Programming in C

Dr. Rupali P. Patil

Department of Electronics and Telecommunications

Topics for today

Module 3:
Control Structures in C

Control Structures in C

Conditional Branching Statements in C:

1. Simple if statement
2. if… else statement
3. Nested if…else statement
4. else…if ladder

Control Structures in C

Conditional Branching Statements in C:

1. Simple if statement

Syntax:

```
if(test-expression/condition)
{
True statement-block ;
 }
statement-x;
```

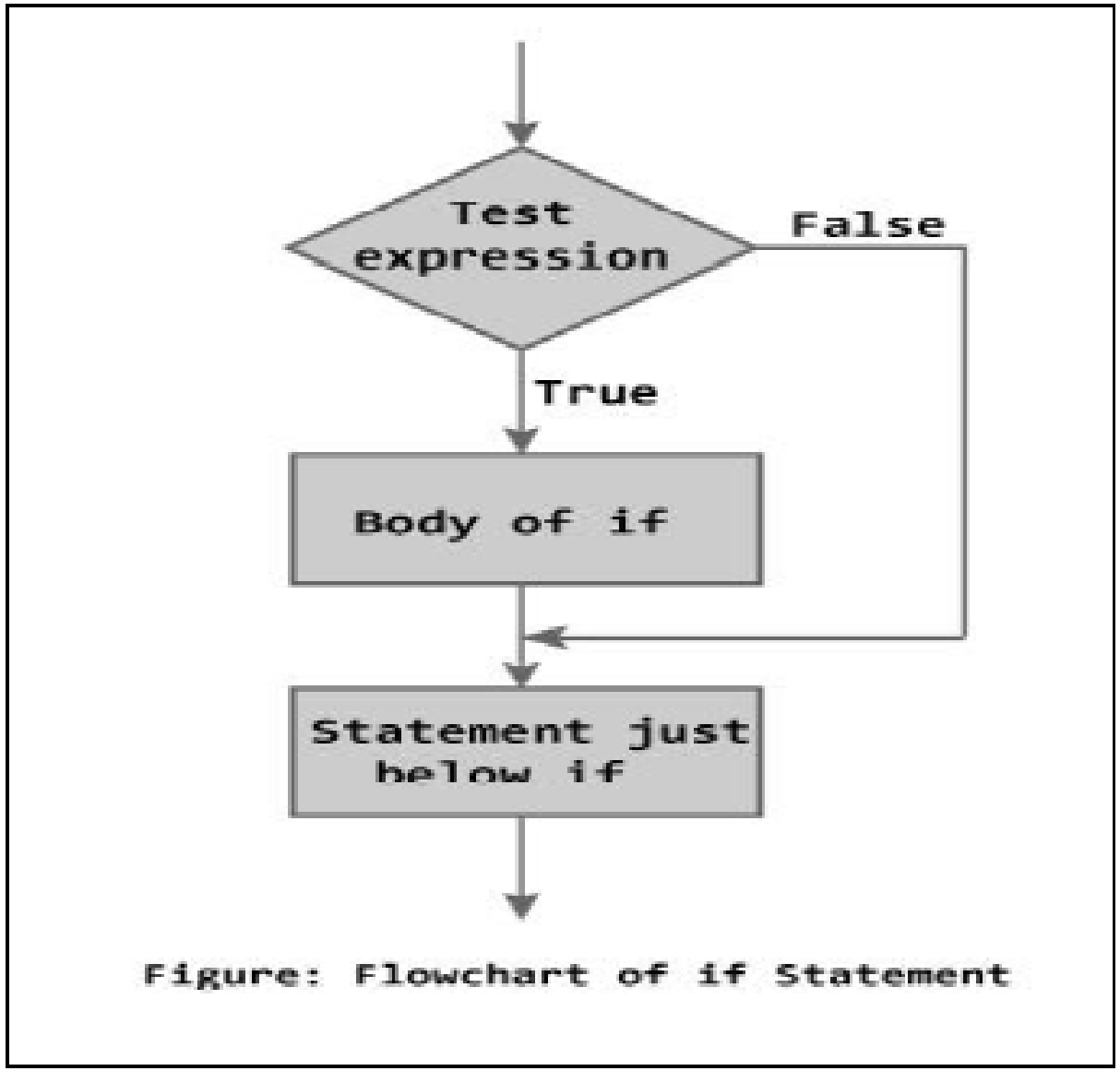


Figure: Flowchart of if Statement

Control Structures in C

Conditional Branching Statements in C:

1. Simple if statement

Example: C Program to check equivalence of two numbers using if statement

```
#include<conio.h>
#include<stdio.h>
void main()
{
    int m,n;
    clrscr();
    printf(" \n enter two numbers:");
    scanf(" %d %d", &m, &n);
    if(m-n= = 0)
      printf(" \n two numbers are equal");
    getch();
}
```

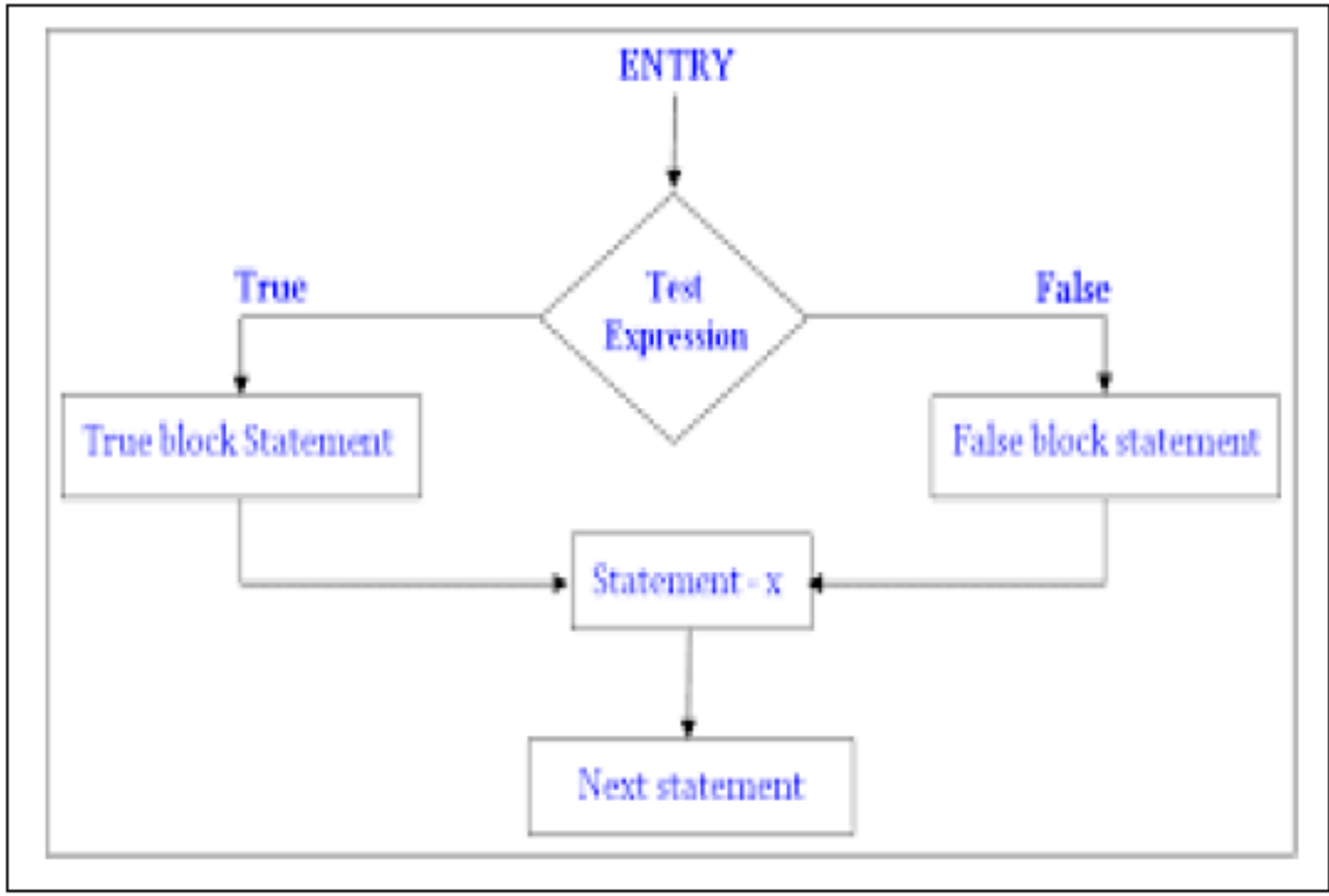
Control Structures in C

Conditional Branching Statements in C:

1. Simple if statement
2. if… else statement

Syntax:

```
if(test-expression/condition)
{
        true-block statements;
}
else
{
        false-block statements;
}
statement-x
```



Control Structures in C

Conditional Branching Statements in C:

1. Simple if statement
2. if… else statement

Example: C program to read any number as input through the keyboard and find out whether it is Odd Number or Even Number.

Control Structures in C

Conditional Branching Statements in C:

1. Simple if statement
2. if… else statement

Example: C program to read any number as input through the keyboard and find out whether it is Odd Number or Even Number.

```
#include<stdio.h>
#include<conio.h>
void main()
{
   int n;
   clrscr();
   printf("Enter the Number");
   scanf("%d",&n);
   if(n%2==0)
   {
      printf("This is Even Number");
   }
   else
   {
      printf("This is Odd Number");
   }
   getch();
}
```

Control Structures in C

Conditional Branching Statements in C:

1. Simple if statement
2. if… else statement

Example: C program to find biggest among two numbers using if else.

```
#include<stdio.h>
#include<conio.h>
void main()
{
   int a,b;
   clrscr();
   printf("Enter the two Number");
   scanf("%d%d",&a,&b);
   if(a>b)
   {
      printf("The number a=%d is bigger", a);
   }
   else
   {
      printf("The number b=%d is bigger",b);
   }
   getch();
}
```

