

Bazele programării I

Structuri de control. Structura alternativă

Structura alternativă

Exemplu: determinarea perimetrului unui triunghi cu lungimile laturilor a , b și c .

Condițiile problemei:

1. laturile trebuie să fie strict pozitive;
2. trebuie ca fiecare din ele să fie strict mai mic decât suma celorlalte două.

Tipuri de construcții
alternative

construcția
IF

construcția
Case

Construcția IF

Formele construcției IF

construcția
If - Then - Else

construcția
If - Then

Formatul construcției If-Then-Else:

If condiție **Then**

Secvența_1

Else

Secvența_2

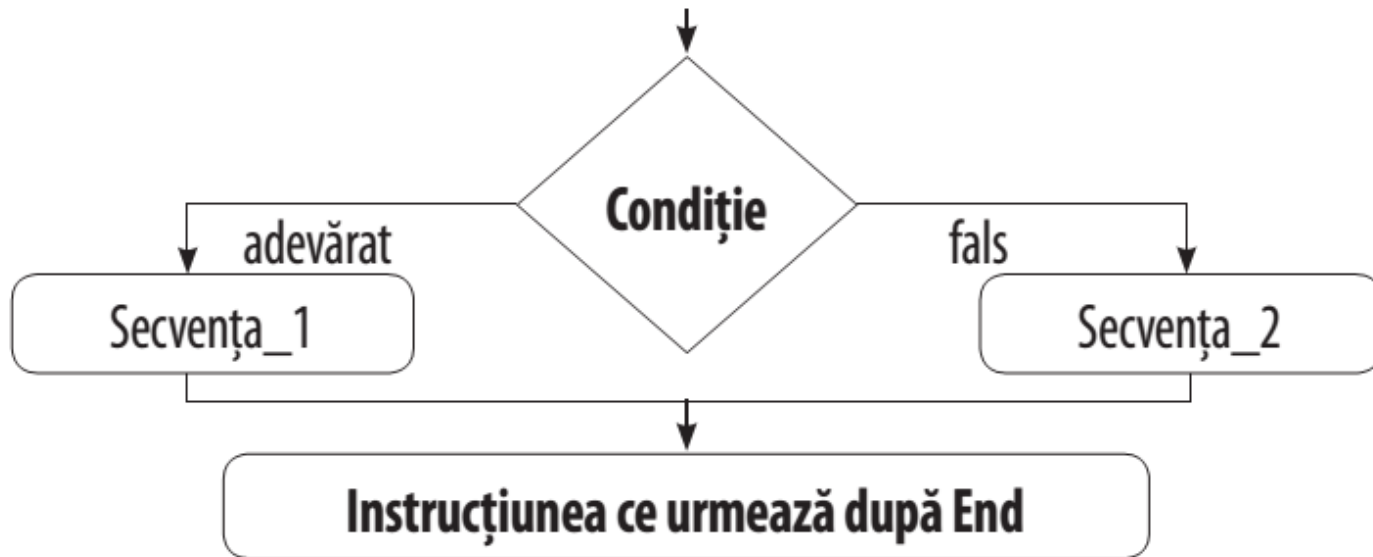
End

Condiția – reprezintă o expresie logică;

Secvența_1 – reprezintă un set de instrucțiuni care se va executa în cazul când valoarea de adevăr a condiției este *True*;

Secvența_2 – reprezintă un set de instrucțiuni care se va executa în cazul când valoarea de adevăr a condiției este *False*.

