Programming in C

(Decision making – if–else)

Prepared By

Alok Haldar

Assistant professor

Department of Computer Science & BCA

Kharagpur College

Decision making in C

Decision making is about deciding the order of execution of statements based on certain conditions or repeat a group of statements until certain specified conditions are met. C language handles decision-making by supporting the following statements,

- `if` statement
- `switch` statement
- conditional operator statement (`? :` operator)
- `goto` statement

Decision making with `if` statement

The `if` statement may be implemented in different forms depending on the complexity of conditions to be tested. The different forms are,

1. Simple `if` statement
2. `if....else` statement
3. Nested `if....else` statement
4. Using `else if` statement

Simple `if` statement

The general form of a simple `if` statement is,

```
if(expression)
{
    statement inside;
}
    statement outside;
```

If the *expression* returns true, then the **statement-inside** will be executed, otherwise **statement-inside** is skipped and only the **statement-outside** is executed.

Example:

```
#include <stdio.h>

void main( )
{
    int x, y;
    x = 15;
    y = 13;
    if (x > y )
    {
        printf("x is greater than y");
    }
}
```

Output:

x is greater than y

`if...else` statement

The general form of a simple `if...else` statement is,

```
if(expression)
{
    statement block1;
}
else
{
    statement block2;
}
```

If the *expression* is true, the **statement-block1** is executed, else **statement-block1** is skipped and **statement-block2** is executed.

Example:

```
#include <stdio.h>

void main( )
{
    int x, y;
    x = 15;
    y = 18;
    if (x > y )
    {
        printf("x is greater than y");
    }
    else
    {
        printf("y is greater than x");
    }
}
```

Output:

y is greater than x

Nested `if....else` statement

The general form of a nested `if...else` statement is,

```
if( expression )
{
    if( expression1 )
    {
        statement block1;
    }
    else
    {
        statement block2;
    }
}
else
```

```
{
    statement block3;
}
```

if *expression* is false then **statement-block3** will be executed, otherwise the execution continues and enters inside the first `if` to perform the check for the next `if` block, where if *expression 1* is true the **statement-block1** is executed otherwise **statement-block2** is executed.

Example:

```
#include <stdio.h>

void main( )
{
    int a, b, c;
    printf("Enter 3 numbers...");
    scanf("%d%d%d",&a, &b, &c);
    if(a > b)
    {
        if(a > c)
        {
            printf("a is the greatest");
        }
        else
        {
            printf("c is the greatest");
        }
    }
    else
    {
        if(b > c)
        {
            printf("b is the greatest");
        }
        else
        {
            printf("c is the greatest");
        }
    }
}
```

`else if` ladder

The general form of else-if ladder is,

```
if(expression1)
{
    statement block1;
}
else if(expression2)
{
    statement block2;
}
else if(expression3 )
{
    statement block3;
}
else
    default statement;
```

The expression is tested from the top(of the ladder) downwards. As soon as a **true** condition is found, the statement associated with it is executed.

Example :

```
#include <stdio.h>

void main( )
{
    int a;
    printf("Enter a number...");
    scanf("%d", &a);
    if(a%5 == 0 && a%8 == 0)
    {
        printf("Divisible by both 5 and 8");
    }
    else if(a%8 == 0)
    {
        printf("Divisible by 8");
    }
    else if(a%5 == 0)
    {
        printf("Divisible by 5");
    }
    else
    {
        printf("Divisible by none");
    }
}
```

The Coditional or Ternary Operator (? :)

We have covered **conditional operator ? :** in the previous chapter which can be used to replace **if...else** statements. It has the following general form −

```
variable = Exp1 ? Exp2 : Exp3;
```

Where Exp1, Exp2, and Exp3 are expressions. Notice the use and placement of the colon.

The value of a ? expression is determined like this −

- Exp1 is evaluated. If it is true, then Exp2 is evaluated and becomes the value of the entire ? expression.
- If Exp1 is false, then Exp3 is evaluated and its value becomes the value of the expression.

It can be visualized into if-else statement as:

```
if(Expression1)
{
    variable = Expression2;
}
else
{
    variable = Expression3;
}
```

Since the Conditional Operator ‘?:’ takes three operands to work, hence they are also called **ternary operators**.

