

003 - FLOW CONTROL WITH IF-ELSE

FLOW CONTROL

C Flow Control



- ✓ C if...else
- ✓ C for Loop
- ✓ C while Loop
- ✓ C break and continue
- ✓ C switch...case
- ✓ C Programming goto
- ✓ Control Flow Examples

Control structure

Condition



Control structure

C control structure are divided into two parts:

❖ Decisions making control structure If

- ✓ if
- ✓ If else
- ✓ Else If
- ✓ Nested If
- ✓ Switch(case control structure

❖ Loop control structure

- ✓ For loop
- ✓ While loop
- ✓ Do while loop

Continue, break keywords

If control structure

If control structure

- The if statement evaluates the test expression inside parenthesis.
- If test expression is evaluated to true (nonzero), statements inside the body of if is executed.
- If test expression is evaluated to false (0), statements inside the body of if is skipped.
- Syntax:-

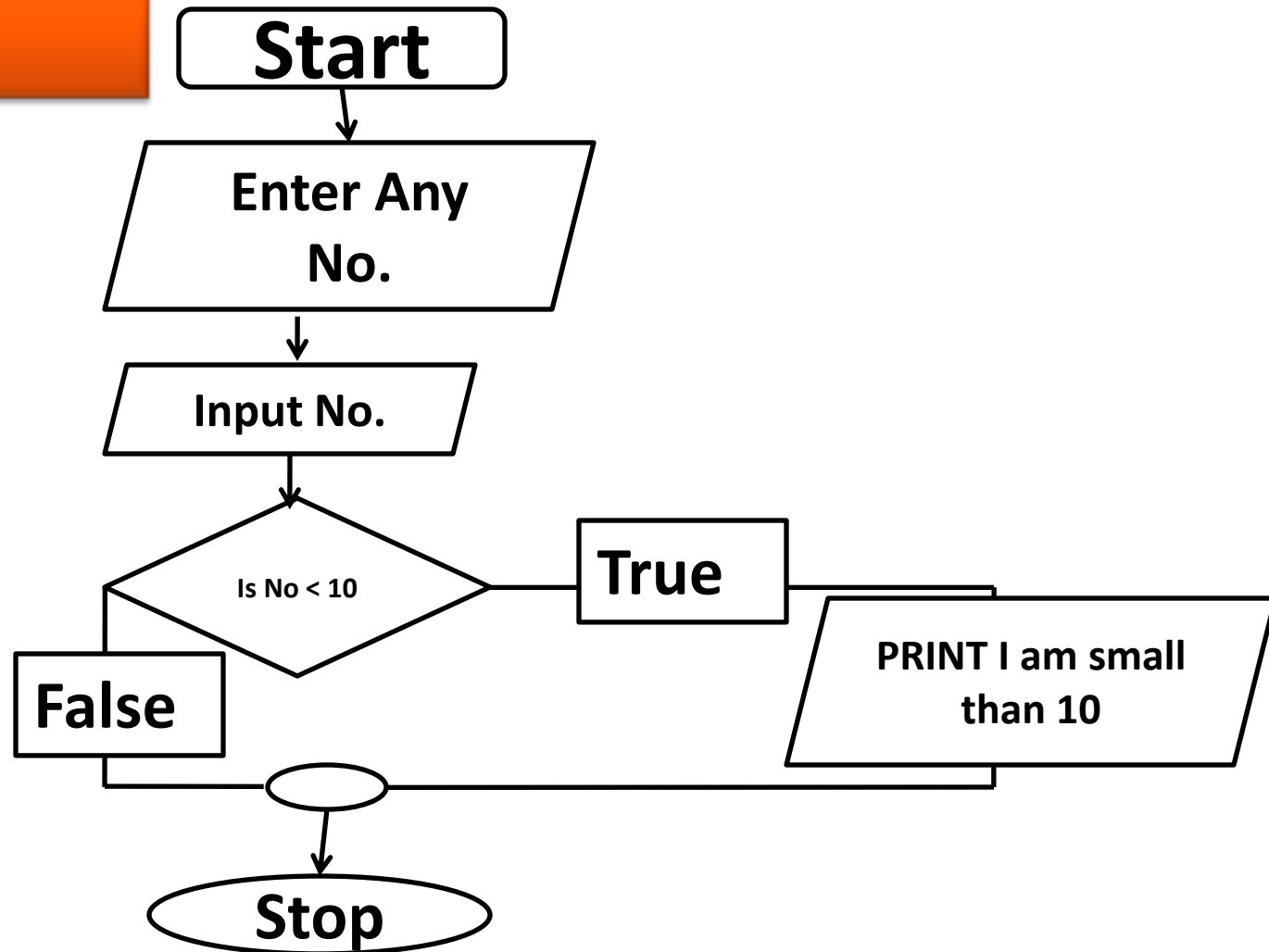
if (Condition)

