

Introduction to Computers and C language

Dr. Ashwaq Alabaich

ashwakalabaichi2007@yahoo.com

ashwaq.alabaichi[@gmail.com](mailto:ashwaq.alabaichi@gmail.com)

Iteration Statements

In C, and all other modern programming languages, iteration statements (also called loops) allow a set of instructions to be repeatedly executed until a certain condition is reached. C provides three iteration structures: while, do/while, and for loop

The while Loop

The general form of the while loop is
`while(condition) statement;`

where *statement* is either a single statement, or a block of statements. The *condition* may be any expression, and true is any nonzero value. The loop iterates while the condition is true. When the condition becomes false, program control passes to the line of code immediately following the loop.

Write C program to print the numbers from 0...3 using while

```
#include<stdio.h>

main()
{
    int i;
    i = 1;

    while (i!=4)
    {
        printf("\a%d ", i);
        i++;
    }

    printf(" Go!\n");
}
```

Write C program to find the average of n numbers

```
#include<stdio.h>
#include<conio.h>
main()
{
    int n, i;
    float av , s, a;
    s=0;
    printf("enter : n\n");
    scanf("%d",&n);
    i=1;
    while (i<=n)
    {
        printf("enter the numbers: a\n");
        scanf("%f",&a);
        s=s+a;
        ++i;
    }
    av=s/n;
    printf(" the average is: %f ",av);
    getch();
}
```

Write C program to find the following

$$\sum_{i=1}^n i^2 = 1^2 + 2^2 + 3^2 + \dots + n^2$$

```
#include <stdio.h>
#include <conio.h>
main()
{
    int i = 1, n ,sum = 0;
    printf("enter number: n ");
    scanf("%d",&n);
    while ( i <= n )
    {
        sum += i * i ;
        i++;
    }
    printf("sum is %d:", sum);
    getch();
}
```

Write C program to print the following using while

```
* * * *  
* * * *  
* * * *  
* * * *
```

```
#include <stdio.h>  
#include <conio.h>  
main()  
{  
    int i=1;  
    while (i<=4)  
    {  
        int j=1;  
        while(j<=4)  
        {  
            printf("* ");  
            j++;  
        }  
        printf("\n");  
        i++;  
    }  
    getch();  
}
```

The do-while Loop

- Unlike **while** loops, which test the loop condition at the top of the loop, the **do-while** loop checks its condition at the bottom of the loop. This means that a **do-while** loop always executes at least once.
- The general form of the **do-while** loop is

```
do
{
    statement;
}
while(condition);
```

What is the output of the program

```
#include<stdio.h>
#include<conio.h>    /* Just in Windows and DOS OS compilers */

main()
{
    int i;
    i = 1;

    do
    {
        printf("\a%d ", i);
        i++;
    }while(i != 4);

    printf(" Go!\n");
}
```


write C program to find the average of student's degrees. assume the student has 8 degrees using do while

```
#include <stdio.h>
#include <conio.h>
main( )
{
    int i;
    float av, degree, sum;
    i = 0;
    sum=0;
    do
    {
        printf("enter degree:");
        scanf("%f",&degree);
        sum = sum + degree;
        i++;
    }
    while ( i < 8 );
    printf("sum is %f:\n",sum);
    av = sum / 8;
    printf("the average is :%f\n", av);
    getch();
}
```

Write C program to find the factorial of n:: $n! = n * n-1 * n-2 * n-3 * \dots * 1$.

```
#include <stdio.h>
#include <conio.h>
main()
{
    int n, f = 1;
    printf ("enter the number: n ");
    scanf("%d", &n);
    do
    {
        f = f * n;
        n --;
    }
    while(n>=1);
    printf ("factorial is: %d", f);
    getch();
}
```

write C program to find the average of the n students , assume every student has m degrees:

```
#include <stdio.h>
#include <conio.h>
main( )
{
    int n=0,m=0, i;
    float av, degree;
    printf("enter numbers of students and number of degrees n ,m:");
    scanf("%d%d",&n,&m);
    i = 1;
    do
    {
        float s=0;
        int j=1;
        do
        {
            printf("enter degree:");
            scanf("%f",&degree);
            s= s+ degree;
            j++;
        }
        while ( j <= m );
        printf("sum is %f:\n",s);
        av = s / m;
        printf("the average is :%f\n", av);
        ++i;
    }
    while ( i <= n );
    getch();
}
```