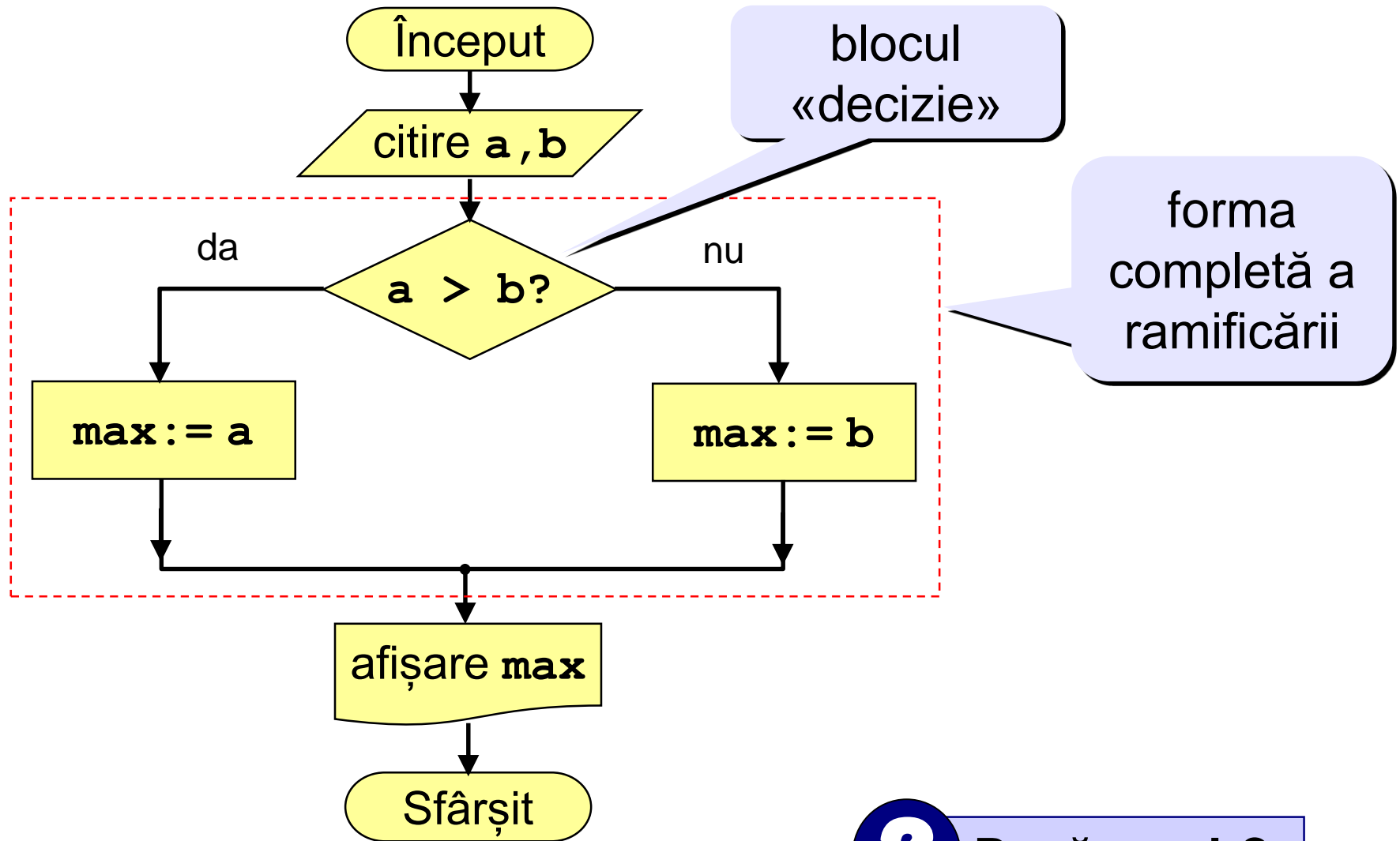
Execuția construcției If-Then-Else



Execuția construcției If-Then-Else:

- *evaluarea condiției;*
 - *dacă valoarea condiției este True se execută Secvența_1 (ramura Then), apoi gestiunea se transmite la executarea instrucțiunii ce urmează după End;*
 - *dacă valoarea condiției este False se execută Secvența_2 (ramura Else), apoi gestiunea se transmite la executarea instrucțiunii ce urmează după End.*

Exemplu 1. Schema-bloc



? Dacă $a = b$?

Exemplu 1. Algoritm

Algoritm Ex1

Var a, b, max: Integer

Begin

WriteString('Introdu doua numere intregi')

ReadInt(a, b)

If a >= b then

max := a

else

max := b

End

WriteString('Numar maxim = ')

WriteInt(max)

End

forma completă
a ramificării

Construcții de ramificare imbricate

Problemă: În variabilele **a** și **b** sunt păstrate valorile vârstelor lui Andrei și Boris. Cine din ei este mai mare?



Câte variante?

```
If a > b then
  WriteString('Andrei este mai mare')
else
  If a = b then
    WriteString('Au aceeași vîrstă')
  else
    WriteString('Boris este mai mare')
  End
End
```

Construcție de
ramificare imbricată

Condiții mixte

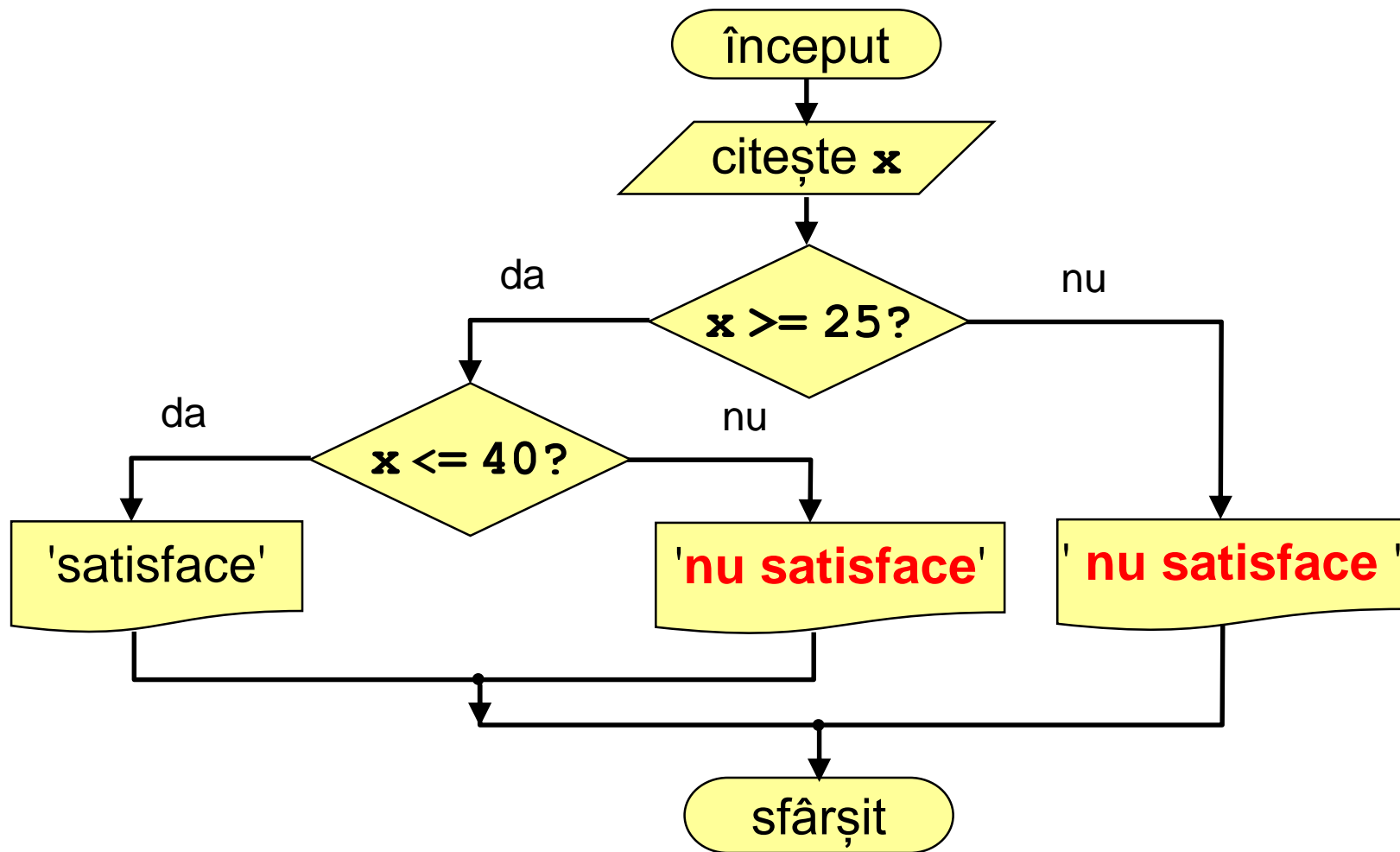
Problemă. O organizație recrutează lucrători cu vârsta cuprinsă între 25 și 40 ani inclusiv. Să se introducă vârsta persoanei și să se determine, dacă persoana satisface condiției (să se afișeze răspuns «satisfacă» sau «nu satisfacă»).