{

codes inside the body of if

}

Flowchart



if... statement

**Program to display a a if user enters negative a
If user enters positive a, that a won't be displayed**

```
#include<stdio.h>
#include<conio.h>
```

```
void main()
{
```

```
int a;
```

```
printf("Enter Your Age:\n");
scanf("%d", &a);
```

```
if (a > 18)
{
printf("Welcome: %d.\n", a);
}
}
```

If else statement :

if else statement will perform the task even though condition become false.

Syntax:-

if (Condition)

{

codes inside the body of if

}

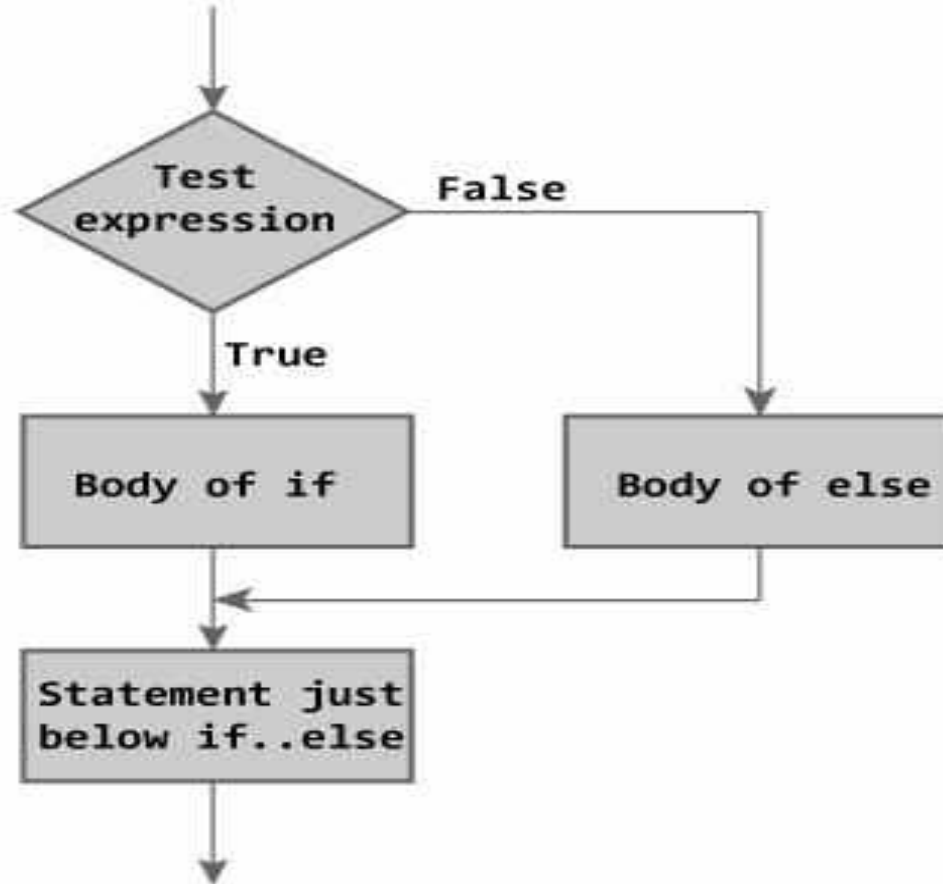
else

{

codes inside the body of else

}

Flowchart of if...else statement



if... else statement

Program to display a if user enters negative a If user enters positive a, that a won't be displayed

```
#include<stdio.h>
#include<conio.h>
void main()
{
int a;
printf("Enter Your Age:\n");
scanf("%d", &a);
```

```
    if (a > 18)
    {
        printf("Welcome\n");
    }
    else
    {
        printf("Not Allowed\n");
    }
    getch();
}
```

if...else statement - to check even or odd

```
void main()
{
int a;
printf("Enter an integer: ");
scanf("%d",&a);
```

```
if( a%2 == 0 )
{
printf("%d is an even integer",a);
}
else
{
printf("%d is an odd integer", a);
} }
```

if...else statement
(if...elseif....else Statement)

