Lecture - 12

Programming in C

Flow of Control :

प्रवाह ⟷ नियंत्रण

"To control the flow of execution of lines of code"
(execution → Instruction)

Control Statement :
- Conditional Statement
- Loop Statement

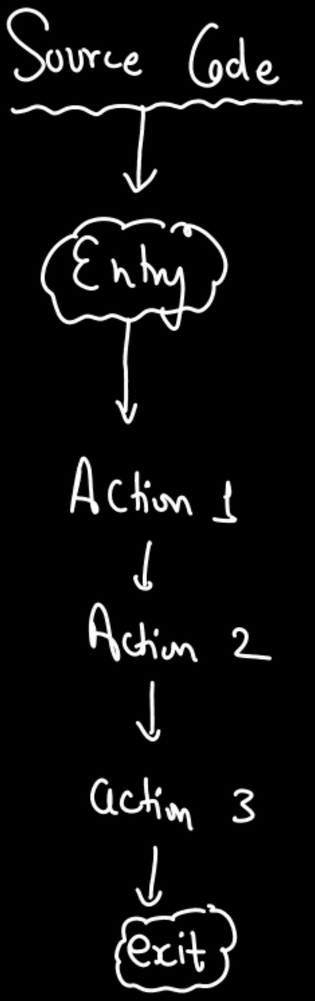
Control Structure : (संरचना)
- Sequence Structure ✓
- Selection Structure
- Loop structure

Sequence Structure :

In this structure the program executes instruction one after other.

```
1  # include <stdio.h>
2  int main() {
3      int a, b, c;
4      a= 10;  b= 20;  c= 30;
5      printf ("%d", a+b+c);
6      return 0;
7  }
```

entry



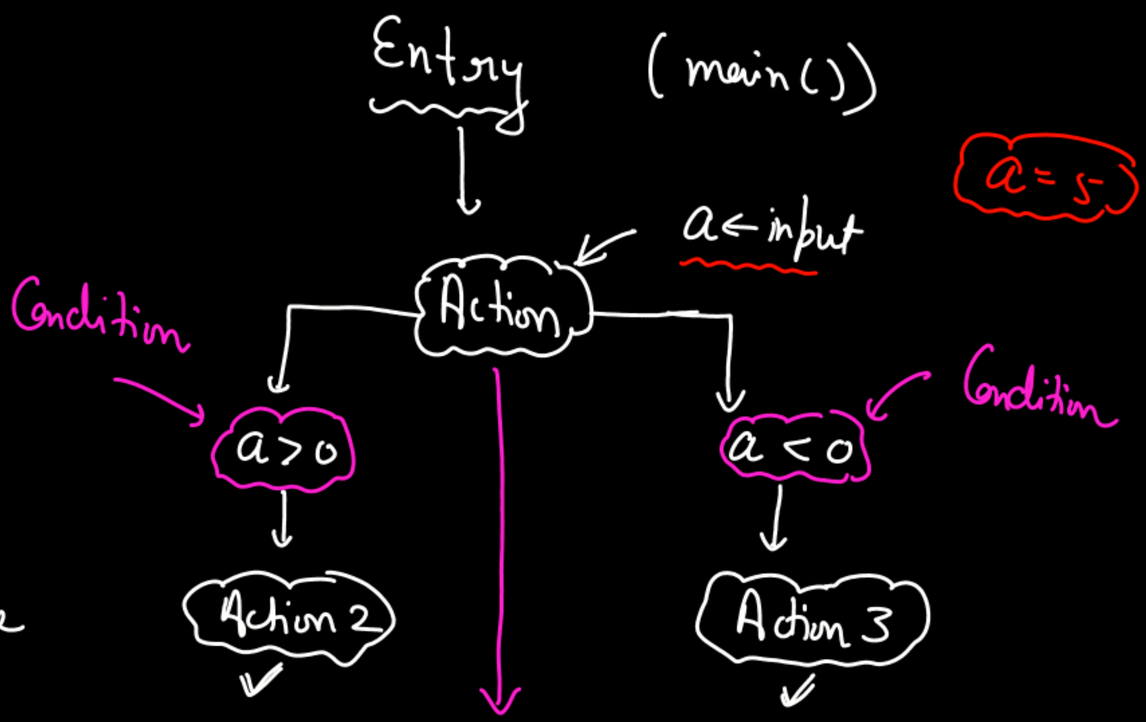
Selection Structure :

It refers to execution of an instruction as per the selected Condition.

