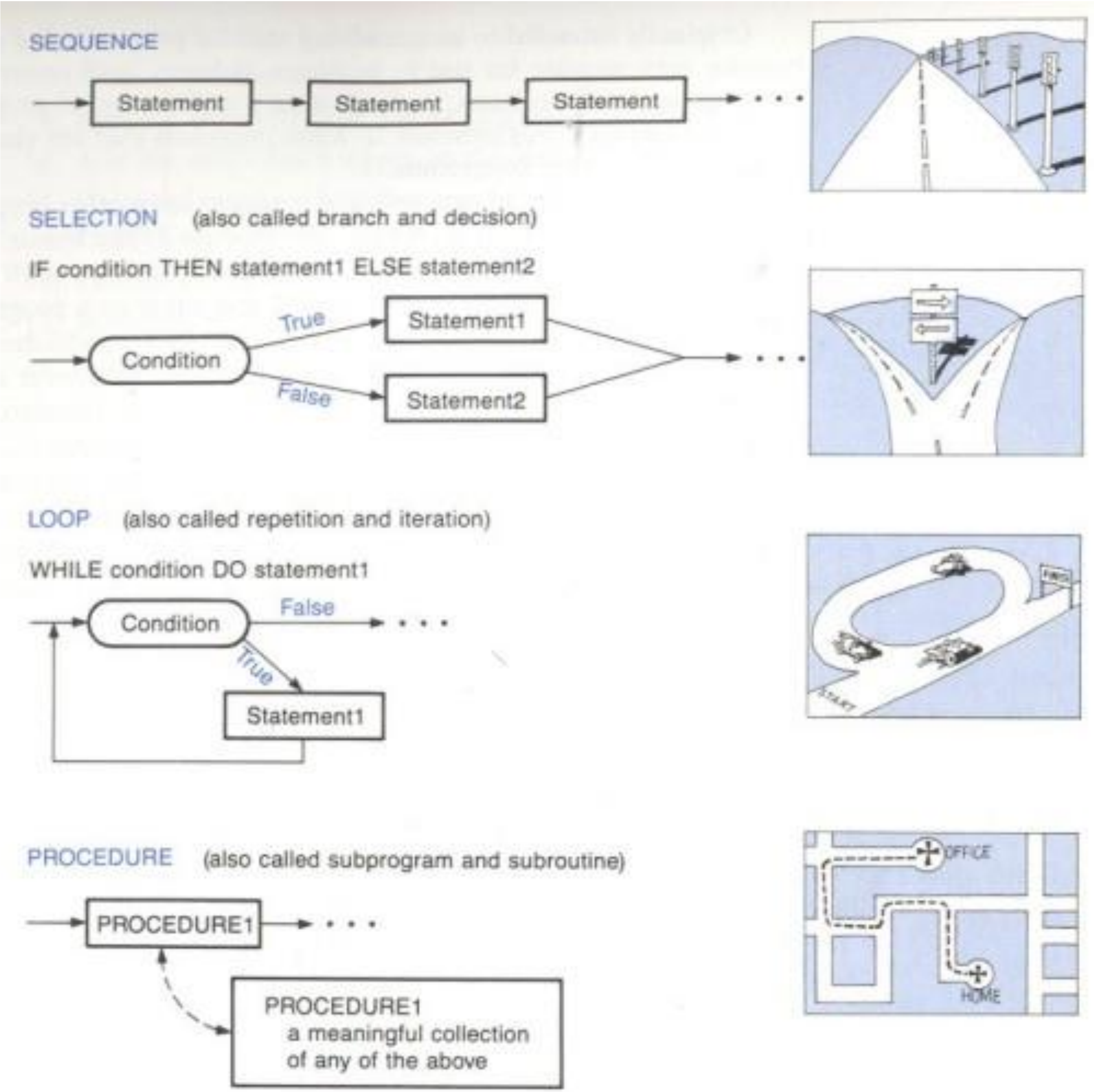
Loops

PC L04

Loops



SEQUENCE

SELECTION (also called branch and decision)

