

DECISION MAKING

Q.what do you mean by control instruction? what are the types of control instructions? Marks:3 Exam-ACCE-2013

Q.what do you mean by control statement? Exam:ICB-2015,APPE
Ans:

C language possesses such decision making capabilities by supporting the following statements :-

- (a) if statement
- (b) Switch statement
- (c) Conditional statement
- (d) Go to statement
- (e)

These statements are popularly known as decision making statements . since these statements control control statements.

Q.What do you mean by If statement?

Ans:

The if statement is a powerful decision making statement and is used to control the flow of execution of statements . It is basically a two-way decision statement and is used in conjunction with an expression.
Or

The if, if...else used to make one-time decisions in C Programming, that is, to execute some code/s and ignore some code/s depending upon the test expression.

The general form of a simple if statement is,

```
If(test expression)
{ statement-block executed if test expression is
true;
}
Statement_x;
```

The statement block may be a single statement or a group of statements. If the test expression is true, The statement-block will be executed; otherwise the statement-block will be skipped and the execution will jump to the Statement_x. Remember , when the condition is true both the statement-block and the Statement_x are executed in sequence.

For an example, consider the following program-

```
Int main()
{
    Int a,b;
    Printf("Enter the value of a: ");
    Scanf("%d",&a);
    Printf("Enter the value of b: ");
    Scanf("%d",&b);
    If(a>b)printf("The bif number is =%d",a);
}
```

Q. What do you mean by else-If statement? Exam:ICB-

Ans:

The if-else statement is extension of the simple if statement , the general form is-

```
If(test expression)
{ True-block statement(s)
}
Else {
    False-block statement (s)
```

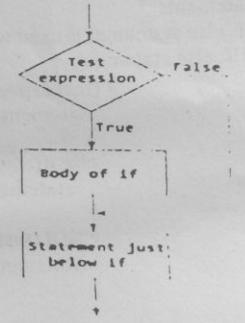


FIGURE: Flowchart of If statement

```

        }
    Statement_x
}

```

If the test expression is true then the true-block statement(s) immediately following the if statements are executed. In either case, either true-block or false-block will be executed not both.

Let us consider an example is -

```

#include <stdio.h>
int main()
{
    printf("Enter mark:");
    scanf("%d",&mark);
    if(mark>=33)
        printf("Pass");
    else
        printf("Fail");
    return 0
}

```

Q. Describe Nested If...else statement ?

Ans:

If...elseif...else Statement:

The nested if...else statement is used when program requires more than one test expression.
Syntax of nested if...else statement.

```

if (test expression1){
    statement to be executed if test expression1 is true;
}
else if(test expression2) {
    statement to be executed if test expression1 is false and 2 is true;
}
else if (test expression 3) {
    statement to be executed if test expression1 and 2 are false and 3 is true;
}

else {
    statements to be executed if all test expressions are false;
}

```

Q.How nested If...else works?

Ans:

The nested if...else statement has more than one test expression. If the first test expression is true, it executes the code inside the braces{ } just below it. But if the first test expression is false, it checks the second test expression. If the second test expression is true, it executes the statement inside the braces{ } just below it. This process continues. If all the test expression are false, code/s inside else is executed and the control of program jumps below the nested if...else

The ANSI standard specifies that 15 levels of nesting may be continued.

Write a C program to relate two Integers entered by user using = or > or < sign.

```

#include <stdio.h>
int main(){
    int numb1, numb2;
    printf("Enter two integers to check\n");
    scanf("%d %d",&numb1,&numb2);
    if(numb1==numb2) //checking whether two Integers are equal.
        printf("Result: %d = %d",numb1,numb2);
    else

```

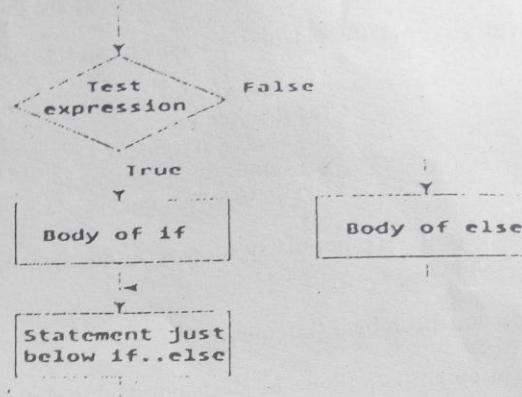


Figure: Flowchart of if...else Statement

```

        if(numb1>numb2) //checking whether numb1 is greater than
numb2;
        printf("Result: %d > %d",numb1,numb2);
    else
        printf("Result: %d > %d",numb2,numb1);
    return 0;
}

Output 1
Enter two integers to check.
5
3
Result: 5 > 3
Output 2
Enter two integers to check.
-4
-4
Result: -4 = -4