- True
- False

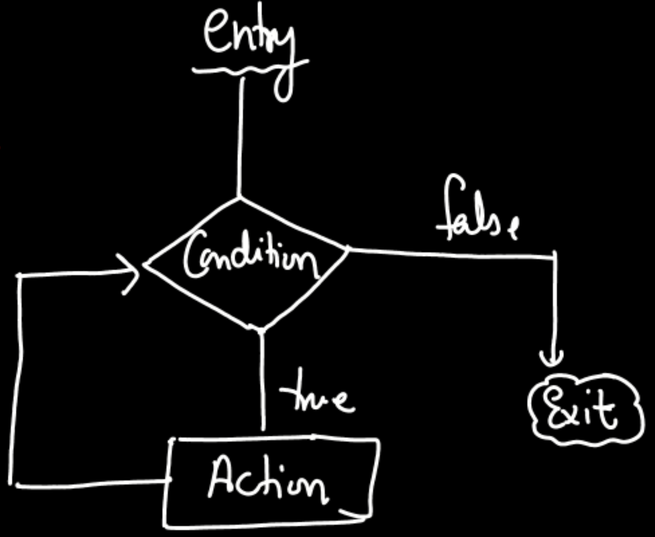
→ the program always flows to the True Statements.

two ways → If - else Statement
→ Switch Case Statement
} Conditional Statements



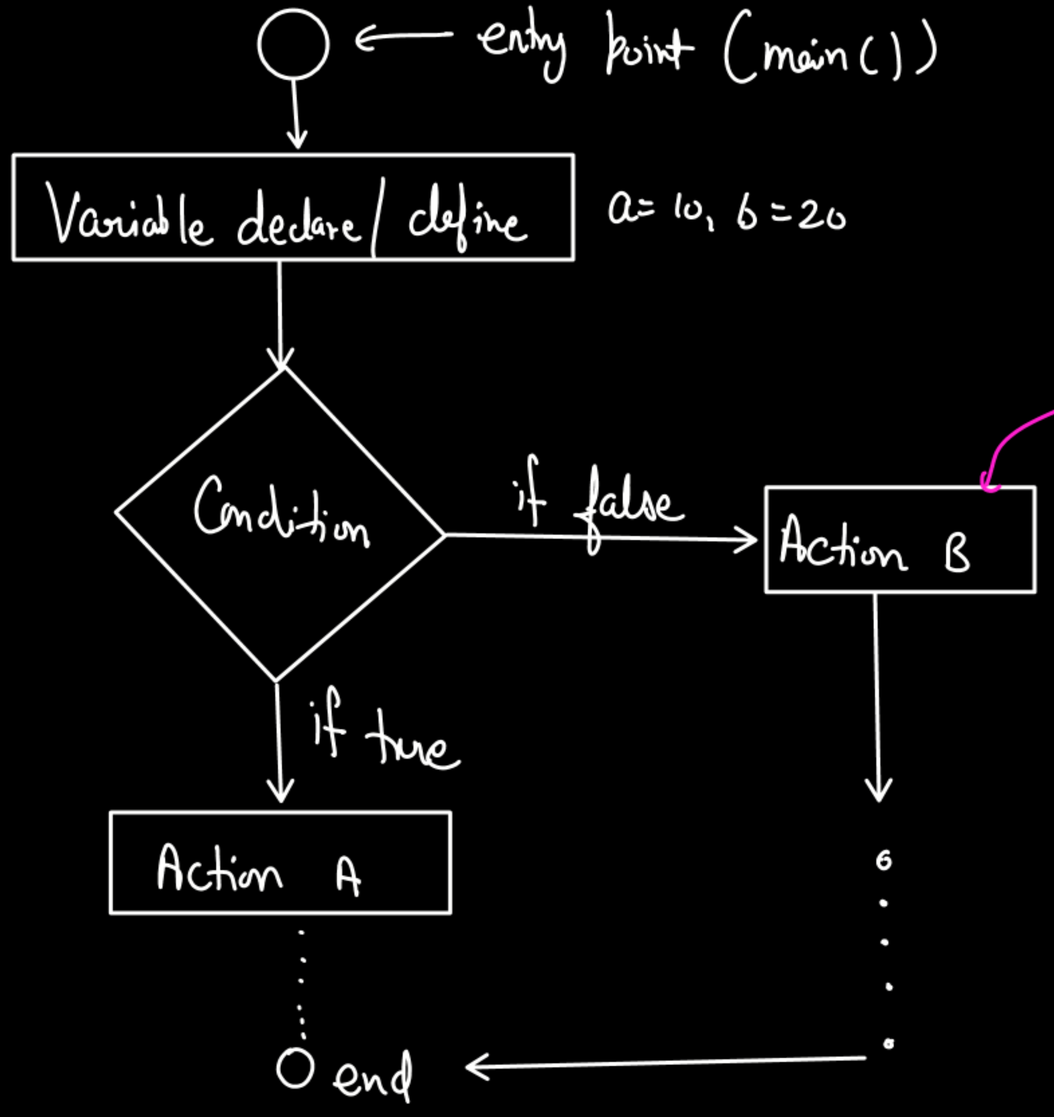
Loop structure :

Loop structure refers to the execution of an instruction in a loop until condition gets false.

