

Title: Decision making and branching in C

Objective:

The main objectives of this lab are to

- Learn about how to display and compare Maximum Minimum Number.
- Learn about how to code Grade.
- Learn about how to code Grade using switch.

Theory:

In C programming it support sequential program statements which execute one statement immediately after another. Here the flow is sequential it never change the flow of control from the next line. The compiler executes the program sequentially in the order which they appear. This happens when there are no options and if the repeated steps are not needed. But there are another option when we need the options and when we want to use the selective steps repeatedly. In many situations we may have to change the order of execution and want to repeat certain steps based on any condition or until the condition meets, which means decision making. C supports much decision making or control statements.

The main important decision making statements in c are as follows.

- If statement
- Switch statement

If statement

‘If’ is the most powerful decision making statement in C language. Thus it is used to control the execution of statements. ‘If’ is used along with an expression. It is a two way branching or decision making statement.

Syntax :

```
if(test expression)
{
    Body of if;
}
```

Using an 'if' a computer first evaluate the expression and if it is true the body of if is executed which means the flow of control passes to the if's body, or if the test condition is false then the flow of control passes to the immediate step next to if's body. If there are more than one step in the body of 'if' they are normally written with in brackets '{ }' to denote it is the body of 'if'.

The test expression will return a Boolean value which is a '0' or '1'. Here 1 is 'true' and 0 denotes it is 'false'.

Depending upon the complexity and conditions to be tested if can be further subdivided.

- Simple if
- if.....else
- nested if...else
- if...else...if ladder

Simple if statement

The general syntax of a simple if statement is

```
if(test expression)
{
    Body of if;
}
```

If else if statement

An extension to simple if is called if.. else statement.

Syntax:

```
if(test expression)
{
    Body of if;
}
else
```

```
{  
    Body of else;  
}
```

Nested if statement

It is used when a series of decisions are involved and have to use more than one if...else statement.

Syntax :

```
if(condition 1)  
{  
    if(condition 2)  
    {  
        body of if;  
    }  
    else  
    {  
        Body of else;  
    }  
}  
else  
{  
    Body of else;  
}
```

The if... else... if ladder

To take multipath decisions or chain of ifs then we use the if ...else...if ladder. It is in the following general form

```
if(condition 1)  
    statement 1;  
else if(condition 2)  
    statement 2;  
else if(condition 3)
```

```
statement 3;  
else  
    statement 4;
```

The switch statement

If is a conditional statement where 'switch' is a selection statement which means if we want to select one from many alternatives we use 'switch'. The selective statement 'switch' is a multi way decision statement, that is using a single expression we can direct the flow of control to multiple paths. 'Switch' statement tests the value of given expression or variable against a list of case values and when a match is found that block of statements associated with that case is executed.

Syntax :

```
switch(expression)  
{  
    case constant1 :  
        block1;  
        break;  
    case constant2 :  
        block2;  
        break;  
    case constant n:  
        block n;  
        break;  
        .  
        .  
        .  
    default:  
        default block;  
        break;  
}  
  
    printf("wrong choice");  
    break;  
}
```

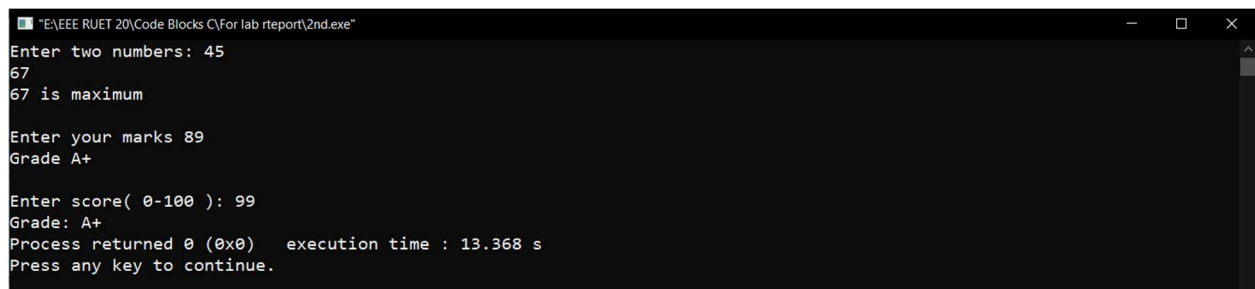
Source Code:

```
1. /// Maximum Minimum Number
2.
3. #include <stdio.h>
4.
5. int main()
6. {
7.     int num1, num2;
8.
9.     //Input two numbers from user
10.    printf("Enter two numbers: ");
11.    scanf("%d%d", &num1, &num2);
12.
13.    //If num1 is maximum
14.    if(num1 > num2)
15.    {
16.        printf("%d is maximum", num1);
17.    }
18.
19.    //If num2 is maximum
20.    if(num2 > num1)
21.    {
22.        printf("%d is maximum", num2);
23.    }
24.
25.    //Additional condition check for equality
26.    if(num1 == num2)
27.    {
28.        printf("Both are equal");
29.    }
30.
31.    printf("\n\n");
32.
33.    /// Display Grade of a Student
34.
35.    int marks;
36.    printf("Enter your marks ");
37.    scanf("%d", &marks);
38.    if(marks<0 || marks>100)
39.    {
40.        printf("Wrong Entry");
```

```
41.     }
42.     else if(marks<40)
43.     {
44.         printf("Grade F");
45.     }
46.     else if(marks>=40 && marks<50)
47.     {
48.         printf("Grade D");
49.     }
50.     else if(marks>=50 && marks<60)
51.     {
52.         printf("Grade B");
53.     }
54.     else if(marks>=60 && marks<70)
55.     {
56.         printf("Grade A-");
57.     }
58.     else if(marks>=70 && marks<80)
59.     {
60.         printf("Grade A");
61.     }
62.     else
63.     {
64.         printf("Grade A+");
65.     }
66.
67.     printf("\n\n");
68.
69.     /// Display Grade of a Student using Switch Statement
70.
71.     int score;
72.
73.     printf("Enter score( 0-100 ): ");
74.     scanf("%d", &score);
75.
76.     switch( score / 10 )
77.     {
78.
79.     case 10:
80.     case 9:
81.         printf("Grade: A+");
82.         break;
83.
84.     case 8:
85.         printf("Grade: A");
```

```
86.         break;
87.
88.     case 7:
89.         printf("Grade: B");
90.         break;
91.
92.     case 6:
93.         printf("Grade: C");
94.         break;
95.
96.     case 5:
97.         printf("Grade: D");
98.         break;
99.
100.    default:
101.        printf("Grade: F");
102.        break;
103.
104.    }
105.    return 0;
106. }
107.
```

Output:



```
E:\EEE RUET 20\Code Blocks C\For lab rreport\2nd.exe
Enter two numbers: 45
67
67 is maximum

Enter your marks 89
Grade A+

Enter score( 0-100 ): 99
Grade: A+
Process returned 0 (0x0)   execution time : 13.368 s
Press any key to continue.
```

Discussion and Conclusion:

In this program, at first I compare which one is maximum or minimum by taking numbers from the users. In the second I take a student exam number from the user and display his/her grade. In the last program I done the previous tasks but this time I use switch statement.