```

Q. Write a C program to find largest number from 3 numbers using if...else statement?
Ans:

```

#include <stdio.h>
int main(){
    int a, b, c;
    printf("Enter three numbers: ");

    scanf("%d %d %d", &a, &b, &c);
    if (a>=b)
    {
        if(a>=c)
            printf("Largest number = %d ",a);
        else
            printf("Largest number = %d ",c);
    }
    else
    {
        if(b>=c)
            printf("Largest number = %d ",b);
        else
            printf("Largest number = %d ",c);
    }
    return 0;
}

```

Q. Write a C Program that reads marks of a student and computes and displays grade using if...else .
Ans:

```

#include<stdio.h>
int main ()
{
    int a;
    printf("Please Enter the course result marks: ");
    scanf("%d", &a);
    if (a>=80)
    {
        printf("The GPA is: A+");
    }
}

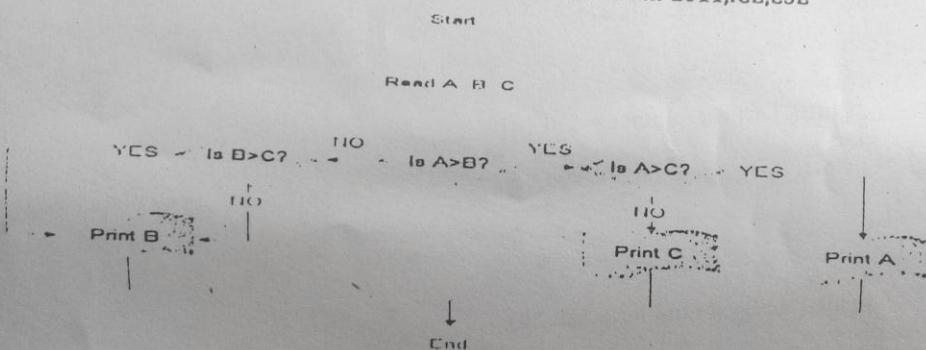
```

```

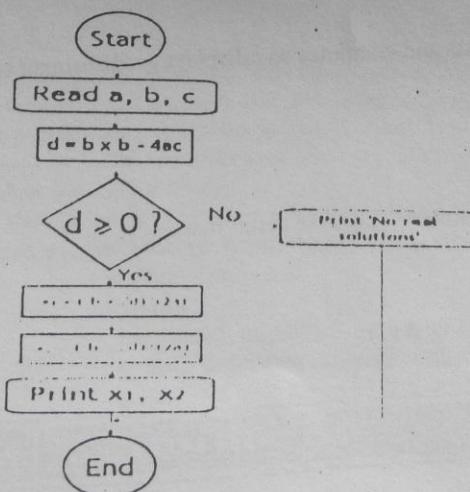
else if (a>=75 && a<=79){
    printf("The GPA is: A");
}
else if (a>=70 && a<=74){
    printf("The GPA is: A-");
}
else if (a>=65 && a<=69){
    printf("The GPA is: B+");
}
else if (a>=60 && a<=64){
    printf("The GPA is: B");
}
else if (a>=55 && a<=59){
    printf("The GPA is: C");
}
else if (a>=50 && a<=54){
    printf("The GPA is: C-");
}
else if (a>=40 && a<=49){
    printf("The GPA is: D");
}
else
{
    printf("Fail");
}
return 0;

```

Q. Draw a flow chart to pick the largest of three numbers? Exam-2011, ICE, CSE



Q. Draw a flow chart for computing the roots of a quadratic equation? Ans:



Write a C program to Find Roots of Quadratic Equation, Exam:ACCE-2013

```
#include <math.h> /* This is needed to use sqrt() function.*/
int main()
{
    float a, b, c, determinant, r1,r2, real, imag;
    printf("Enter coefficients a, b and c: ");
    scanf("%f%f%f",&a,&b,&c);
    determinant=b*b-4*a*c;
    if (determinant>0)
    {
        r1=(-b+sqrt(determinant))/(2*a);
        r2=(-b-sqrt(determinant))/(2*a);
        printf("Roots are: %.2f and %.2f",r1,r2);
    }
    else if (determinant==0)
    {
        r1 = r2 = -b/(2*a);
        printf("Roots are: %.2f and %.2f",r1,r2);
    }
    else
    {
        real=-b/(2*a);
        imag=sqrt(-determinant)/(2*a);
        printf("Roots are: %.2f+%.2fi and %.2f-%.2fi", real, imag, real, imag);
    }
    return 0;
}
```

Output 1

Q.What would be the output of the following programs? Exam:ACCE-2013

Ans:

```
#include<stdio.h>
int main()
{
    int x=4,y,z;
    y=--x;
    z=x--;
    printf("\n%d%d%d",x,y,z);
    return 0;
}
```

Ans: 2,3,3

```
#include<stdio.h>
int main()
{
    int j;
    while(j<=10)
    {
        printf("\n%d",j);
        j=j+1;
    }
    return 0;
}
```

Ans: There is no value of j; because j is not initialized.