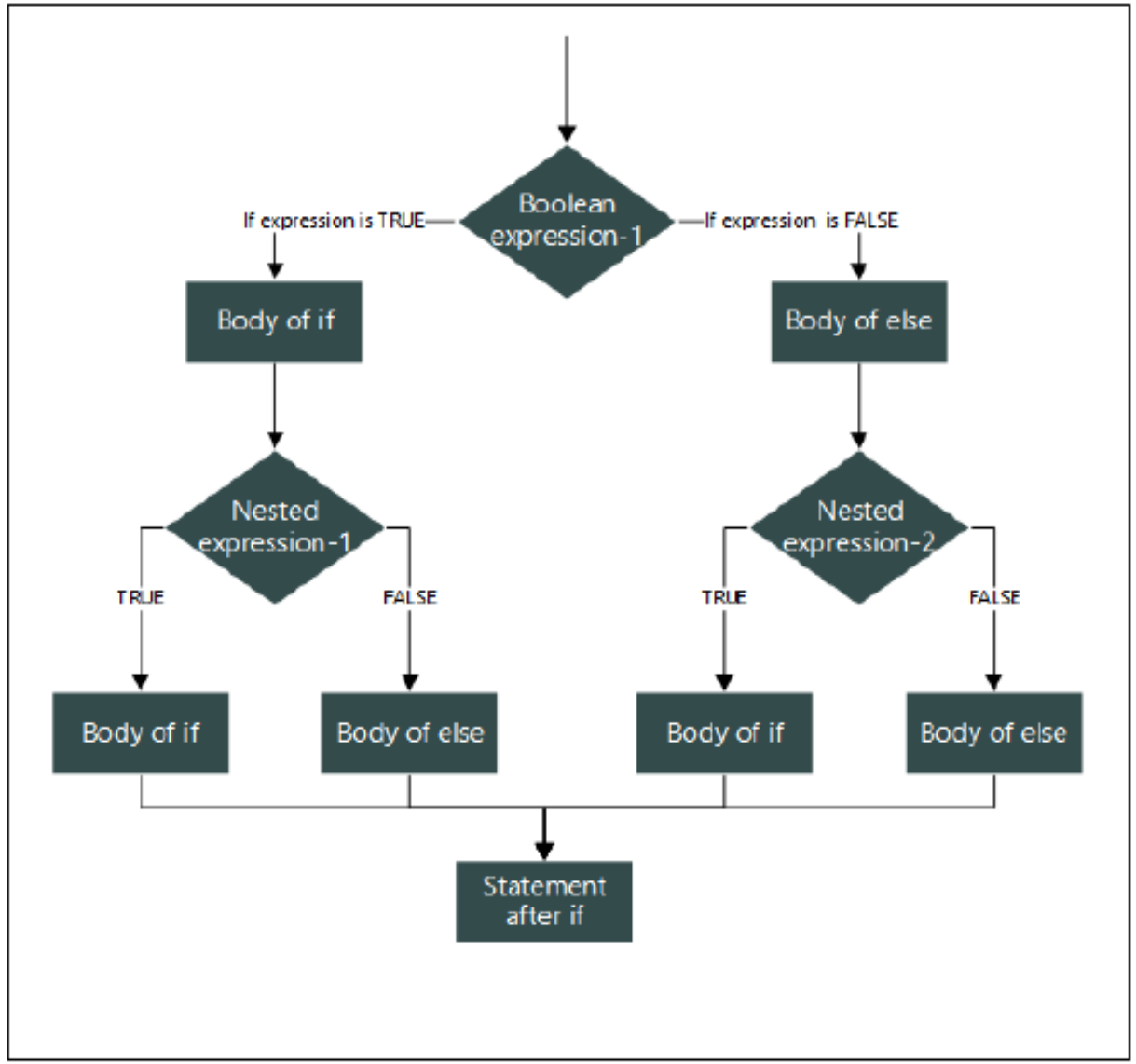
Control Structures in C

Conditional Branching Statements in C:

1. Simple if statement
2. if… else statement
3. Nested if…else statement

Syntax:

```
if(test-condition-1)
  {
    if(test-condition-2)
      {
        statement-1;
      }
    else
      {
        statement-2;
      }
  }
else
  {
    if(test-condition-3)
      {
        statement-3;
      }
    else
      {
        statement-4
      }
  }
statement-x
```



Control Structures in C

Conditional Branching Statements in C:

1. Simple if statement
2. if… else statement
3. Nested if…else statement

Example: C program to check whether person is eligible for work or not.

Control Structures in C

Conditional Branching Statements in C:

1. Simple if statement
2. if… else statement
3. Nested if…else statement

Example: C program to check whether person is eligible for work or not.

If age <18….Minor and not eligible
If age >=18 and age <=60 eligible
Else You are too old to work

Control Structures in C

Conditional Branching Statements in C:

1. Simple if statement
2. if… else statement
3. Nested if…else statement

Example: C program to check whether person is eligible for work or not.

```
#include <stdio.h>
 void main()
 {
        int age;
        printf("Please Enter Your Age Here:\n");
        scanf("%d",&age);
        if ( age < 18 )
        {
              printf("You are Minor.\n");
             printf("Not Eligible to Work");
        }
        else
        {
              if (age >= 18 && age <= 60 )
              {
                    printf("You are Eligible to Work \n");
                    printf("Please fill in your details and apply\n");
              }
             else
              {
                  printf("You are too old to work as per the Government rules\n");
                  printf("Please Collect your pension! \n");
              }
        }
 getch();
}
```

Control Structures in C

Conditional Branching Statements in C:

1. Simple if statement
2. if… else statement
3. Nested if…else statement

Example: C program if the ages of Ram, sham and Ajay are input through the keyboard, write a program to determine the youngest of the three

Control Structures in C

Conditional Branching Statements in C:

1. Simple if statement
2. if… else statement
3. Nested if…else statement

Example: C program if the ages of Ram, sham and Ajay are input through the keyboard, write a program to determine the youngest of the three

```
#include< stdio.h >
#include< conio.h >
void main()
{
    int ram,sham,ajay;
    clrscr();
    printf("Enter the Three Ages of Ram,Sham and Ajay\n");
    scanf("%d%d%d",&ram,&sham,&ajay);
    if(ram < sham)
     {
     if(ram < ajay)
       {
        printf("Ram is Youngest");
       }
     else
      {
       printf("Ajay is Youngest");
      }
    }
    else
      {
       if(sham < ajay)
         {
          printf("Sham is Youngest");
         }
       else
         {
          printf("Ajay is Youngest");
         }
      }
    getch();
}
```

