iesire>,<element>
<element> := <expresie> | <constanta alfanumerică>
<constanta alfanumerică> := "<șir de caractere>"
```

```

citeste lungime, lățime
a= lungime * lățime
scrie "Aria =", a, " Perimetru =", 2*(lungime + lățime)

```

Oprirea

```
<oprire> := stop
```

Structura secvențială

Este formată dintr-un grup de operații simple și/sau alte structuri de control ce se execută în mod secvențial.

```

int a, b, suma
scrie "\nIntrodu doua valori: "
citeste a, b
suma = a+b
scrie "\nsuma = ", suma
stop

```

Structura decizionala

Sunt posibile următoarele forme de structură decizionale care exprimă o decizie ce trebuie luată în cursul execuției unui algoritm

Forma I

```
dacă <condiție> atunci <operație_de_bază> n
```

Forma II

```

┌ dacă <condiție> atunci
│   <secvență>
│
└ n

```

Forma III

```

┌ dacă <condiție> atunci
│   <secvența_1>
│   altfel
│   <secvența_2>
└ n

```

PROBLEME REZOLVATE

Să se scrie un program care cere utilizatorului un număr și apoi afișează dacă acel număr este negativ sau pozitiv.

```

start
int nr
scrie "Dati un numar si va voi spune daca este pozitiv sau negativ "
citeste nr
daca nr < 0 atunci

```

```

    scrie "\n Numarul este negativ"
altfel
    scrie "\n Numarul este pozitiv"
@
scrie "\n Daca ati uitat cumva, ati introdus numarul", nr
stop

```

Rezolvarea ecuatiei de gradul I.

```

float A, B
scrie "\nRezolvare ec gr. I: Ax + b = 0"
scrie "\nIntrodu A, B: "
citeste A, B
daca A == 0 atunci
    daca B == 0 atunci
        scrie "\nO infinitate de solutii"
    altfel
        scrie "\nFara solutii"
@
altfel
scrie "\nX = ", -B/A
@
stop

```

PROBLEME PROPUSE SPRE REZOLVARE

1. Să se scrie un program care cere utilizatorului nota la examen și afișează dacă aceasta a promovat sau nu.
2. Să se scrie un program care oferă utilizatorului 3 opțiuni de a petrece timpul, de exemplu:
 - 1 - la școală
 - 2 - acasă
 - 3 - la plimbare
 iar la introducerea opțiunii să i se afișeze un mesaj corespunzător.
3. Să se scrie un program care rezolvă ecuația de gradul I, luând în considerare și situațiile speciale.
4. Să se scrie un program care afișează valoarea unei funcții într-un punct. Funcția este:

$$f(x) = \begin{cases} x+10, & \text{pentru } x < -2 \\ 2*(x^2+8), & \text{pentru } -2 \leq x \leq 10 \\ x-20, & \text{pentru } x > 10 \end{cases}$$

5. Să se scrie un program care rezolva ecuația de gradul II, luând în considerare și situațiile speciale.

6. Să se scrie un program care citește laturile unui triunghi, face verificările dacă ele pot forma un triunghi, iar apoi afișează tipul de triunghi.