Q. Consider the following program Exam:ACCE-2013

Ans:

A GUIDED BOOK OF COMPUTER PROGRAMMING

```
#include<stdio.h>
int main()
{
    int i=10,n=0;
    while(i<1)
    {
        if(i&1==1)
        {
            i+=i<<2;
            i=i|1;
        }
        else i>>=1;
        n++;
    }
    printf("%d",n);
    return 0;
}
Ans: 0
```

Q.What do you mean by "Nesting of if else statement s" Exam;ICE-2013
Ans:

when a series of decisions are involved, we may have use more than one if-else statement in nested form as shown below:

```
If(test-condition_1)
{
    If(test-condition_2)
    {
        Statement_1
    }
    Else
    {
        Statement_2
    }
}
Else
{
    Statement_3
}
```

If the condition_1 is false the statement_3 will be execute. Otherwise it continues to perform the second test . if the condition_2 is true the statement_1 will be evaluated and then the control is transferred to the statement_x

Let a program to explain it which plays to contain same number or none.

```
#include<stdio.h>
Int main()
{
    Int a, b, c,max;
    Printf("a=");
    Scanf("%d", &a);
    Printf("b=");
    Scanf("%d",&b);
    Printf("c=");
    Scanf("%d",&c);
    If(a>b)
    {
        If(a>c)
```

```

        Max=a;
    Else
        Max=b;
    }
    Else
    {
        If(b>c)
            Max=b;
        Else
            Max=c;
    }
    Print("Maximum=%d",max);
}

```

Q. Compare different loop statements used in C programming with flowchart and example? Exam: APEE, ACCE-2013

Ans: for loop, while loop, do while loop, switch and goto statement briefly describe.

Q. Write the syntax of switch statement?

Ans:

The syntax of switch statement is as shown below :

```

switch(expression)
{
    case value_1:
        block-1;
        break;
    case value_2:
        block-2;
        break;
    .....
    .....
    default:
        default-block;
        break;
    }
    statement-x;
}

```

Q. Write the advantage and Disadvantage of If statement?

Ans:

Advantage of IF:

It can be used whenever you want a condition to get satisfied and then execute a particular set of codes.

Disadvantage of IF:

In IF we wont mention what to do if the condition does not get satisfied, to overcome this we use IF ELSE.

Q. write the advantage and Disadvantage of Loop?

Ans:

Advantage of loop:

It can be used whenever a single or a set of statements have to be repeated again and again. It reduces the typing effort.

Disadvantage of loops:

If the condition is not properly specified then there is a chance that the program may go into infinite loop.

Q. Explain the switch case statement with example? Marks:2.5 Exam-MSE,ACCE-2014,13

Q. What is switch statement?

Q. Define Switch Statement?

}

Output:11,12,13,14...20

Q. How for loop works in C programming?

Ans:

The initialization statement is executed only once at the beginning of the for loop. Then the test expression is checked by the program. If the test expression is false, for loop is terminated. But if test expression is true then the code inside body of for loop is executed and then update expression is updated. This process repeats until test expression is false.

This flowchart describes the working of for loop in C programming

How for loop work example below:

Q. Write a program to find the sum of first n natural numbers where n is entered by user. Note: 1,2,3... are called natural numbers.

Ans:

```
#include <stdio.h>
int main()
{
    int n, count, sum=0;
    printf("Enter the value of n.\n");
    scanf("%d",&n);
    for(count=1;count<=n;++count) //for loop terminates if count>n
    {
        sum+=count; /* this statement is equivalent to sum=sum+count */
    }
    printf("Sum=%d",sum);
    return 0;
}
```

Output

Enter the value of n.

19

Sum=190

Explain: In this program, the user is asked to enter the value of n . Suppose you entered 19 then, count is initialized to 1 at first. Then, the test expression in the for loop, i.e., ($count \leq n$) becomes true. So, the code in the body of for loop is executed which makes sum to 1. Then, the expression $++count$ is executed and again the test expression is checked, which becomes true. Again, the body of for loop is executed which makes sum to 3 and this process continues. When count is 20, the test condition becomes false and the for loop is terminated.

Note: Initial, test and update expressions are separated by semicolon(:).