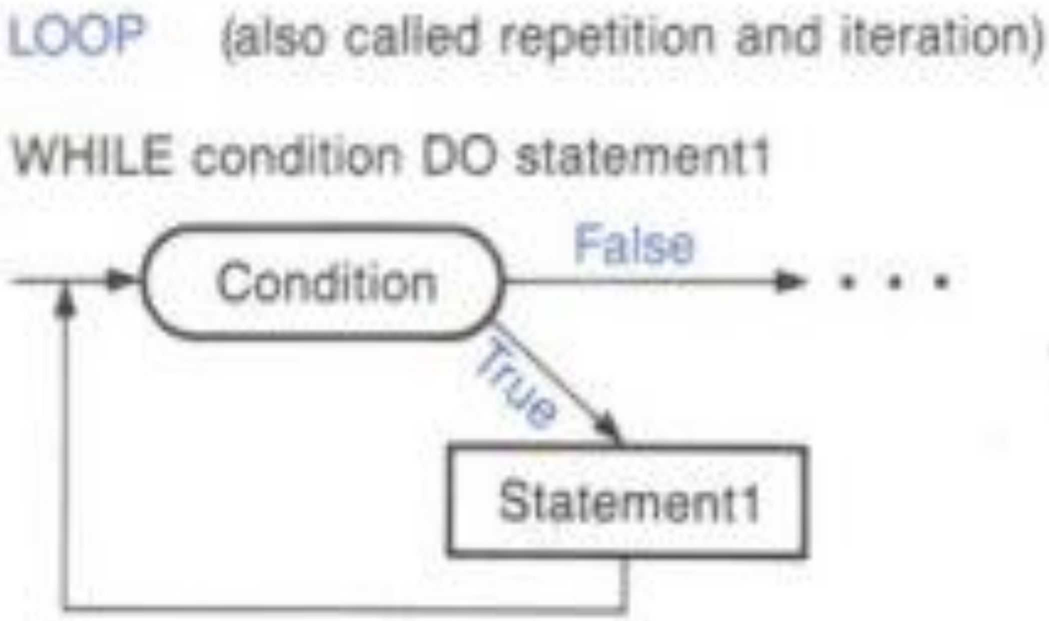
IF condition THEN statement1 ELSE statement2

LOOP (also called repetition and iteration)

WHILE condition DO statement1

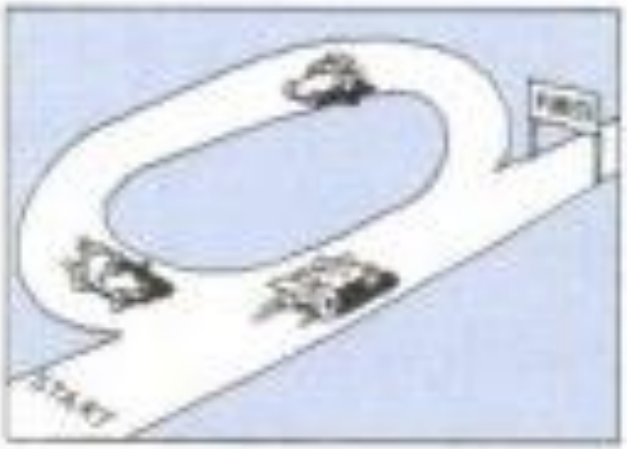
PROCEDURE (also called subprogram and subroutine)

PROCEDURE1
a meaningful collection
of any of the above



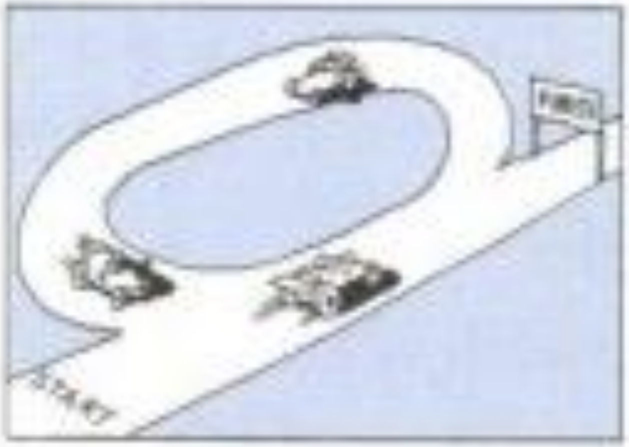
LOOP (also called repetition and iteration)