Program to check Positive, Negative or Zero Integer

```
#include<stdio.h>
#include<conio.h>
void main()
{
int a;
clrscr();
printf("Enter an integer:\n");
scanf("%d",&a);
if(a>0)
{
printf(" positive");
}
else if(a==0)
{
printf("zero ");
}
else
{
printf("negative");
} getch(); }
```

Program to relate two integers using =, > or <

```
main()  
{  
    int a, b;  
    clrscr();  
    printf("Enter two integers: ");  
    scanf("%d %d", &a, &b);  
    //checks if two integers are equal.  
    if(a == b)  
    {  
        printf("Result: %d = %d",a,b); }  
    //checks if a is greater than b.  
    else if (a > b)  
    {  
        printf("Result: %d > %d",a ,b);  
    }  
    // if both test expression is false  
    else  
    {  
        printf("Result: %d < %d",a ,b);  
    }  
    getch();  
}
```

PROGRAM TO FIND THE BIGGEST OF THREE

```
void main()
{
    int a, b, c;
    clrscr();
    printf("Enter the value for a:\n");
    scanf("%d", &a);
    printf("Enter the value for b:\n");
    scanf("%d", &b);
    printf("Enter the value for c:\n");
    scanf("%d", &c);
```

```
    if ((a > b) && (a > c))
    {
        printf("\n The big one is a= %d", a);
    }
    else if (b > c)
    {
        printf("\n The big one is b= %d", b);
    }
    else
    {
        printf("\n The big one is c= %d", c);
    } }
```

Nested IF statement

Nested If:

Syntax:

putting of if else
statement
within the body
of the if
statement or
the body of an
else statement
this is called
nested if.

```
if (condition1)
{
    if(condition2)
    {
        /* do this*/
    }
    else
    {
        /*do this*/
    }
    else
    {
        /*do this*
    }
}
```

Nested if Login & Password

```
void main()
{
char username;
int password;
clrscr();
printf("Username:");
scanf("%c",&username);

if(username=='a')
{
printf("Password:");
scanf("%d",&password);
if(password==12345)
{
printf("Login successful");
}
}
else
{
printf("User name does not exists, Try again.");
}
};
```

Nested if Login & Password

```
void main()
{
char username;
int password;
clrscr();
printf("Username:");
scanf("%c",&username);
```

```
if(username=='a')
{
    printf("Password:");
    scanf("%d",&password);
    if(password==12345)
    {
        printf("Login successful");
    }
    else
    {
        printf("Incorrect Password");
    }
}
else
{
    printf("User name does not exists, Try again.");
}
};
```