Q. How can 'for' be infinite? Exam-2011

```
For(l=0;l=1;l--)
{
```

```
//statement  
}
```

Q.Explain the syntax for for loop.
 Ans:

```
for(exp-1;exp-2;exp-3) {  
    //set of statements  
}
```

When control reaches for exp-1 is executed first. Then following exp-2, and if exp-2 evaluates to non-zero 'set of statements' and exp-3 is executed, follows exp-2.

Q Write a program for factorial of a number?
 Ans:

```
#include<stdio.h>  
int main(){  
    int i,f=1,num;  
    printf("Enter a number: ");  
    scanf("%d",&num);  
    for(i=1;i<=num;i++)  
        f=f*i;  
    printf("Factorial of %d is: %d",num,f);  
    return 0;  
}
```

Sample output:

```
Enter a number: 5  
Factorial of 5 is: 120
```

Q.Write a c program to find out the sum of series $1 + 2 + \dots + n$.
 Ans:

Sum of $1 + 2 + \dots + n$ series in c programming language

```
#include<stdio.h>  
int main(){  
    int n,i;  
    int sum=0;  
    printf("Enter the n i.e. max values of series: ");  
    scanf("%d",&n);  
    sum = (n * (n + 1)) / 2;  
    printf("Sum of the series: ");  
    for(i=1;i<=n;i++){  
        sum=sum+i;  
    }  
    printf("%d",sum);  
    return 0;  
}
```

Sample output:

Enter the n: 5

Sum of the series: $1 + 2 + 3 + 4 + 5 = 15$

Q.Code for PROGRAM TO PRINT THE SUM OF SERIES $1 + 1/2 + 1/3 + 1/4 + \dots + 1/N$ in C Programming
 Ans;

```
#include<stdio.h>  
#include<conio.h>  
void main() {  
    float n,sum=0,i;  
    printf("\n Please Give The Value of N: ");  
    scanf("%d",&n);  
    for(i=1;i<=n;i++)  
    {  
        sum = sum + (1/i);  
        if(i==1)
```

Ans:

Loop: A loop statement allows us to execute a statement or group of statements multiple times and following is the general form of a loop statement in most of the programming languages.
Or

February 1, 2016

Loops cause program to execute the certain block of code repeatedly until test condition is false. Loops are used in performing repetitive task in programming. Consider these scenarios:

- You want to execute some code 100 times.
- You want to execute some code certain number of times depending upon input from user.

These types of task can be solved in programming using loops

C programming language provides the following types of loops to handle looping requirements.

- 1.goto statements
- 2.forloop statements
- 3.while loop
3. do while loop
- 4.switch case statement

Q.What are the valid places for the keyword break to appear.
Ans:

Break can appear only with in the looping control and switch statement. The purpose of the break is to bring the control out from the said blocks.

Q.What is difference between including the header file with-in angular braces <> and double quotes
Ans:

If a header file is included with in <> then the compiler searches for the particular header file only with in the built in include path. If a header file is included with in "", then the compiler searches for the particular header file first in the current working directory, if not found then in the built in include path.

Q.What is an infinite loop? CSE-ICE-2013ACCE-2011
Ans:

A loop executing repeatedly as the loop-expression always evaluates to true such as

while(0 == 0) {
}

or

while(1) {
}

Q.Can variables belonging to different scope have same name? If so show an example.
Ans:

Variables belonging to different scope can have same name as in the following code snippet.

int var;

void f() {
 int var;
}

main() {
 int var;
}

Q.Can a pointer access the array?

Ans:

Pointer by holding array's base address can access the array.

Q.What is recursion?

Ans:

Function calling itself is called as recursion.

Q.What is a constant?

Ans:

A value which cannot be modified is called so. Such variables are qualified with the keyword const.

Q.What is the meaning of base address of the array?

Ans:

The starting address of the array is called as the base address of the array.

Q.What do you know about for loop explain with example?

Ans:

for loop:

A for loop is a repetition control structure that allows you to efficiently write a loop that needs to execute a specific number of times. Syntax

The syntax of a for loop in C programming language is:

```
for( init; condition; increment )
{ statement(s);
}
```

Here is the flow of control in a for loop:

Step 1: first initialization happens and the counter variable gets initialized, here variable is i, which has been assigned by value 1.

Step 2: then condition checks happen, where variable has been tested for a given condition, if the condition results in true then C statements enclosed in loop body gets executed by compiler, otherwise control skips the loop and continue with the next statement following loop.

Step 3: After successful execution of loop's body, the counter variable is incremented or decremented, depending on the operation (++ or --).