Particularități: e necesar de verificat dacă ambele condiții se îndeplinesc concomitent.



Se poate rezolva prin metodele cunoscute?

Exemplu 2. Schema-bloc



Exemplu 2A. Algoritm

Algoritm Ex2

Var x: Integer

Begin

WriteString('Introduceti virsta')

ReadInt(x)

If x >= 25 then

 If x <= 40 then

 WriteString('Satisface')

 else

 WriteString('Nu satisface')

 End

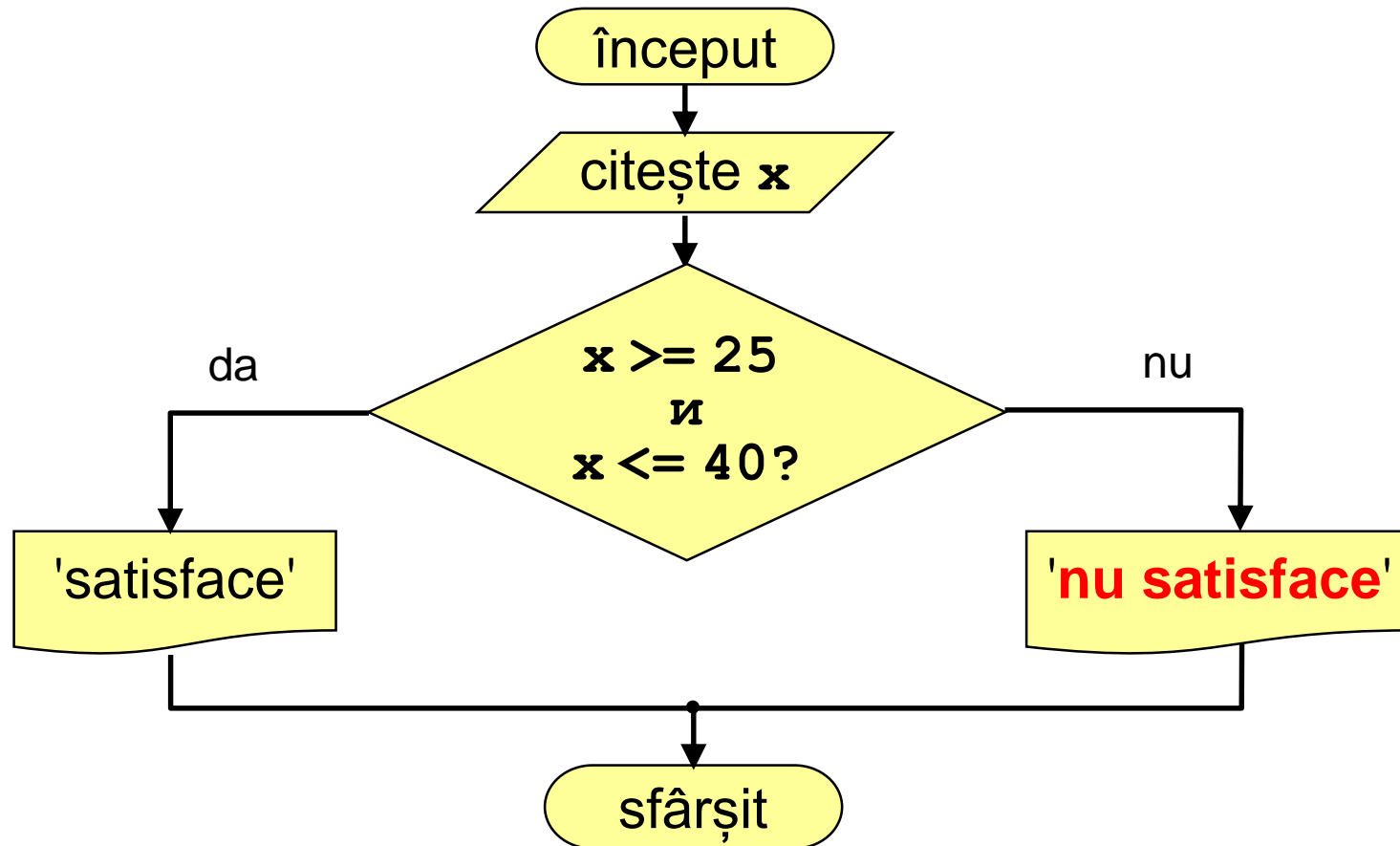
else

 WriteString('Nu satisface')

End

End

Exemplu 2. Schema-bloc



Exemplu 2B. Algoritm

Algoritm Ex2

Var x: Integer

Begin

WriteString('Introduceti virsta')

ReadInt(x)

If (x >= 25) and (x <= 40) then

WriteString('Satisface')

else

WriteString('Nu satisface')

End

End

condiție
mixtă

Condiții mixte

Condiții simple (relații)

< <= > >= = <>

egal

diferit

Condiție mixtă – este o condiție, alcătuită din câteva condiții simple (relații), legate între ele cu ajutorul **operațiilor logice**:

- **not** – NU (negație)
- **and** – ȘI (înmulțire logică, conjuncție, îndeplinirea concomitentă a condițiilor)
- **or** – SAU (adunare logică, disjuncția, îndeplinirea cel puțin unei condiții)
- **xor** – SAU exclusiv (îndeplinirea numai unei condiții și nu a ambelor)

Condiții mixte

Ordinea execuției (prioritatea)

- expresii din paranteze
- **not**
- **and**
- **or, xor**
- **<, <=, >, >=, =, <>**

Particularitate – fiecare condiție simplă trebuie de inclus în paranteze.