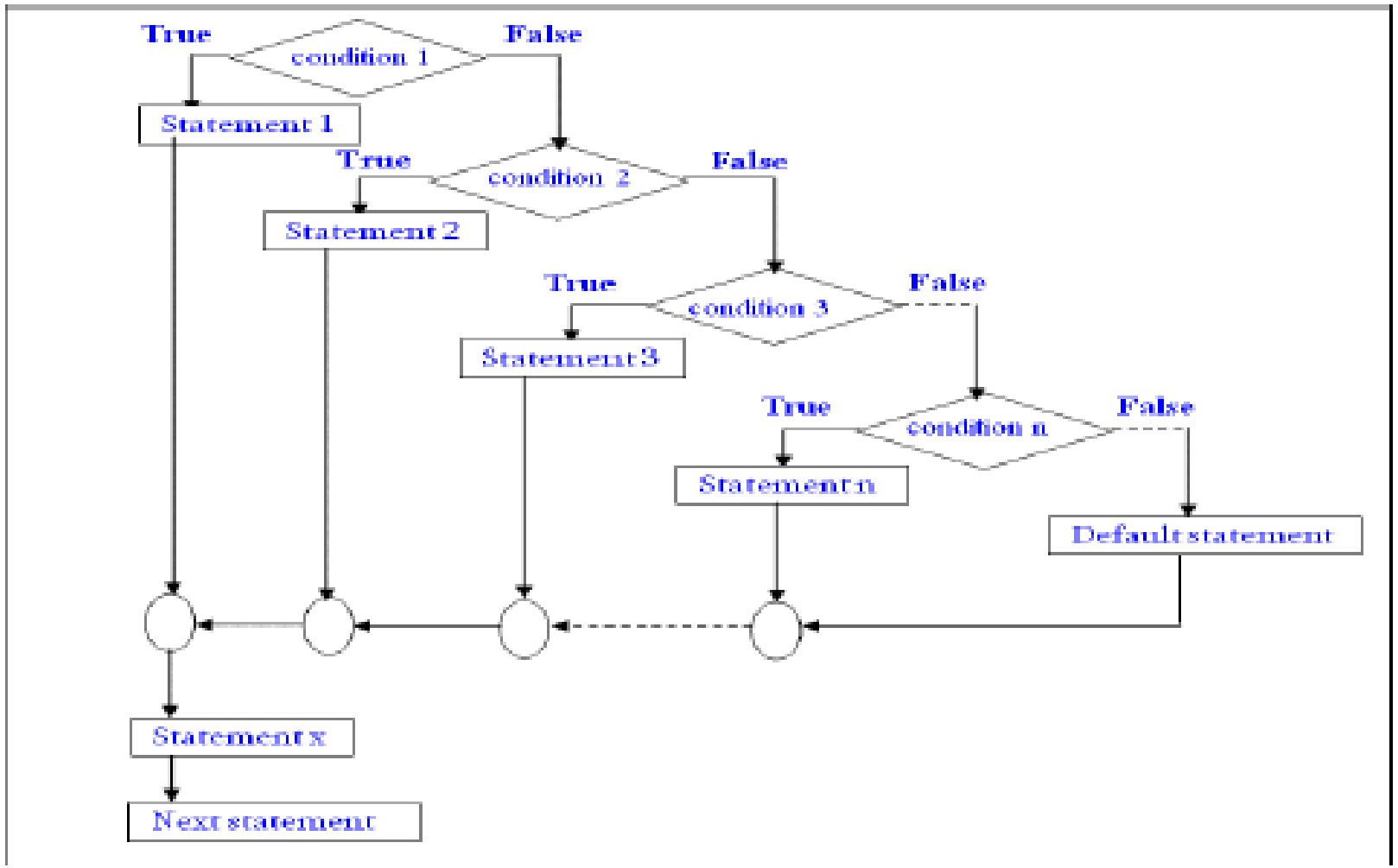
Control Structures in C

Conditional Branching Statements in C:

1. Simple if statement
2. if… else statement
3. Nested if…else statement
4. else…if ladder

Syntax:

```
if(condition-1)
   Statement-1;
else if(condition-2)
   Statement -2;
else if(condition-3)
    Statement -3;
else if(condition-n)
   Statement -n;
else
   Default Statement;
Statement -x;
```



Control Structures in C

Conditional Branching Statements in C:

1. Simple if statement
2. if… else statement
3. Nested if…else statement
4. else…if ladder

Example: C Program to print grade of a student using If Else Ladder Statement.