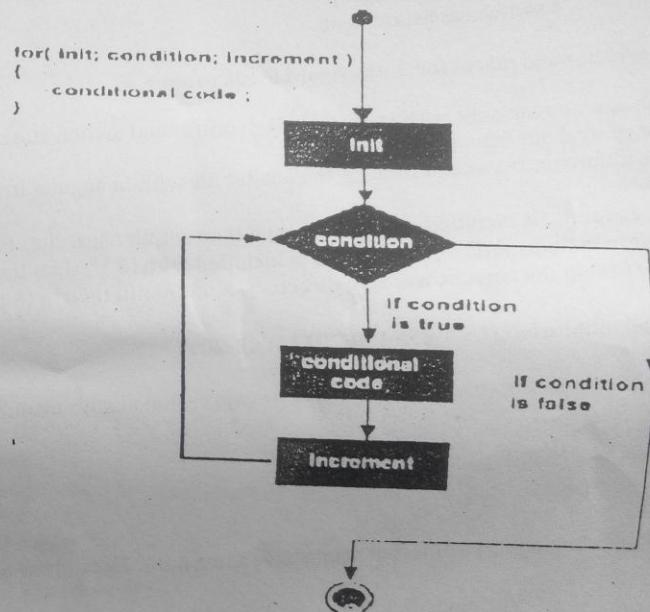
4. The condition is now evaluated again.

Example

```
int i;
for (i=1; i<=3; i++)
{
    printf("hello, World");
}
...Output:
hello, World
hello, World
hello, World
```

or

```
#include <stdio.h>
int main ()
{
for( int a = 10; a < 20; a = a + 1 )
{
printf("value of a: %d\n", a);
}
return 0;
```



```

        printf("\n i +");
    else if(i==n)
        printf(" (1/%d) ",i);
    else
        printf(" (1/%d) + ",i);
    }
    printf("\n\n THE SUM OF THIS SERIES IS %f",sum);
    return 0;
}

```

Q. Write a program to generate the Fibonacci series ?
Ans:

```

#include <stdio.h>
int main(){
    int k,r;
    int i=0,j=1,f;

    printf("ENTER THE NUMBER OF TERMS : ");
    scanf("%d",&r);
    printf("FIBONACCI SERIES: ");
    printf("%d %d",i,j);
    for(k=2;k<r;k++){
        f=i+j;
        i=j;
        j=f;
        printf(" %d",j);
    }
    return 0;
}

```

Sample output:

Enter the number range: 15
FIBONACCI SERIES: 0 1 1 2 3 5 8 13 21 34 55 89 144 233 377.

Q.Find the following program.

Ans:

```

#include<stdio.h>
int main()
{
    int i=15,j=10;
    printf("hello");
    if(i>15)j++;
    else{
        if(i<15){j--;}
        else{
            if(j<=10){
                i+=++j;
                j=i++;
            }
            else
            {
                j++;
            }
        }
    }
    if(i>15)
    {
        i=-j;
    }
    else{i+=j;}
}

```

```

    }
    printf("i=%d\nj=%d\n",++i,++j);
    return 0;
}

```

Ans: i=3; j=25;

Q. Describe while statement?

Ans:

while loop:

A while loop statement in C programming language repeatedly executes a target statement as long as a given condition is true.

Syntax

The syntax of a while loop in C programming language is:

```

while(condition)
{
    statement(s);
}

```

Here, statement(s) may be 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 immediately following the loop.

Example

Q. Write a program that print 10 to 20 increment 1, using while loop.

Ans:

```

#include <stdio.h>
int main()
{
    /* local variable definition */
    int a = 10;
    /* while loop execution */
    while( a < 20 )
    {
        printf("value of a: %d\n", a);
        a++;
    }
    return 0;

```

Output: 11, 12, 13, 14...20

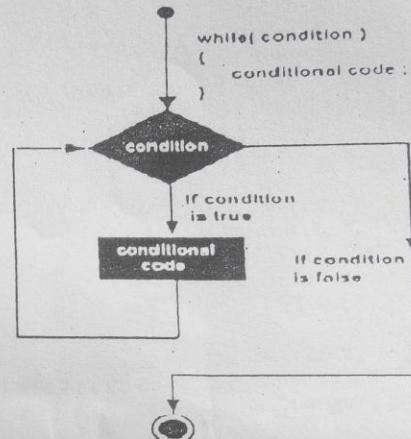
Q write a program for factorial of a number?