WHILE condition DO statement1



Loops

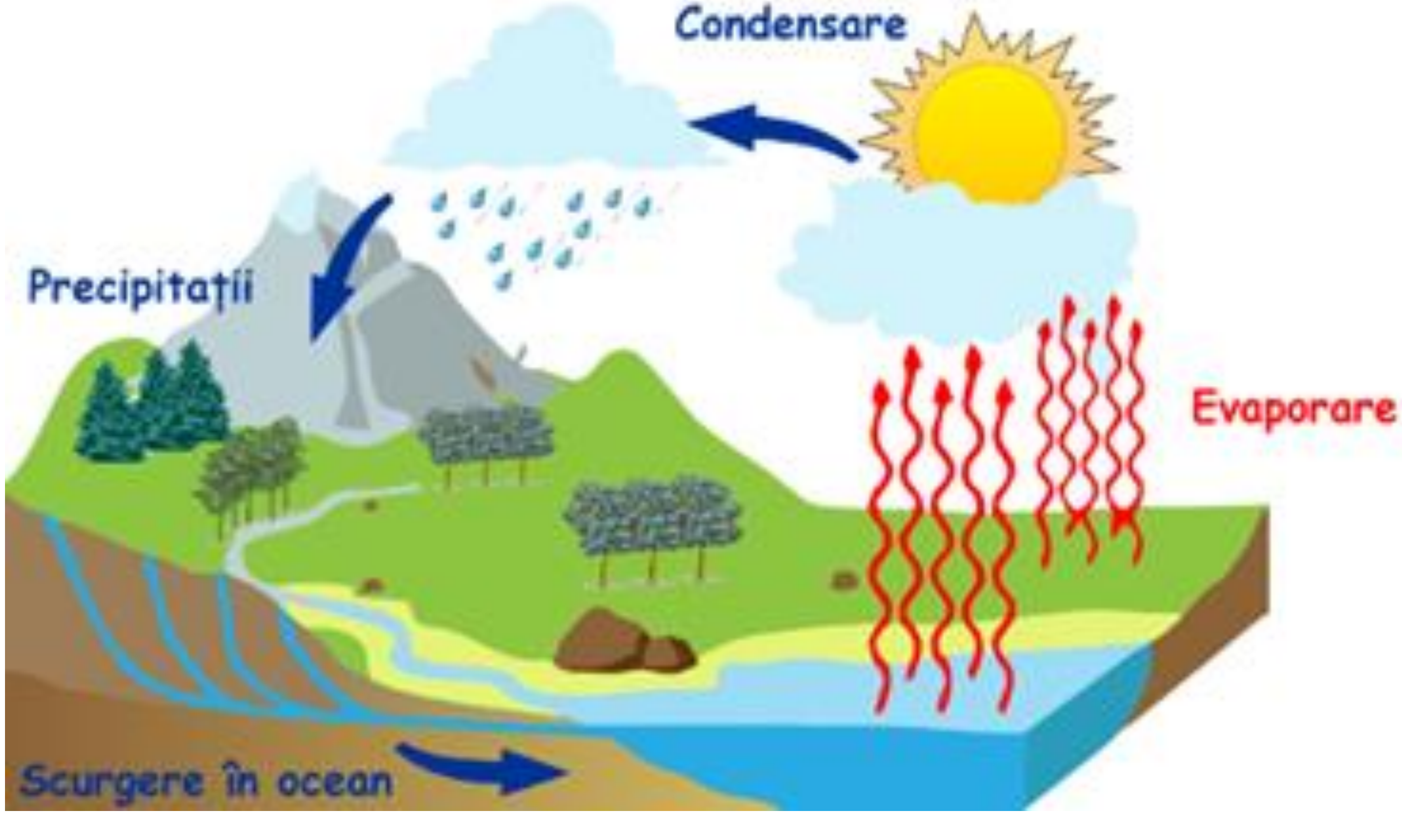
Loops



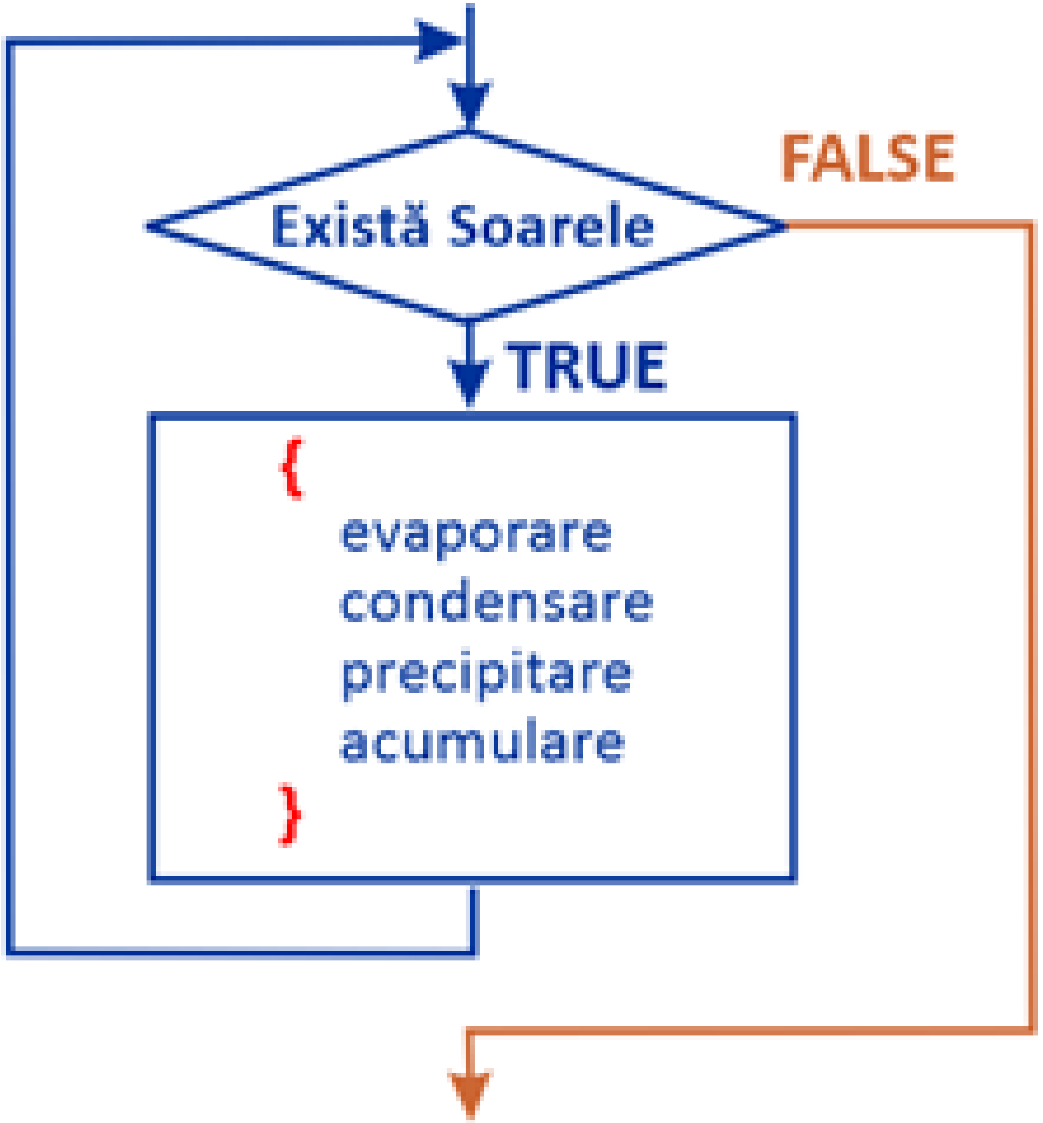
Looping control structures give us a way to make the logical order different from the physical order: A loop executes the same statement (simple or compound) over and over, as long as a condition (or set of conditions) is met.

Loop types

Conditional controlled loops:



The scheme:


FALSE
Există Soarele
TRUE
{
evaporare
condensare
precipitare
acumulare
}

Condition-controlled loop - the loop that repeats the execution of a sequence of instructions as long as a condition remains true. The condition is described by a Boolean or arithmetic test expression.

while loop structure

```
while ( expression )
    {
     statements (commands);
    }
```

Example 1: the sum of digits for number n

Problem analysis:

1. Input number
2. Set the sum to 0
3. Separate last digit…
   a. Divide by 10 … modulo!
   b. Divide by 10… use division!
   c. Add separated digit to sum!
4. If n remains greater than 0 go to step 3 else goto step 5

5. Print the sum!

The code

```
#include <iostream>
   using namespace std;
int main()
{
   int n, sum = 0;
   cin >> n;
   while (n)
   {
      int k = n % 10;
      sum = sum + k;
      n = n / 10;
   }
   cout << sum;
   return 0;
}
```

Controll

```
#include <iostream>
        using namespace std;
int main()
{
        int n, sum = 0;
        cin >> n;
        while (n)
        {
                int k = n % 10;
                sum = sum + k;
                n = n / 10;
                 cout << sum << k << n << endl;
        }
        cout << sum;
        return 0;
}
```