Exemplu

```
      4      1      6      2      5      3  
If not (a > b) or (c <> d) and (b <> a) then  
    ...  
end
```

Tipuri de condiții

- o **constantă booleană** (True sau False);
- o **variabilă booleană**, care poate avea una din valorile True sau False;
- o **expresie booleană**, de exemplu (x or y);

O **expresie booleană** este o expresie care se obține prin aplicarea de un număr finit de ori a operatorilor And, Or, Xor, Not asupra unei variabile de tip Boolean.

- o **expresie relațională**, de exemplu $x < y$, unde x și y sunt variabile de același tip;
- o **expresie mixtă**, de exemplu $(x < y) \text{ and } (a > 7)$;

Expresia mixtă conține concomitent și operații logice (And, Or, Xor, Not) și operații relaționale ($=$, $<$, $>$, $<=$, $>=$, $<>$).

- **apelul unei funcții logice.**

Exemple de condiții

- A este **număr pozitiv**: $A > 0$;
- A este **număr par**: $A \bmod 2 = 0$;
- A este **număr impar**: $A \bmod 2 \neq 0$;
- A este **divizibil prin 7**: $A \bmod 7 = 0$;
- A, B și C pot fi **lungimile laturilor unui triunghi**:
 $(A > 0) \text{ and } (B > 0) \text{ and } (C > 0) \text{ and } (B + C > A) \text{ and } (A + C > B) \text{ and } (A + B > C)$;
- A reprezintă un **număr natural format din 3 cifre**: $(A \geq 100) \text{ and } (A \leq 999)$;
- A reprezintă **numărul unei luni de vară**:
 $(A \geq 6) \text{ and } (A \leq 8) \text{ sau } (A = 6) \text{ or } (A = 7) \text{ or } (A = 8)$;
- A, B și C reprezintă **lungimile laturilor unui triunghi isoscel**: $(A = B) \text{ or } (B = C) \text{ or } (C = A)$.

Simplificarea expresiilor logice

Exemplu: A reprezintă numărul unei luni de primăvara:

1. $(A \geq 3)$ and $(A \leq 5)$ sau
2. $(A=3)$ or $(A=4)$ or $(A=5)$.

Calcularea valorii de adevăr a expresiei logice va fi mai rapidă în cazul când expresia este mai simplă.

Axioma distributivă 1:

$$(P \text{ or } Q) \text{ and } (P \text{ or } R) = P \text{ or } (Q \text{ and } R)$$

Două expresii logice de aceleași variabile, sunt **echivalente** dacă pentru aceleași șiruri de valori ale variabilelor, expresiile iau aceleași valori.

Tabelul de adevăr pentru expresia (P or Q) and (P or R)

P	Q	R	P or Q	P or R	(P or Q) and (P or R)
False	False	False	False	False	False
False	False	True	False	True	False
False	True	False	True	False	False
False	True	True	True	True	True
True	False	False	True	True	True
True	False	True	True	True	True
True	True	False	True	True	True
True	True	True	True	True	True

Tabelul de adevăr pentru expresia $P \text{ or } (Q \text{ and } R)$

P	Q	R	Q and R	P or (Q and r)
False	False	False	False	False
False	False	True	False	False
False	True	False	False	False
False	True	True	True	True
True	False	False	False	True
True	False	True	False	True
True	True	False	False	True
True	True	True	True	True

Simplificarea expresiilor logice

Axioma distributivă 2:

$$(P \text{ and } Q) \text{ or } (P \text{ and } R) = P \text{ and } (Q \text{ or } R)$$

Legile de Morgan

- $(\text{Not } P) \text{ and } (\text{Not } Q) = \text{Not } (P \text{ or } Q)$
- $(\text{Not } P) \text{ or } (\text{Not } Q) = \text{Not } (P \text{ and } Q)$

Ce este greșit?

```
if a > b then  
    a := b  
else begin  
    b := a  
end
```

```
if a > b then  
    a := b  
else  
    b := a  
end
```

```
if a > b then  
    a := b  
else b > a  
    b := a  
end
```

```
if a > b then  
    a := b  
else  
    b := a  
end
```

Expresii mixte

Adevărat sau fals pentru $a := 2; b := 3; c := 4;$

$\text{not } (a > b)$ True

$(a < b) \text{ and } (b < c)$ True

$\text{not } (a \geq b) \text{ or } (c = d)$ True

$(a < c) \text{ or } (b < c) \text{ and } (b < a)$ True

$(a < b) \text{ xor not } (b > c)$ FALSE

Pentru ce valori ale lui X expresiile logice sunt adevărate:

$(x < 6) \text{ and } (x < 10)$

$(x < 6) \text{ and } (x > 10)$

$(x > 6) \text{ and } (x < 10)$

$(x > 6) \text{ and } (x > 10)$

$(x < 6) \text{ or } (x < 10)$

$(x < 6) \text{ or } (x > 10)$

$(x > 6) \text{ or } (x < 10)$

$(x > 6) \text{ or } (x > 10)$

$(-\infty; 6)$	$x < 6$
\emptyset	
$(6; 10)$	
$(10; \infty)$	$x > 10$
$(-\infty; 10)$	$x < 10$
$(-\infty; 6) \cup (10; \infty)$	
$(-\infty; \infty)$	
$(6; \infty)$	$x > 6$

Exemplu 3:

*Fie că variabila N de tip Integer păstrează valoarea unui număr întreg. În funcție de valoarea acestui număr să se afișeze mesajul „**Numărul este par**” sau „**Numărul este impar**”.*

Rezolvare:

If $N \bmod 2 = 0$ **Then**

 WriteStrihg ('Număr par')

Else

 WriteString('Număr impar')

End

Construcția If-Then

Formatul construcției If-Then:

If condiție **Then**

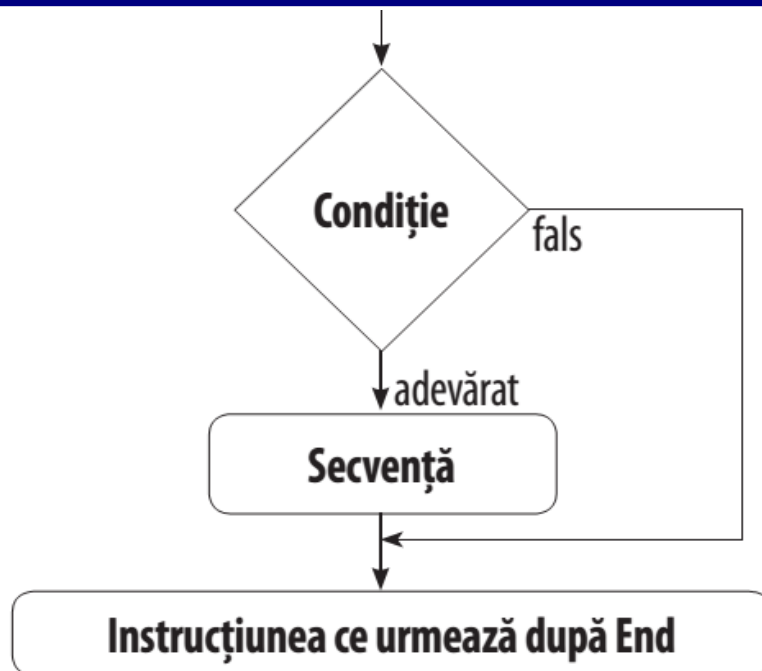
Secvența_1

End

Condiția – reprezintă o expresie logică;

Secvența_1 – reprezintă un set de instrucțiuni care se va executa în cazul când valoarea de adevăr a condiției este *True*.

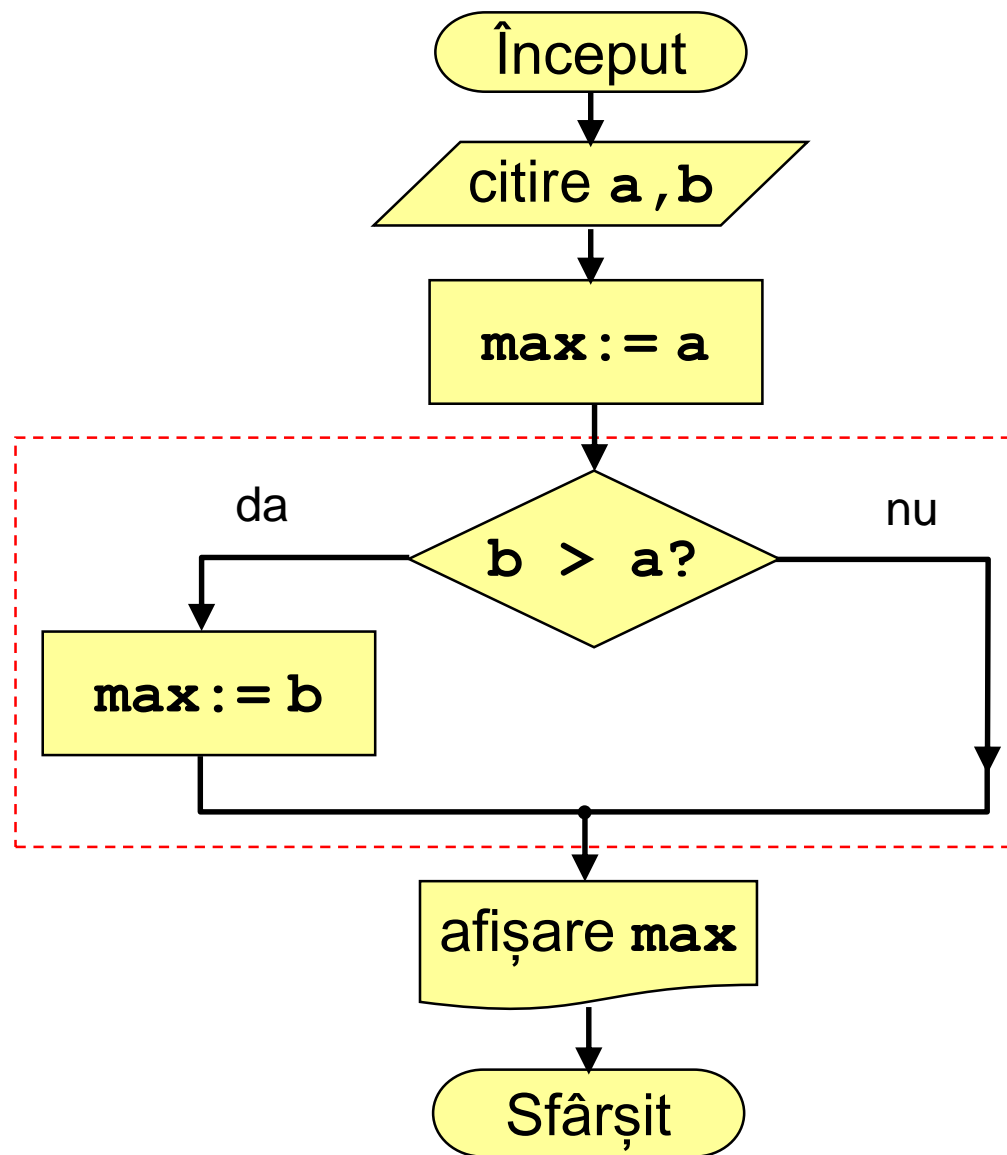
Execuția construcției If-Then



Execuția construcției If-Then:

- *evaluarea condiției:*
 - *dacă valoarea condiției este True se execută **Secvența** (ramura Then), apoi gestiunea se transmite la executarea instrucțiunii ce urmează după End;*
 - *dacă valoarea expresiei este False se ocolește executarea secvenței și gestiunea se transmite la executarea **instrucțiunii ce urmează după End**.*