Ans:

```

#include <stdio.h>
int main(){
    int i=1,f=1,num;
    printf("Enter a number: ");
    scanf("%d",&num);
    while(i<=num){
        f=f*i;
        i++;
    }
    printf("Factorial of %d is: %d",num,f);
    return 0;
}

```



Q. Describe do while statement?

Ans:

do_while

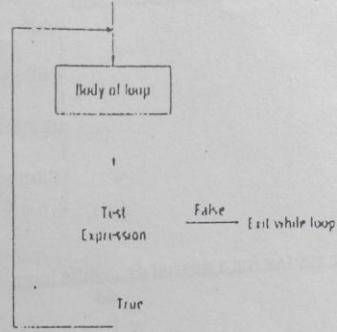
Unlike for and while loops, which test the loop condition at the top of the loop, the do_while loop in C programming language checks its condition at the bottom of the loop. A do_while loop is similar to a while loop, except that a do_while loop is guaranteed to execute at least one time.

Syntax

The syntax of a do_while loop in C programming language is:

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

Flow Diagram



Example:

```
main()
{
    int i=0
    do
    {
        printf("while vs do-while\n");
    }while(i==1);
    printf("Out of loop");
}
Output:
while vs do-while
Out of loop
```

Q. Write a program that print 10 to 20 increment 1, using do while loop

Ans:

```
#include <stdio.h>
int main ()
{
    //local variable definition
    int a = 10;           // do loop execution
    do
    {
        printf("value of a: %d\n", a);
        a = a + 1;
    }while(a < 20);
    return 0;
}
Output:10,11,12,13...20
```

Q. Discuss about the nested loops (nested for loop, nested while loop and nested do-while loop):

Ans:

C programming language allows to use one loop inside another loop. Following section shows few examples to illustrate the concept.

Syntax

The syntax for a nested for loop statement in C is as follows:

```
for ( init; condition; increment )
{
    for ( init; condition; increment )
```

```

    {
    statement(s);
}
statement(s);
}

```

The syntax for a nested while loop:

```

while(condition)
{
    while(condition)
    {
        statement(s);
    }
    statement(s);
}

```

The syntax for a nested do-while loop:

```

do
{
    statement(s);
do
{
    statement(s);
}while( condition );
}while( condition ); TUTORIA

```

Q.Discuss the Break and Continue Statement with example. Exam:ACCE-2013,11CSE,ICE.
Ans;

break	continue
A break can appear in both switch and 3 loop (for, while, do) statements.	A continue can appear only in 3 loop (for, while, do) statements.
A break causes the switch or loop statements to terminate the moment it is executed. Loop or switch ends abruptly when break is encountered.	A continue doesn't terminate the loop, it causes the loop to go to the next iteration. All iterations of the loop are executed even if continue is encountered. The continue statement is used to skip statements in the loop that appear after the continue.
When a break statement is encountered, it terminates the block and gets the control out of the switch or loop.	When a continue statement is encountered, it gets the control to the next iteration of the loop.
A break causes the innermost enclosing loop or switch to be exited immediately.	A continue inside a loop nested within a switch causes the next loop iteration

Example break statement using break using while loop and for loop:

```
int counter=10;
while (counter >=0)
{
    if (counter==7)
    {
        counter--;
        break;
    }
    printf("%d ", counter);
    counter--;
}
Output: 10 9 8 Out of while-loop

for (int j=0; j<=8; j++)
{
    if (j==4)
    {
        Break;
    }

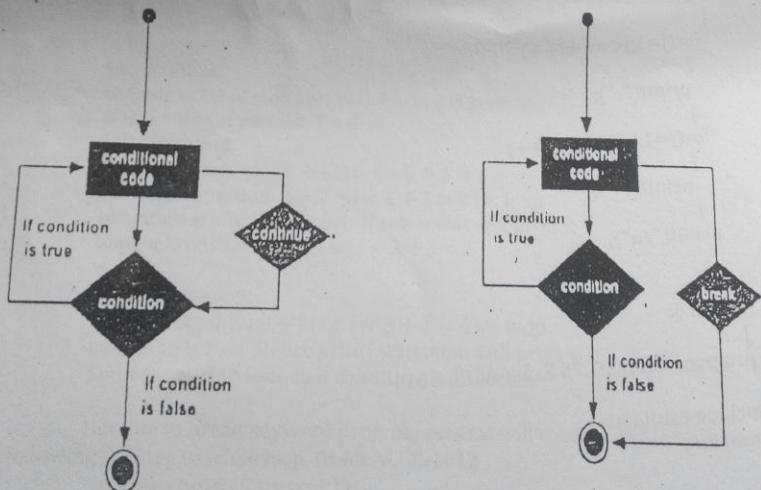
    printf("%d ", j);
}
Output:
0 1 2 3 Out of for loop
```

Example continue statement using break using while loop and for loop:

```
int counter=10;
while (counter >=0)
{
    if (counter==7)
    {
        counter--;
        continue;
    }
    printf("%d ", counter);
    counter--;
}
Output: 10 9 8 6 5 4 3 2 1 0

for (int j=0; j<=8; j++)
{
    if (j==4)
    {
        continue;
    }

    printf("%d ", j);
}
Output:
0 1 2 3 5 6 7 8
```



Q.What is Go to statement?