Problem analysis:

Example 2: inverse the number n

1. Input number
2. Set the inverse to 0 .
3. Separate last digit from n…
   a. Multiply the inverse with 10
   b. Divide n by 10 … modulo!
   c. Divide n by 10… use division!
   d. Add separated digit to inverse!
4. If n remains greater than 0 go to step 3 else goto step 5

5. Print the inverse!

The code

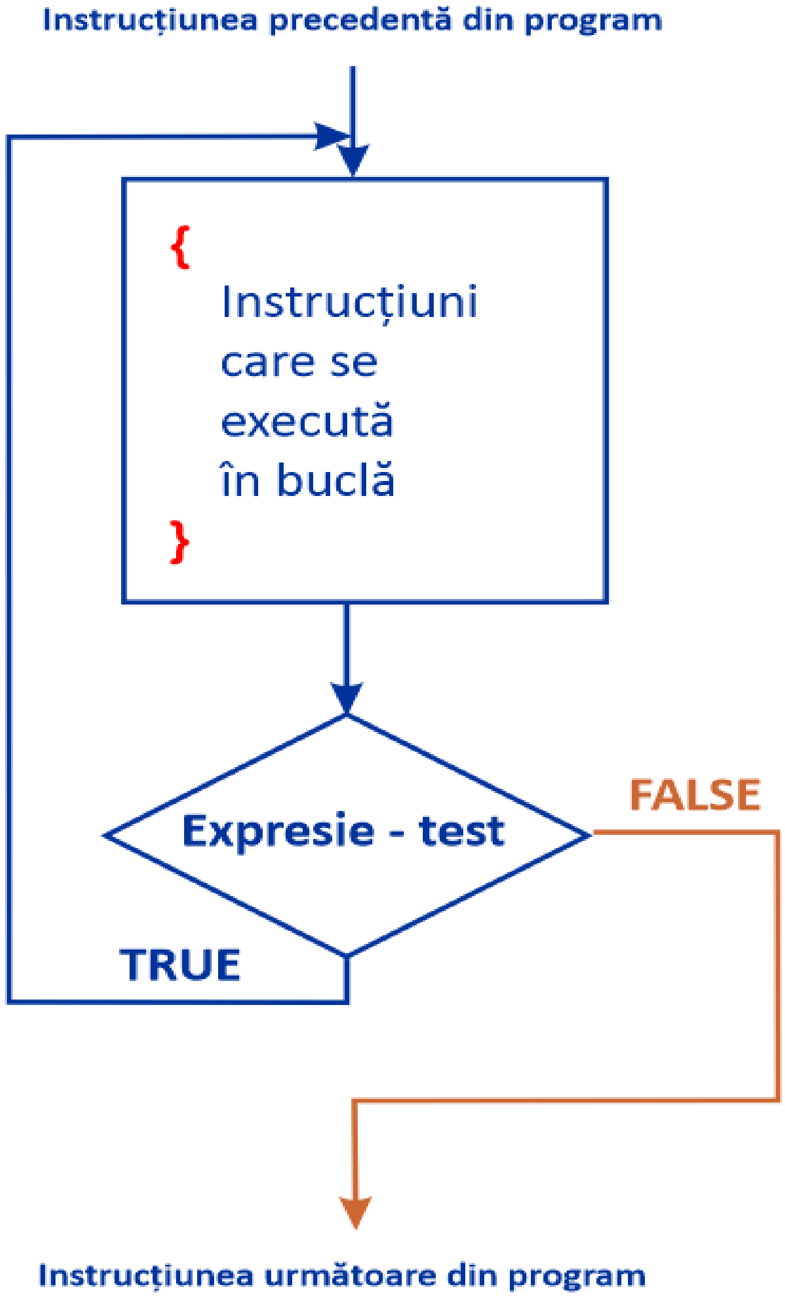
```
#include <iostream>
   using namespace std;
int main()
{
   int n, invers = 0;
   cin >> n;
   while (n)
   {
      inv = inv *10;
      int k = n % 10;
      inv = inv + k;
      n = n / 10;
   }
   cout << inv;
   return 0;
}
```

