

Understanding C Basics-II *

K.M.Shafi[†]

December 9, 2016

1 Decision Statements

Decision (conditional) statements in C are used to change the normal (sequential) flow of the program determined by the trueness of condition provided in the decision statement. C provides different decision statements like **if** statement and its variants, **switch** statement and **ternary** operator.

2 if

The basic syntax of **if** statement is :

```
if(<condition >)  
{  
    <valid C Statements>  
}
```

<condition> above can be a relational, logical or an arithmetic expression or a simple constant value. Statements within the *if* will get executed if the condition yields true value otherwise they will be ignored. The body of if is always enclosed in braces if it has more than one statement otherwise it is optional.

3 if else

The basic syntax of **if else** statement is :

*This tutorial is originally intended for the PGDCA students batch 2015-16 of DDE, KU

[†]mshafi710@gmail.com

```

if(<condition >)
{
    <valid C Statements>
}
else
{
    <valid C Statements>
}

```

Statements within the **if** will get executed if the condition yields true value otherwise they will be ignored and the statements within **else** part will get executed. The body of if/else is always enclosed in braces if it has more than one statement otherwise it is optional.

4 else if ladder

```

if(<condition >)
{
    <valid C Statements>
}
else if(<condition >)
{
    <valid C Statements>
}
else if(<condition >)
{
    <valid C Statements>
}
else
{
    <valid C Statements>
}

```

Statements within the **if** will get executed if the condition yields true value otherwise they will be ignored and the condition within **else if** part will be checked. if that evaluates to true the statements within that very part gets executed, otherwise, the statements within the else part will get executed.

Example below demonstrates the usage of if, if else and else if ladder.

```

/* Checking whether entered no is odd*/
int main()
{

```

```

    int x;
    printf(" Enter x:");
    scanf("%d",&x);
    if (x%2==1)
    {
        printf("%d is odd",x);
    }
    return 0;
}

```

if the user enters, for instance, 10 as value of x, the output will be nothing and accordingly if he enters 11 the output will be '*11 is odd*'.

```

/* Checking whether entered no is odd or even*/
int main()
{
    int x;
    printf(" Enter x:");
    scanf("%d",&x);
    if (x%2==1)
    {
        printf("%d is odd",x);
    }
    else
    {
        printf("%d is even",x);
    }
    return 0;
}

```

if the user enters, for instance, 10 as value of x, the output will be '*10 is even*' and accordingly if he enters 11 the output will be '*11 is odd*'.

```

/* Checking grade for an entered percentage using else if ladder*/
int main()
{
    int percentage;
    printf(" Enter percentage:");
    scanf("%d",&percentage);

    if (percentage >= 75)
    {
        printf(" Distinction!");
    }
}

```

```

    }
    else if (percentage >= 60)
    {
        printf("Grade A");
    }
    else if (percentage >= 45)
    {
        printf("Grade B");
    }
    else
    {
        printf("Fail!");
    }
    return 0;
}

```

if the user enters, for instance, 47 as value of percentage, the output will be 'Grade B' as the condition *percentage* >= 75 of first if fails, then condition *percentage* >= 60 also fails, and condition *percentage* >= 45 evaluates to true. The else part will get executed only if all the previous conditions fail.

5 switch

Switch case statement is an alternate to else if ladder and offer a convenient way for selection logic i.e., when the programmer has to select a option from multiple options based on some condition or criteria.

```

switch (expression)
{
    case <case_value>:
        <valid C Statements>
    case <case_value>:
        <valid C Statements>
    case <case_value>:
        <valid C Statements>
    default:
        <valid C Statements>
}

```

Here the expression which is restrictively an integer expression, represents the criteria or condition which is evaluated and matched to the given case values. Then if there is a match, the block of statements attached to that case is executed and if there is no break statement at the end of that case, subsequent case statements also get executed till break statement is encountered or end of switch statement. If there is no match to the integer, got through evaluation of an expression, the default case (if provided) will get executed.

```
/* Demo of switch case statement*/
int main()
{
    int option;
    printf("Enter option from 1 to 3 or any thing to exit");
    scanf("%d",&option);

    switch(option)
    {
        case 1:
            printf("Start New Game!");
            break;
        case 2:
            printf("Start Saved Game");
            break;
        case :
            printf("Controls");
            break;
        default :
            printf("Exit!");
    }
    return 0;
}
```