Exemplu 1. Schema-bloc



forma
incompletă a
ramificării

Exemplu 1A. Algoritm

Algoritm Ex1A

Var a, b, max: Integer

Begin

WriteString('Introdu doua numere intregi')

ReadInt(a, b)

max := a

If b > a then

max := b

end

WriteString('Numar maxim =')

WriteInt(max)

End

forma
incompletă a
ramificării

Exemplu 1B. Algoritm

Algoritm Ex1B

Var a, b, max: Integer

Begin

WriteString('Introdu doua numere intregi')

ReadInt(a, b)

max := b

If a > b then

max := a

End

WriteString('Numar maxim =')

WriteInt(max)

End

Exemplu 2:

*Să se elaboreze algoritmul care citește valorile a trei temperaturi; determină **temperatura minimă** și afișează mesajul '**E frig**' dacă temperatura minimă este mai mică de - 4 grade.*

Datele problemei: valorile a trei temperaturi.

Rezultatul: temperatura minimă.

Detalierea problemei

1. Introducerea valorilor celor trei temperaturi:

- 1.1. Introducerea valorii primei temperaturi;
- 1.2. Introducerea valorii celei de a doua temperaturi;
- 1.3. Introducerea valorii celei de a treia temperaturi.

2. Determinarea valorii temperaturii minime:

- 2.1. Determinarea temperaturii minime dintre primele două temperaturi;
- 2.2. Determinarea temperaturii minime dintre cea de a treia temperatură și temperatura minimă determinată în subproblema 2.1.

3. Afișarea rezultatelor:

- 3.1. Afișarea valorii temperaturii minime;
- 3.2. Afișarea mesajului 'E frig', dacă este necesar.

Detalierea subproblemei 1.

Introducerea valorilor celor trei temperaturi

Declararea variabilelor:

Temp1 : Integer

Temp2 : Integer

Temp3 : Integer

Algoritmul de rezolvare a subproblemei 1.

WriteString('Introduceți 3 temperaturi')

ReadInt(Temp1)

Writeln

ReadInt(Temp2)

Writeln

ReadInt(Temp3)

Writeln

Detalierea subproblemei 2.

Deteminarea valorii temperaturii minime

Declararea variabilei:

Min_temp : Integer

Algoritmul de rezolvare a subproblemei 2.1.

If Temp1 < Temp2 Then

Min_temp := Temp1

Else

Min_temp := Temp2

End

Algoritmul de rezolvare a subproblemei 2.2.

If Temp3 < Min_Temp Then

Min_temp := Temp3

End

Detalierea subproblemei 3.

Afișarea rezultatelor

Detalierea subproblemei 3.1:

```
WriteInt(Min_temp)  
WriteLn
```

Detalierea subproblemei 3.2:

```
If Min_temp < -4 Then  
    WriteString('E frig')  
End
```

Descrierea algoritmului

Algorithm Temp_min

Var

Temp1: Integer
Temp2: Integer
Temp3: Integer
Min_temp: Integer

Lista variabilelor
utilizate

Subproblema 1

Begin

WriteString('Introduceti 3 temperaturi')
ReadInt(Temp1)
Writeln
ReadInt(Temp2)
Writeln
ReadInt(Temp3)
Writeln

Descrierea algoritmului

```
If Temp1 <= Temp2 Then
    Min_temp := Temp1
Else
    Min_temp := Temp2
End
If Temp3 < Min_Temp Then
    Min_temp := Temp3
End
```

Subproblema 2

```
WriteInt(Min_temp)
Writeln
If Min_temp < -4 Then
    WriteString('E frig')
End
```

Subproblema 3

```
End
```

Exemplu 3:

Să se calculeze valoarea funcției Y definite în felul următor:

$$Y = \begin{cases} X+2, & \text{dacă } X > 0 \\ 4, & \text{dacă } X = 0 \\ X * 3, & \text{dacă } X < 0 \end{cases}$$

Datele problemei: valoarea întreagă a funcției Y.

Rezultatul: valoarea funcției Y.

Subproblemele:

1. Introducerea valorii X;
2. Calcularea valorii funcției Y;
3. Afișarea rezultatului.

Detalierea subproblemei 1.

Introducerea valorii X

Declararea variabilei:

X: Integer

Algoritmul de rezolvare a subproblemei 1.

WriteString('Introduceți număr întreg')

ReadInt(X)

WriteLn

Detalierea subproblemei 2.

Calcularea valorii funcției Y

Declararea variabilei:

Y : Integer

Algoritmul de rezolvare a subproblemei 2.

If* $X > 0$ ***Then**

Y := $X + 2$

Else

If* $X = 0$ ***Then**

Y := 4

Else

Y := $X * 3$

End

End

Detalierea subproblemei 3.

Afișarea rezultatului

Algoritmul de rezolvare a subproblemei 3.

```
WriteString('Y= ')  
WriteInt(Y)
```

Descrierea algoritmului

Algorithm Functia_Y

Var

X, Y: Integer

Lista variabilelor
utilizate

Subproblema 1

Begin

WriteString('Introduceti numar intreg')

ReadInt(X)

WriteLn

If X>0 Then

Y:=X+2

Else

If X=0 Then

Y:=4

Else

Y:=X*3

End

End

Subproblema 2

Subproblema 3

WriteInt(Y)

End

Realizarea selecției multiple

Fie că n condiții c_1, c_2, \dots, c_n provoacă execuția unei secvențe s_1, s_2, \dots, s_n , la un moment dat existând posibilitatea îndeplinirii unei singure condiții din cele n .

Realizarea neeficientă a problemei

```
If  $c_1$  Then  
     $s_1$   
End  
If  $c_2$  Then  
     $s_2$   
End  
...  
If  $c_n$  Then  
     $s_n$   
Else  
    Eroare  
End
```

Se vor executa toate instrucțiunile If, chiar dacă s-a ajuns în situația că una dintre condiții este îndeplinită.