The

```
do
{



} while (expression)
```

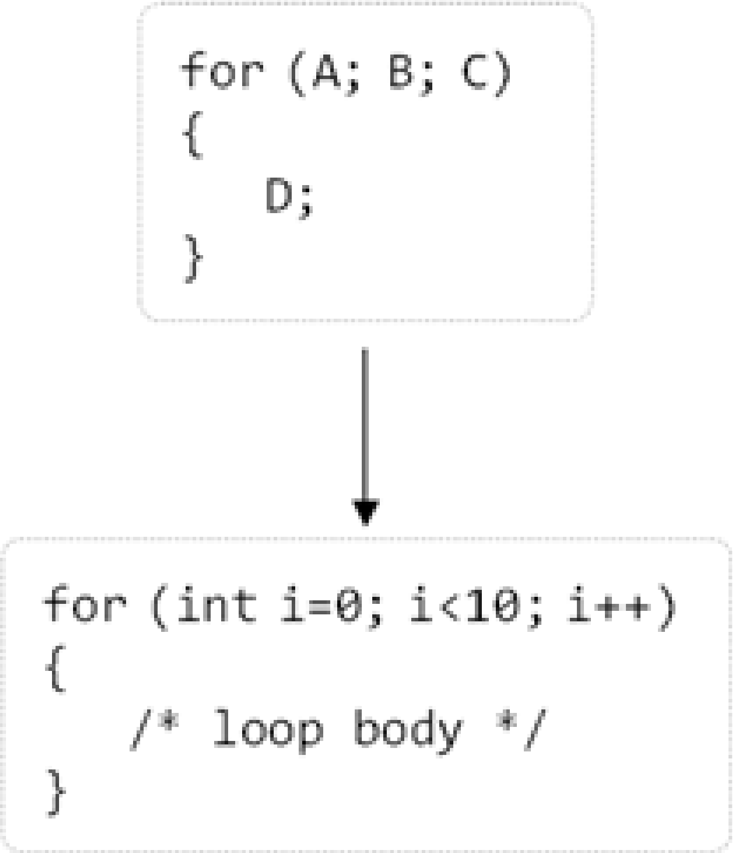

Instrucțiunea precedentă din program
{
Instrucțiuni
care se
execută
în buclă
}
Expresie - test
FALSE
TRUE
Instrucțiunea următoare din program

Example:

Entering numbers until 0 is introduced

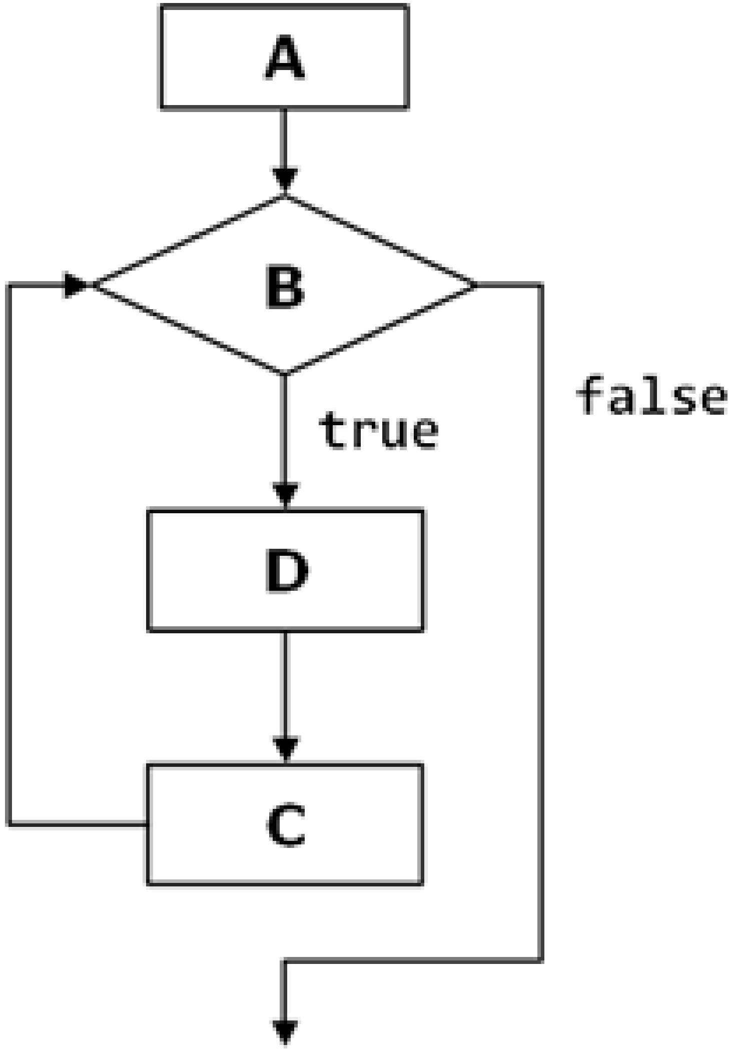
```
#include <iostream>
   using namespace std;
int main()
{
   int n, sum = 0;
   do
   {
      cin >> n;
      sum = sum + n;
   }
   while (n)
   cout << sum;
   return 0;
}
```