Example :

// C program to find largest among two

// numbers using ternary operator

```
#include <stdio.h>


int main()
{
        int m = 5, n = 4;


        (m > n) ? printf("m is greater than n that is %d > %d",
                                        m, n)
                : printf("n is greater than m that is %d > %d",
                                        n, m);


        return 0;
}
```

Points to Remember

1. In `if` statement, a single statement can be included without enclosing it into curly braces `{ ... }`

   ```
   int a = 5;
   if(a > 4)
       printf("success");
   ```

   No curly braces are required in the above case, but if we have more than one statement inside `if` condition, then we must enclose them inside curly braces.

2. `==` must be used for comparison in the expression of `if` condition, if you use `=` the expression will always return **true**, because it performs assignment not comparison.
3. Other than **0(zero)**, all other values are considered as **true**.

   ```
   if(27)
       printf("hello");
   ```

   In above example, **hello** will be printed.

List of program on decision making statement :

1. Write a C program to accept two integers and check whether they are equal or not.
Test Data : 5 5
Expected Output :
Number1 and Number2 are equal

2. Write a C program to check whether a given number is even or odd.
Test Data : 7
Expected Output :
7 is an odd integer

3. Write a C program to check whether a given number is positive or negative.
Test Data : 5
Expected Output :
5 is a positive number

4. Write a C program to find whether a given year is a leap year or not.
Test Data : 2016
Expected Output :
2016 is a leap year.

5. Write a C program to read the age of a candidate and determine whether it is eligible for casting his/her own vote.

Test Data : 21
Expected Output :
Congratulation! You are eligible for casting your vote.

6. Write a C program to find the largest of three numbers.
Test Data : 12 35 57
Expected Output :
1st Number = 12, 2nd Number = 35, 3rd Number = 57
The 3rd Number is the greatest among three

7. Write a C program to find the eligibility of admission for a professional course based on the following criteria:
Eligibility Criteria : Marks in Maths >=65 and Marks in Phy >=55 and Marks in Chem>=50 and Total in all three subject >=190 or Total in Maths and Physics >=140

Input the marks obtained in Physics : 65

Input the marks obtained in Chemistry :51

Input the marks obtained in Mathematics :72

Total marks of Maths, Physics and Chemistry : 188

Total marks of Maths and Physics : 137

The candidate is not eligible.
Expected Output :
The candidate is not eligible for admission.

8. Write a C program to calculate the root of a Quadratic Equation.
Test Data : 1 5 7
Expected Output :
Root are imaginary;
No solution.

9. Write a C program to read temperature in centigrade and display a suitable message according to temperature state below :
Temp < 0 then Freezing weather
Temp 0-10 then Very Cold weather
Temp 10-20 then Cold weather
Temp 20-30 then Normal in Temp
Temp 30-40 then Its Hot
Temp >=40 then Its Very Hot
Test Data :
42
Expected Output :
Its very hot.

10. Write a C program to check whether a triangle is Equilateral, Isosceles or Scalene.
Test Data :
50 50 60
Expected Output :

This is an isosceles triangle.

11. Write a program in C to calculate and print the Electricity bill of a given customer. The customer id., name and unit consumed by the user should be taken from the keyboard and display the total amount to pay to the customer. The charge are as follow :

Unit	Charge/unit
upto 199	@1.20
200 and above but less than 400	@1.50
400 and above but less than 600	@1.80
600 and above	@2.00

If bill exceeds Rs. 400 then a surcharge of 15% will be charged and the minimum bill should be of Rs. 100/-

Test Data :
1001
Soumya
800
Expected Output :
Customer IDNO :1001
Customer Name : Soumya
unit Consumed :800
Amount Charges @Rs. 2.00 per unit : 1600.00
Surchage Amount : 240.00
Net Amount Paid By the Customer : 1840.00

12. Write a program in C to accept a grade and declare the equivalent description :

Grade	Description
E	Excellent
V	Very Good
G	Good
A	Average
F	Fail

Test Data :
Input the grade :A
Expected Output :
You have chosen : Average

Question ??

Thank You.