Ans:

A goto statement in C programming language provides an unconditional jump from the goto to a labeled statement in the same function.

Syntax

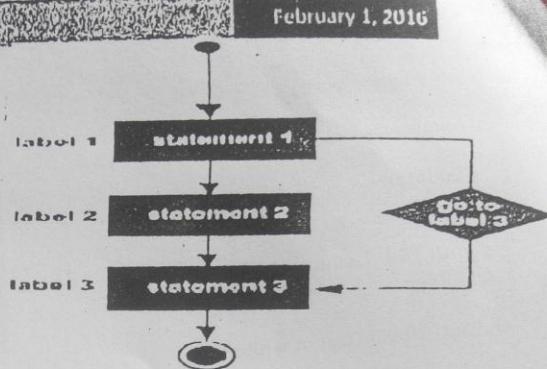
The syntax for a `goto` statement in C is as follows:

```
goto label; .
```

```
label: statement;
```

Here `label` can be any plain text except C keyword and it can be set anywhere in the C program above or below to `goto` statement.

Flow Diagram:



Q. Write a C program that will draw the following pyramid with height `n` on the console. [the example given below is a triangle with height `n=4`] Marks:4 Exam-ACCB-2014
Ans:

```

*
 *
 *
* * * *
* * * * *
#include<stdio.h>
int main()
{
    int i,j,n;
    printf("Enter How many line?");
    scanf("%d",&n);
    for(i=1;i<=n;i++)
    {
        for(j=1;j<=n-i;j++)
        {
            printf(" ");
        }
        for(j=1;j<=i*2-1;j++)
        {
            printf("*");
        }
        printf("\n");
    }
    return 0;
}
  
```

Q. Consider the following program Marks:2.75 Exam-ACCE-2014

```

#include<stdio.h>
int main()
{
    int i;
    for(i=5;i!=3;i++)
        printf("%d\n",i%8);
    return 0;
}
  
```

computer the output of above program.

Ans: 5,6,7,0,1,2;

```
Include<math.h>
FLOAT X;
X=2.5
Y=exp(x);
Print(x,y);
```

Ans: there are many error in this program, we know c programming is case sensitive
 1. First error Include I capital letter and also # is absence.
 2. FLOAT is capital letter.
 Y=initialize element is not constant;

Q. What will be output of following c code? Exam: ACCE 2012

```
#include<stdio.h>
int main(){
    static int i;
    for(++i;++i;++i) {
        printf("%d ",i);
        if(i==4) break;
    }
    return 0;
}
```

- (A) 4
- (B) 24
- (C) 25
- (D) Infinite loop
- (E) Compilation error

Answer: (b)**Explanation:**

Default value of static int variable in c is zero. So,
 initial value of variable i = 0

First Iteration:

For loop starts value: ++i i.e. i = 0 + 1 = 1

For loop condition: ++i i.e. i = 1 + 1 = 2 i.e. loop

condition is true. Hence printf statement will print 2

Loop incrementation: ++i i.e. i = 2 + 1 = 3

Second iteration:

For loop condition: ++i i.e. i = 3 + 1 = 4 i.e. loop
 condition is true. Hence printf statement will print 4.
 Since is equal to four so if condition is also true.

But due to break keyword program control will come out of the for loop.

Q. Change the following for loop to while loop Exam:ACCE-2012

```
for(m=1;m<10;m=m+1)
{
    printf("%d",m);
}
```

Ans:

m=1;

A GUIDED BOOK OF COMPUTER PROGRAMMING

```
while(m<10)
{
    printf("%d",m);
    m++;
}
```

Q. Difference between While and do-while loop? CSE-2011,ICE,APPE,ACCE-2012

Ans:

While loop	Do While loop
In While loop the condition is tested first and then the statements are executed if the condition turns out to be true.	A do while is used for a block of code that must be executed at least once. These situations tend to be relatively rare, thus the simple while is more commonly used.
In do while the statements are executed for the first time and then the conditions are tested, if the condition turns out to be true then the statements are executed again.	
while loop do not run in case the condition given is false.	A do while loop runs at least once even though the condition given is false
In a while loop the condition is first tested and if it returns true then it goes in the loop	In a do-while loop the condition is tested at the last.
While loop is entry control loop whereas do while is exit control loop.	
while loop : while (condition) { Statements; }	do while loop : Do { Statements; }while(condition);
while (choice != 0) { System.out.println("Inside the WHILE loop."); }	do { System.out.println("Inside the DO-WHILE loop."); } while (choice != 0);

Q.