Count controlled loops



```
for (A; B; C)
{
    D;
}
```

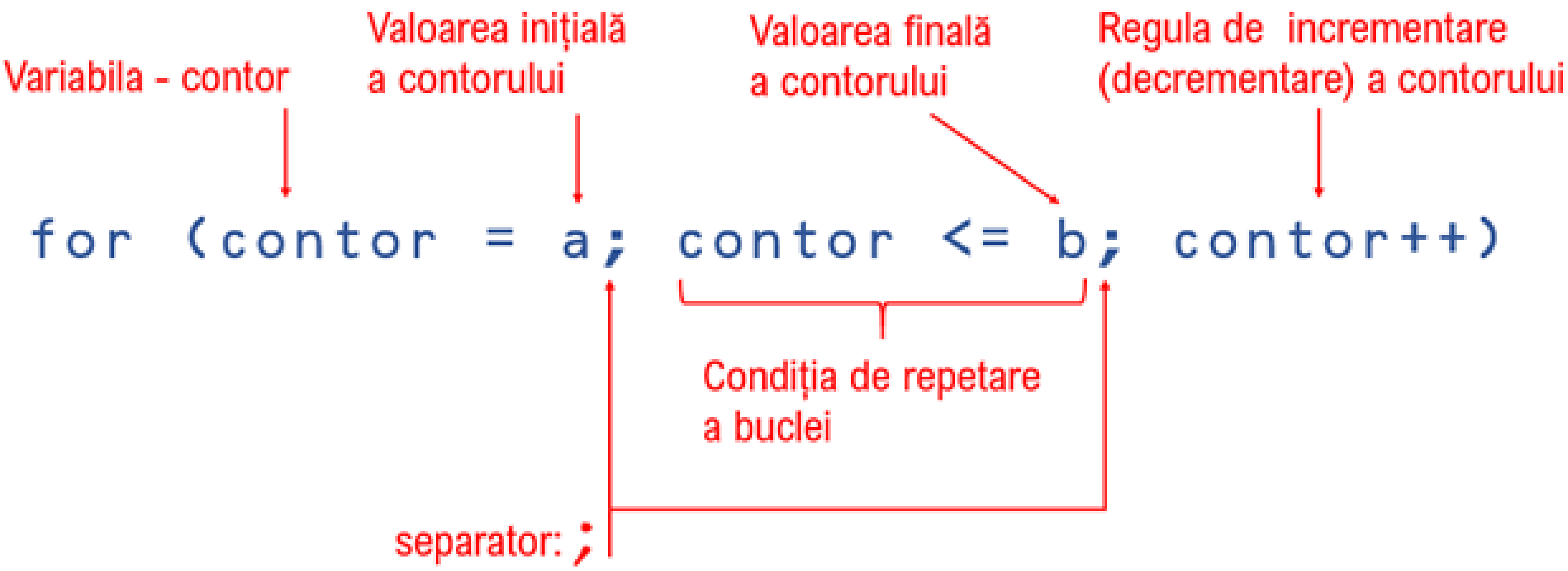
```
for (int i=0; i<10; i++)
{
    /* loop body */
}
```



When use counter controlled loops?

Any time when you have to repeat an action for a number of times which is known before

Count controlled loops



Variabila - contor

Valoarea inițială
a contorului

Valoarea finală
a contorului

Regula de incrementare
(decrementare) a contorului

```
for (contor = a; contor <= b; contor++)
```

Condiția de repetare
a buclei

separator: ;

Examples

Perfect numbers

The sum of divisors of n is equal to n

6 = 1 + 2 + 3

“Lucky” tickets

Six digits numbers, where the sum of odd places digits equal to sum of even places digits