Switch statement

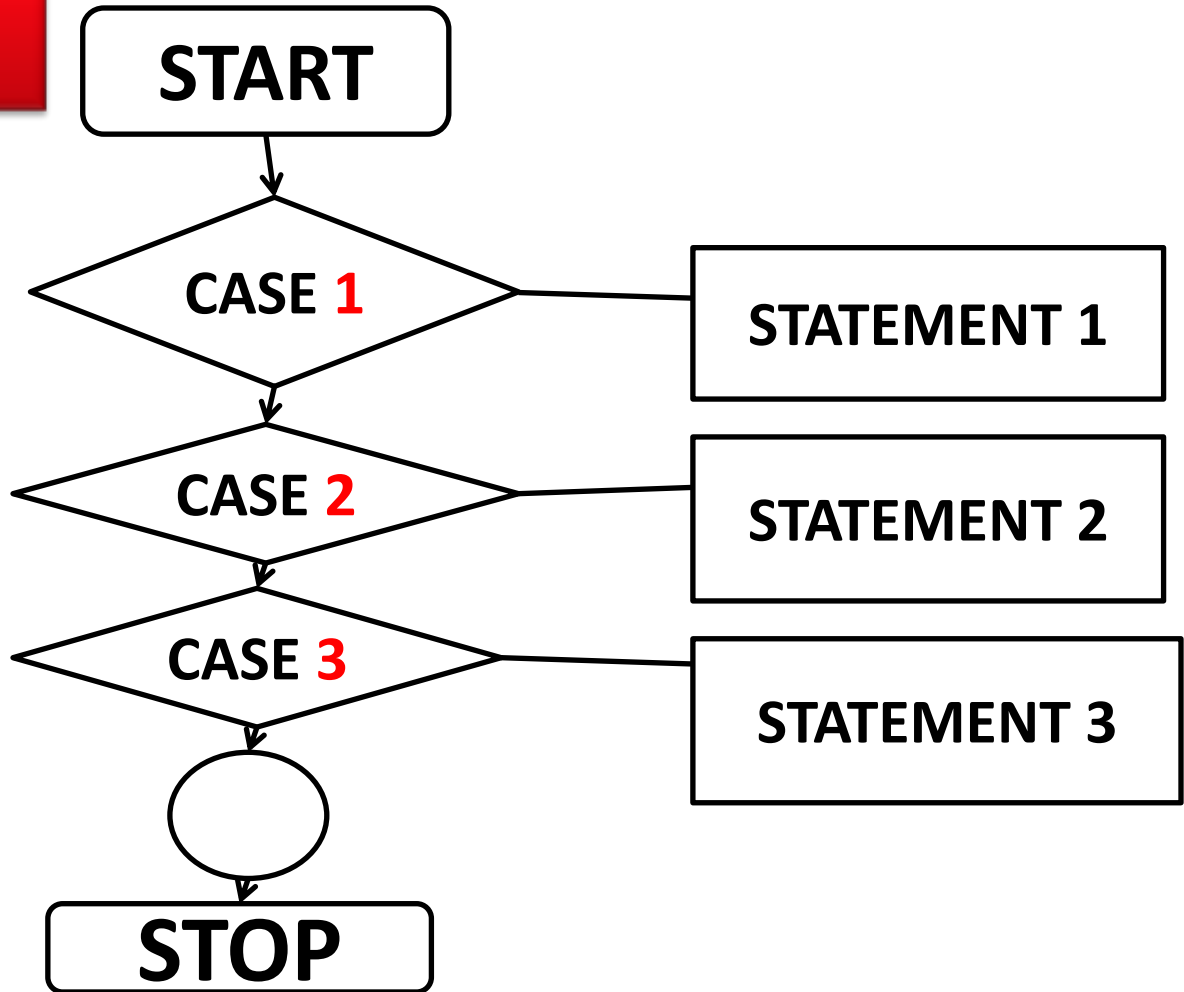
Switch statement

- The control statement that allow us to make a decision from the choice given by user is called a **switch Statement**.

Syntax:

```
Switch(integer expressions)
{
    Case 1: do this;
    Break;
    Case 2: do this;
    Break;
    Case 3: do this;
    Break;
    default: do this;
}
```

Switch case:



Switch Statement for (+, -, x, ÷)

```
void main()  
{
```

```
int a,b,c,i;
```

```
printf("Press 1 for Addition\n Press 2 for Subtraction\n  
Press 3 for Multiplication\n Press 4 for Division\n");
```

```
scanf("%d",&i);
```

```
switch(i)  
{
```

case 1:

```
printf("Enter two numbers for Addition\n");
```

```
scanf("%d%d",&a,&b);
```

```
c=a+b;
```

```
printf("Addition is:%d\n",c);
```

```
break;
```

case 2:

```
printf("You are at Case 2- Subtraction\n");  
printf("enter two numbers\n");  
scanf("%d%d",&a,&b);  
c=a-b;  
printf("Subtraction is:%d\n",c);  
break;
```

case 3:

```
printf("You are at Case 3- Multiplication\n");  
printf("enter two numbers\n");  
scanf("%d%d",&a,&b);  
c=a*b;  
printf("Multiplication is:%d\n",c);  
break;
```

case 4:

```
printf("You are at Case 4- Division\n");  
printf("enter two numbers\n");  
scanf("%d%d",&a,&b);  
c=a/b;  
printf("Answer is:%d\n",c);  
break;
```

default:

```
printf("Invalid Selection Enter a valid Choice");  
}  
getch();  
}
```

Goto statement

Goto statement

```
void main()  
{  
    printf(" go to statement ");  
    goto x;  
y:  
    printf(" unconditional jump statement ");  
    goto z;  
x:  
    printf(" is a ");  
    goto y;  
z:  
    printf(" it make program hard to understand ");  
}
```

Goto statement

- ✦ Control may be transferred to anywhere within the current function .
- ✦ The target statement must be labeled , and a colon must follow the label.
- ✦ NOTE :
- ✦ use of goto statement is highly discouraged in any programming language because
- ✦ it makes difficult to trace the control flow of a program hard to understand and hard to modify .

Thank You!

The image features the text "Thank You!" in a bold, white, three-dimensional sans-serif font. The text is centered and appears to be floating above or attached to a dynamic, multi-colored brushstroke. The brushstroke is composed of numerous overlapping, curved strokes in a spectrum of colors including red, orange, yellow, green, blue, and purple, creating a sense of movement and energy. The entire graphic is set against a plain white background.