Write a program that will take integer number as input and print negative number is entered?

Write the above using 'do-while', 'while', and 'for loop'. Exam:ACCE-2012,CSE

Ans:

For loop:

```
int main()
{
    int i,n,a[12];
    scanf("%d",&n)
    for(i=0;i<n;i++)
    {
        Sscanf("%d",&a[i])
    }
    for(i=0;i<n;i++)
    {
        if(a[i]>=0)
        {printf("%d",a[i]);}
```

```

        }
        return 0;
    }
    While loop:
    int main()
    {

        int i,n,a[12];
        scanf("%d",&n);
        i=0;
        while(i<n)
        {
            scanf("%d",&a[i])
        }
        i=0;
        while(i<n)
        {
            if(a[i]>=0)
            {printf("%d",a[i]);}
            i++;
        }
        return 0;
    }
    Do while loop:
    int main()
    {

        int i,n,a[12];
        scanf("%d",&n);
        do{
            i=0;
            scanf("%d",&a[i])
        } while(i<n)

        do{
            i=0;
            if(a[i]>=0)
            {printf("%d",a[i]);}
            i++;
        } while(i<n)

        return 0;
    }
}

```

GCD and LCM:

Q. Discuss about Greatest common devisor with example?

Ans:

It is simply the largest of the common factors.

Greatest Common Factor of 12 and 16

1. Find all the Factors of each number,
2. Circle the Common factors,

3. Choose the Greatest of those

Factors of 12: 1, 2, 3, 4, 6, 12

Factors of 16: 1, 2, 4, 8, 16

4. The Greatest Common Factor

Factor:

Factors are the numbers we multiply together to get another number:

$$2 \times 3 = 6$$

Factor Factor

A number can have many factors:

Factors of 12 are 1, 2, 3, 4, 6 and 12 ...

... because $2 \times 6 = 12$, or $4 \times 3 = 12$, or $1 \times 12 = 12$.

We can:

- find all factors of both numbers.
- then select the ones that are common to both, and
- then choose the greatest.

Example:

Two Numbers	Factors	Common Factors	Greatest Common Factor
9 and 12	9: 1,3,9 12: 1,2,3,4,6,12	1,3	3

Q. Discuss about Least common Multiple(LCM) with example?

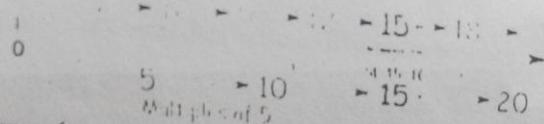
Ans:

It is simply the smallest of the common multiples.

Least Common Multiple of 3 and 5:

List the Multiples of each number,

The multiples of 3 are 3, 6, 9, 12, 15, 18, ... etc
 The multiples of 5 are 5, 10, 15, 20, 25, ... etc



Find the first Common (same) value:

The Least Common Multiple of 3 and 5 is 15

(15 is a common multiple of 3 and 5, and is the smallest, or least, common multiple)

Q. Write a program to find a greatest common divisor(GCD) and least common multiplier(LCM)?
 Ans:

```
#include <stdio.h>
int main() {
    int a, b, x, y, t, gcd, lcm;
    printf("Enter two integers\n");
    scanf("%d%d", &x, &y);
    a = x;
    b = y;
    while (b != 0) {
        t = b;
        b = a % b;
        a = t;
    }
    gcd = a;
    lcm = (x*y)/gcd;

    printf("Greatest common divisor of %d and %d = %d\n", x, y, gcd);
    printf("Least common multiple of %d and %d = %d\n", x, y, lcm);

    return 0;
}
```

Output of program:

```
E:\programmingimplified.com\c\hcf-lcm.exe
Enter two integers
9
24
Greatest common divisor of 9 and 24 = 3
Least common multiple of 9 and 24 = 72
```

Q. Write a program to find a least common multiplier(LCM)?

Ans:

```
#include<stdio.h>
int main()
{
    int n1,n2,temp1,temp2;
    printf("Enter two positive integers: ");
    scanf("%d %d",&n1,&n2);
    temp1=n1;
    temp2=n2;
    while(temp1!=temp2)
    {
        if(temp1>temp2)
            temp1-=temp2;
        else
            temp2-=temp1;
    }
    printf("LCM of two numbers %d and %d is %d",n1,n2,
    (n1*n2)/temp1);
    return 0;
}
```

The output of these two programs is same.

Output

```
Enter two positive numbers: 15
9
LCM of two numbers 15 and 9 is 45
```