Utilizarea eficientă a construcțiilor If incluse

```
If  $c_1$  Then  
     $s_1$   
Else  
    If  $c_2$  Then  
         $s_2$   
    Else  
        If  $c_{n-1}$  Then  
             $s_{n-1}$   
        Else  
            If  $c_n$  Then  
                 $s_n$   
            Else  
                WriteString('Eroare')  
            End  
        End  
    End  
End
```

Toate condițiile vor fi evaluate numai în cazul când condiția c_n este adevărată sau când nici una din condițiile c_1, c_2, \dots, c_n nu este adevărată.

Exemplu

Afișarea denumirii zilei a săptămânii în funcție de numărul ei

```
If Nr_zi=1 Then
    WriteString(,Luni')
Else
    If Nr_zi=2 Then
        WriteString(,Marți')
    Else
        If Nr_zi=3 Then
            WriteString(,Miercuri')
        Else
            If Nr_zi=4 Then
                WriteString(,Joi')
            Else
                If Nr_zi=5 Then
                    WriteString(,Vineri')
                Else
                    If Nr_zi=6 Then
                        WriteString(,Sâmbătă')
                    Else
                        If Nr_zi=7 Then
                            WriteString(,Duminică')
                        Else
                            WriteString(,Eroare')
                        End
                    End
                End
            End
        End
    End
End
End
End
End
End
```

Atribuirea valorii unei variabile logice x

```
If <condiție> Then  
    X:=True  
Else  
    X:=False  
End
```

$X := \text{<condiție>}$

Exemplu: Considerăm că variabila N păstrează valoarea unui număr natural. Să i se atribuie variabilei X valoarea *True*, dacă numărul este par și valoarea *False*, dacă numărul este impar.

Rezolvare:

```
If  $N \bmod 2 = 0$  Then  
     $X := \text{True}$   
Else  
     $X := \text{False}$   
End
```

$X := N \bmod 2 = 0$

Instrucțiunea Case (selecția multiplă)

Formatul construcției Case:

Case selector **Of**

Alternativa_1: secvența_1

Alternativa_2: secvența_2

...

Alternativa_n: secvența_n

Else

Secvența_x

End

Selector – reprezintă o expresie ordinală, care exprimă un criteriu, valoarea căruia va condiționa executarea deciziei respective.

Alternativa_i – reprezintă o constantă. Tipul de date al selectorului și al constantei trebuie să coincidă.

Secvența_i – reprezintă setul de instrucțiuni care se execută în cazul când valoarea selectorului este egală cu *Alternativa_i*.

Execuția construcției Case

Execuția construcției Case:

- *evaluarea selectorului:*
 - *se caută alternativa care cuprinde valoarea selectorului:*
 - *dacă se găsește o asemenea alternativă, se execută secvența care o urmează, apoi gestiunea se transmite la instrucțiunea care urmează după End;*
 - *în caz contrar, se execută secvența care urmează după Else, în lipsa acestuia nu se execută nimic. Gestiunea se transmite la instrucțiunea ce urmează după End.*

Exemplu 4:

*De la tastatură se introduce numărul lunii anului.
Să se afișeze anotimpul corespunzător.*

Datele problemei: numărul lunii anului.

Rezultatul: afișarea anotimpului corespunzător.

Subproblemele:

1. Introducerea numărului lunii anului;
2. Afișarea denumirii anotimpului.

Detalierea subproblemei 1.

Introducerea numărului lunii anului

Declararea variabilei:

N_luna: Natural

Algoritmul de rezolvare a subproblemei 1.

WriteString('Introduceți numărul lunii anului')

ReadNat(N_luna)

WriteLn

Detalierea subproblemei 2.

Afișarea denumirii anotimpului

Algoritmul de rezolvare a subproblemei 2.

Case *N_luna* **of**

1,2,12: *WriteString('Iarna')*

3,4,5: *WriteString('Primavara')*

6..8: *WriteString('Vara')*

9..11: *WriteString('Toamna')*

Else

WriteString('Eroare')

End

Descrierea algoritmului

Algorithm Anotimp

Var

N_luna: Natural

Lista variabilelor
utilizate

Subproblema 1

Begin

```
WriteString('Introdu numarul lunii anului')  
ReadNat(N_luna)  
Writeln
```

Case N_luna **of**

```
  1,2,12:WriteString('Iarna')  
  3,4,5: WriteString('Primavara')  
  6..8:  WriteString('Vara')  
  9..11: WriteString('Toamna')
```

Subproblema 2

else

```
  WriteString('Eroare')
```

end

End

Selecția multiplă

Particularități:

- Alternativele care provoacă executarea aceleiași secvențe pot fi unite

enumerare

interval

mixt

```
Case i of
  1:           a := b
  2,4,6:       a := c
  10..15:      a := d
  20,21,25..30: a := e
  else WriteString('Eroare')
end
```

Exemplu 5:

*De la tastatură se introduc 2 numere reale și operația (+, -, *, /). Să se calculeze rezultatul operației primului număr cu cel de-al doilea.*

Datele problemei: 2 numere reale și operația.

Rezultatul: rezultatul operației.

Subproblemele:

1. Introducerea celor două numere și semnului unei operații;
2. Calcularea rezultatului aplicării operației asupra celor două numere;
3. Afișarea rezultatului.

Detalierea subproblemei 1.

Introducerea celor două numere și semnului unei operații

Declararea variabilelor:

N1: Real
N2: Real
Op: Char

Algoritmul de rezolvare a subproblemei 1.

WriteString('Introduceți 2 numere reale')
ReadReal(N1)
ReadReal(N2)
WriteLn
WriteString('Introduceți semnul unei operații')
ReadChar(Op)
WriteLn

Detalierea subproblemei 2.

Calcularea rezultatului

Declararea variabilei:

Rez: Real

Algoritmul de rezolvare a subproblemei 2.