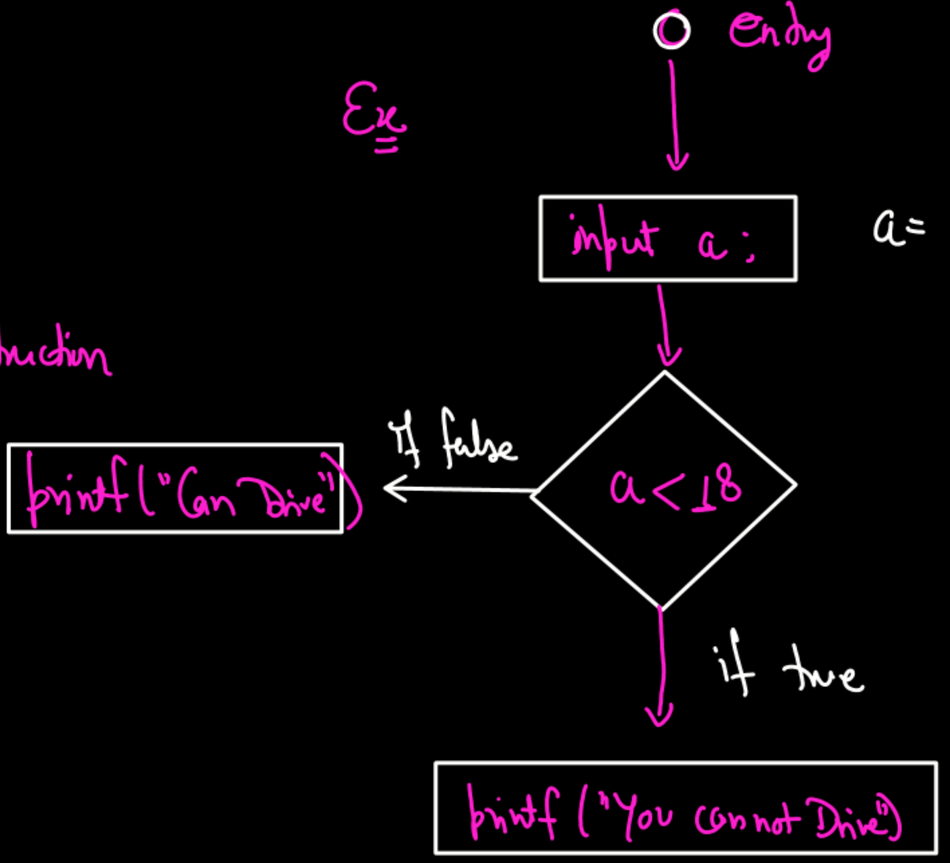


Loop Statement
- while loop
- do while loop
- for loop

flow chart : Graphical representation of data flow / flow of instructions



Ex



Pseudo Code

Conditional Statement :

if ← अगर
else ← वरना

(A) If - Else Statement :

↳ In this structure we can check the condition, if true the execute the block of 'if' otherwise execute the block of 'else'.

⇒ Else is not mendatory / Compulsory (अनिवार्य)

Syntax : format of code for a particular statement.

```
if (Condition) {   ← opening of a block
        Block of if / Body of if
}   ← Closing of the block
else {
        block of else / Body of else
}
```

{ Curly braces is used to define a block

Example: Write a program to check that the number is '+ve' or '-ve'

```
# include<stdio.h>
int main ( ) {
    int number ;
    printf (" Enter a number:") ;
    scanf (" %d", & number ) ;
    if (number >0){
        printf (" Positive Number ") ;
    }
    else {
        printf (" Negative Number");
    }
return 0; }
```

Dry Run

number = 1

Enter a number: 1

Positive Number

1 > 0

number > 0 ?

True

number = -1

Enter a Number: -1

Negative Number.

-1 > 0

number > 0 ?

false

!

else

Write a program to check whether a number is odd or even?

```c
#include <stdio.h>
int main() {
    int num;
    printf("Enter number:");
    Scanf ("%d", &num);
    if (num %2== 0) {
        printf ("Even Number");
    }
    else {
        printf ("odd Number");
    }
    return 0;
}
```

Dry Run

num = 4

Enter a number : 4

Even number

num = 7

Enter a number : 7

Odd Number

%
==

num % 2 == 0 → True = 1

4 % 2 == 0

0 == 0 ?

True

num % 2 == 0 → false

7 % 2 == 0

1 == 0

↳ false = 0