247104

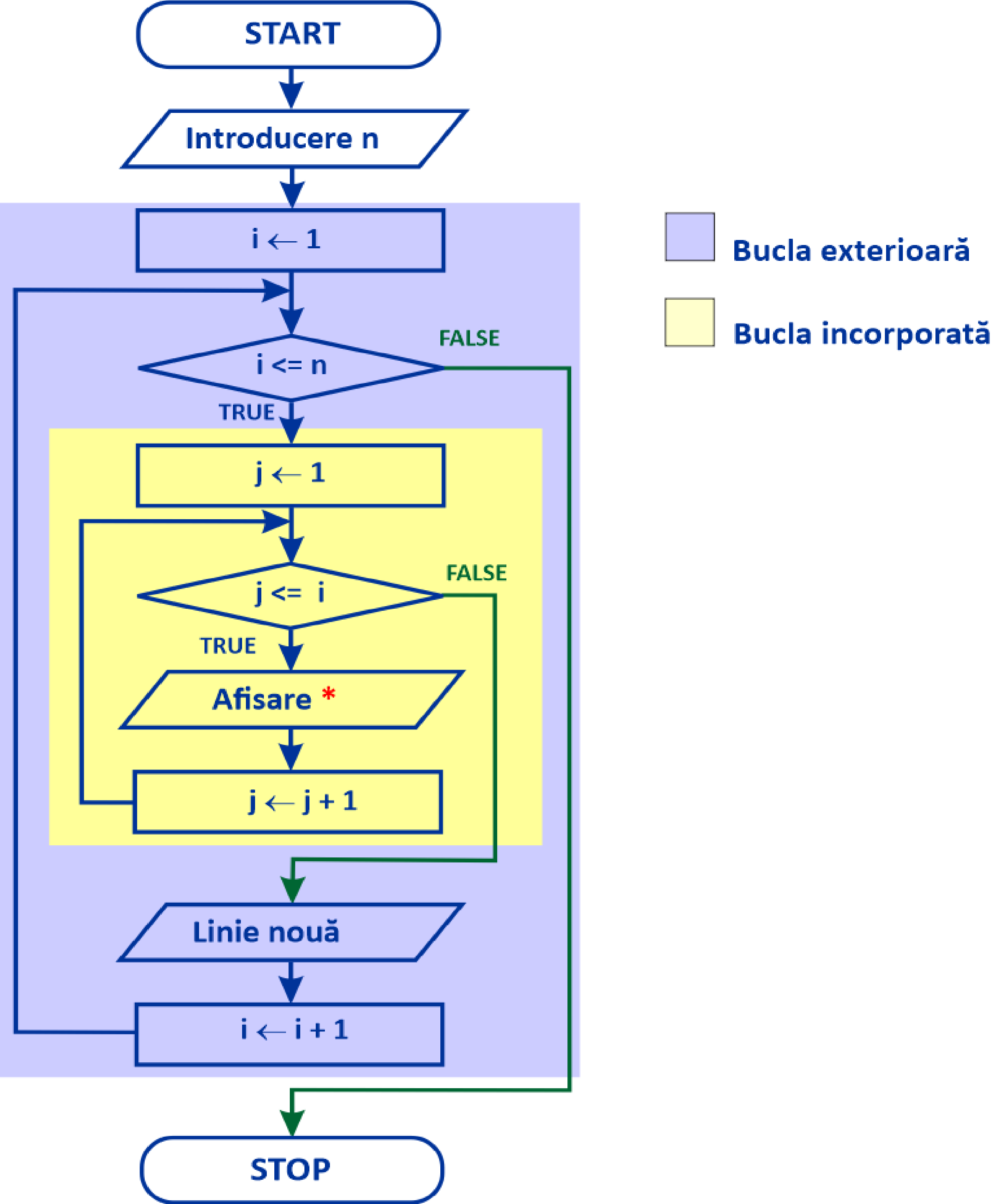
2 + 7 + 0 = 4 + 1 + 4

Nested loops

Write a program to display a triangle of stars of size n (here n = 5):

```
*
**
***
****
*****
```

The scheme



START

Introducere n

i ← 1

i <= n

TRUE

FALSE

j ← 1

j <= i

TRUE

FALSE

Afisare *

j ← j + 1

Linie nouă

i ← i + 1

STOP

▨ Bucla exterioară

□ Bucla incorporată

Exercises

Write a program to calculate:

$1 / 2 + 2 / 3 + 3 / 4 + 4 / 5 + \ldots + N / (N + 1)$

Write a program to calculate:

$+ 2^2 + 3^2 + 4^2 + \ldots + N^2$

Write a program to calculate the n-th element of Fibonacci sequence