Case Op of

'+' : Rez := N1 + N2

'-' : Rez := N1 - N2

'' : Rez := N1 * N2*

*'/' : **If** N2 <> 0 **Then**
 Rez := N1 / N2*

Else

WriteString('Împărțire la zero')

End

Else

WriteString('Operație incorectă')

End

Detalierea subproblemei 3.

Afişarea rezultatului

Algoritmul de rezolvare a subproblemei 3.

```
If ( $Op = '+'$ ) or ( $Op = '-'$ ) or ( $Op = '*'$ ) or ( $Op = '/'$ )  
    and ( $N2 \neq 0$ ) Then  
    WriteReal(Rez)  
End
```

Descrierea algoritmului

Algorithm Calculator

Var

N1: Real

N2: Real

Op: Char

Rez: Real

Lista variabilelor utilizate

Subproblema 1

Begin

```
WriteString('Introduceti 2 numere reale')
```

ReadReal (N1)

ReadReal (N2)

Writeln

```
WriteString('Introduceti semnul unei  
operatii')
```

ReadChar (Op)

Writeln

Descrierea algoritmului

Case Op of

'+' : Rez := N1 + N2

'-' : Rez := N1 - N2

'*' : Rez := N1 * N2

'/' : If N2<>0 then

Rez:= N1 / N2

else WriteString('Impartire la zero')

end

end

If (OP '+') Or (Op=' -') Or (Op ='*')

Or ((Op=' /') And (N2 #0)) then

WriteReal(Rez)

end

End

Subproblema 2

Subproblema 3

Ce este greșit?

```
case a of
  2: begin a := b
  4: a := c
end
```

```
case a of
  2: a := b
  4: a := c
end
```

```
case a of
  2..5: a := b
  4: a := c
end
```

```
case a of
  0..2: a := b
  3..6: a := c
end
```

```
case a+c/2 of
  2: a := b
  4: a := c
end
```

```
case a of
  'a', 'b': a := b; d := 0
  'c': b := a
end
```

Exemplu 6:

Să se calculeze suma a două intervale de timp, exprimate în ore, minute, secunde. Se consideră că suma orelor nu depășește 24 ore și valorile intervalelor sunt introduse corect.

Datele problemei: 2 intervale de timp.

Rezultatul: suma acestor două intervale.

Subproblemele:

1. Introducerea intervalelor de timp;
2. Calcularea sumei intervalelor de timp;
3. Afișarea sumei intervalelor de timp.

Detalierea subproblemei 1.

Introducerea intervalelor de timp

Declararea variabilelor:

Ore1, Min1, Sec1: Natural
Ore2, Min2, Sec2: Natural

Subproblema	Detalierea
1.1.1. Introducerea orelor primului interval de timp	<i>ReadNat(Ore1)</i>
1.1.2. Introducerea minutelor primului interval de timp	<i>ReadNat(Min1)</i>
1.1.3. Introducerea orelor primului interval de timp	<i>ReadNat(Sec1)</i>
1.2.1. Introducerea orelor celui de-al doilea interval de timp	<i>ReadNat(Ore2)</i>
1.2.2. Introducerea minutelor celui de-al doilea interval de timp	<i>ReadNat(Min2)</i>
1.2.3. Introducerea secundelor celui de-al doilea interval de timp	<i>ReadNat(Sec2)</i>

Detalierea subproblemei 2.

Calcularea rezultatului

Declararea variabilelor:

Ore3, Min3, Sec3: Natural

Subproblema	Detalierea
2.1. Calcularea sumei secundelor	<i>Sec3:= Sec1 + Sec2</i>
2.2. Calcularea sumei minutelor	<i>Min3:= Min 1 + Min2</i>
2.3. Calcularea sumei orelor	<i>Ore3:= Ore 1 + Ore2</i>
2.4. Corecția valorii secundelor, dacă este necesar (suma secundelor ≥ 60);	<i>If Sec3\geq60 Then Min3:= Min3 + 1 Sec3:= Sec3 – 60 End</i>
2.5. Corecția valorii minutelor, dacă este necesar (suma minutelor ≥ 60).	<i>If Min3\geq60 Then Ore3:= Ore3 + 1 Min3:= Min3 – 60 End</i>

Detalierea subproblemei 3.

Afişarea rezultatului

Subproblema	Detalierea
3.1. Afişarea sumei orelor	<i>WriteNat(Ore3)</i>
3.2. Afişarea sumei minutelor	<i>WriteNat(Min3)</i>
3.3. Afişarea sumei secundelor	<i>WriteNat(Sec3)</i>

Descrierea algoritmului

Algoritm Timp

Var

Ore1, Min1, Sec1: Natural
Ore2, Min2, Sec2: Natural
Ore3, Min3, Sec3: Natural

Lista variabilelor utilizate

Begin

WriteString('Introduceți 2 intervale de timp')
ReadNat(Ore1)
ReadNat(Min1)
ReadNat(Sec1)
ReadNat(Ore2)
ReadNat(Min2)
ReadNat(Sec2)
Writeln

Subproblema 1

Descrierea algoritmului

```
Sec3:= Sec1 + Sec2
Min3:= Min 1 + Min2
Ore3:= Ore 1 + Ore2
If Sec3>=60 Then
    Min3:= Min3 + 1
    Sec3:= Sec3 - 60
End
If Min3>=60 Then
    Ore3:= Ore3 + 1
    Min3:= Min3 - 60
End
```

Subproblema 2

```
WriteNat(Ore3)
WriteNat(Min3)
WriteNat(Sec3)
```

Subproblema 3

End

Sarcini pentru lucrul independent

1. Să se determine de ce tip (isoscel, echilateral, dreptunghic) este un triunghi cu laturile a , b și c .
2. Considerăm două numere întregi ce reprezintă un an, respectiv o lună a acestui an. Să se elaboreze algoritmul care stabilește numărul de zile din luna respectivă.
3. Cunoscând data curentă, exprimată prin trei numere întregi, reprezentând anul, luna, ziua precum și data nașterii a unei persoane, exprimată la fel, să se elaboreze un algoritm care să calculeze vârsta persoanei respective în număr de ani, luni și zile.