Control Structures in C

Conditional Branching Statements in C:

1. Simple if statement
2. if… else statement
3. Nested if…else statement
4. else…if ladder

Example: C Program to print grade of a student using If Else Ladder Statement.

If marks >90….A grade
If marks >=70 and marks <=90…B grade
If marks >=50 and marks <70…C grade

Else You are Fail

Control Structures in C

Conditional Branching Statements in C:

1. Simple if statement
2. if… else statement
3. Nested if…else statement
4. else…if ladder

Example: C Program to print grade of a student using If Else Ladder Statement.

```
#include<stdio.h>
#include<conio.h>
void main()
{
   int marks;
   printf("Enter your marks between 0-100\n");
   scanf("%d", &marks);
   if(marks >= 90)
    {
       printf("YOUR GRADE : A\n");
    }
  else if (marks >= 70 && marks < 90)
   {
     printf("YOUR GRADE : B\n");
   }
  else if (marks >= 50 && marks < 70)
   {
     printf("YOUR GRADE : C\n");
   }
  else
   {
     printf("YOUR GRADE : Failed\n");
   }
getch();
}
```

Control Structures in C

Conditional Branching Statements in C:

1. Simple if statement
2. if… else statement
3. Same Using Nested if…else statement
4. else…if ladder

```
#include <stdio.h>
void main ()
{
   int marks;
   printf ("Enter the amarks: ");
   scanf ("%d",&marks);
   if (marks>90)
   {

      printf ("The grade is : A");
   }

   if (marks>=70 && marks<=90)

   {
      printf ("The grade is : B");
   }

   if (marks>=50 && marks<70)

   {
      printf("The grade is : C");
   }

   else

   {
      printf("The person is eligible for re-exam ");
   }
}
```

Control Structures in C

Conditional Branching Statements in C:

1. Simple if statement
2. if… else statement
3. Nested if…else statement
4. else…if ladder

Example: Program to find maximum of three numbers using else-if ladder.

```
#include<conio.h>
#include<stdio.h>
viod main()
{
   int a,b,c,max;
   clrscr();
   printf("Enter values of a ,b,c");
   scanf("%d%d%d",&a,&b,&c);
   if(a>b && a>c)
      max=a;
   else if(b>c)
      max=b;
   else
     max=c;
  printf("Maximum is %d",max);
  getch();
}
```

Control Structures in C

Conditional Branching Statements in C:

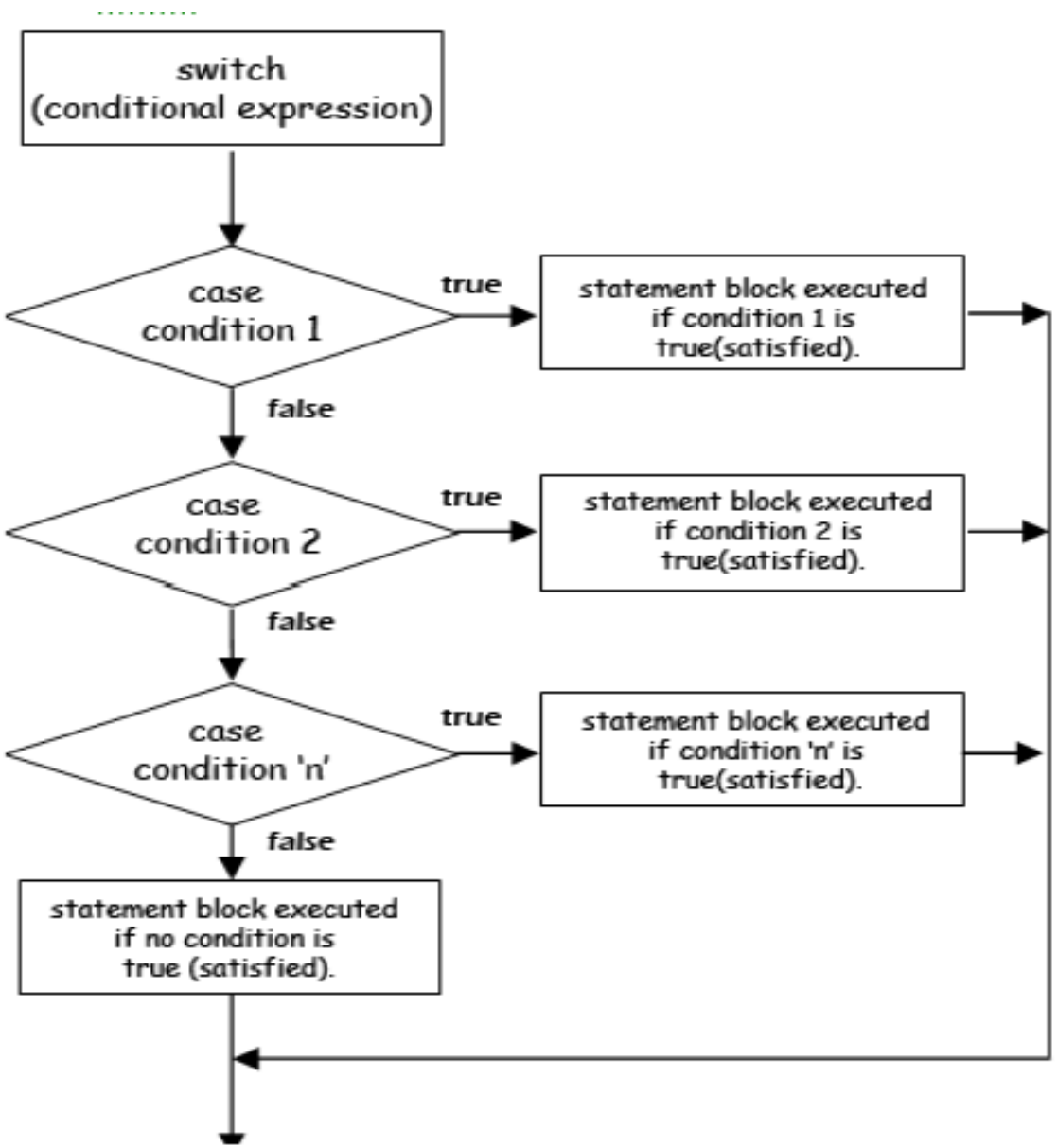
1. Simple if statement
2. if… else statement
3. Nested if…else statement
4. else…if ladder
5. Switch statement

```
switch(<exp>)
{
    case <exp-val-1>: statements block-1
                      break;
    case <exp-val-2>: statements block-2
                      break;
    case <exp-val-3>: statements block-3
                      break;
                  :
                  :
    case <exp-val-N>: statements block-N
                      break;
    default: default statements block
}
```

Control Structures in C

Conditional Branching Statements in C:

1. Simple if statement
2. if… else statement
3. Nested if…else statement
4. else…if ladder
5. Switch statement



Switch statement: Menu driven program for simple calculator

```
#include <stdio.h>
int main() {
   char operator;
   double first, second;
   printf("Enter an operator (+, -, *,): ");
   scanf("%c", &operator);
   printf("Enter two operands: ");
   scanf("%lf %lf", &first, &second);

   switch (operator) {
   case '+':
      printf("%.1lf + %.1lf = %.1lf", first, second, first + second);
      break;
   case '-':
      printf("%.1lf - %.1lf = %.1lf", first, second, first - second);
      break;
   default:
      printf("Error! operator is not correct");
   }
   return 0;
}
```

Switch statement:

Example: C program to Check Character is Vowel or not using Switch Case if it is vowel which vowel.

Switch statement:

Example: C program to Check Character is Vowel or not using Switch Case if it is vowel which vowel.

```
#include <stdio.h>
void main()
{
        char ch;
        printf(" Enter any character: ");
        scanf("%c", &ch);
        switch (ch)
        {
                case 'a':
                        printf(" %c is a vowel", ch);
                        break;
                case 'e':
                        printf(" %c is a vowel", ch);
                        break;
                case 'i':
                        printf(" %c is a vowel", ch);
                        break;
                case 'o':
                        printf(" %c is a vowel", ch);
                        break;
                case 'u':
                        printf(" %c is a vowel", ch);
                        break;
                default:printf(" %c is not a vowel", ch);
        }
        getch();
}
```

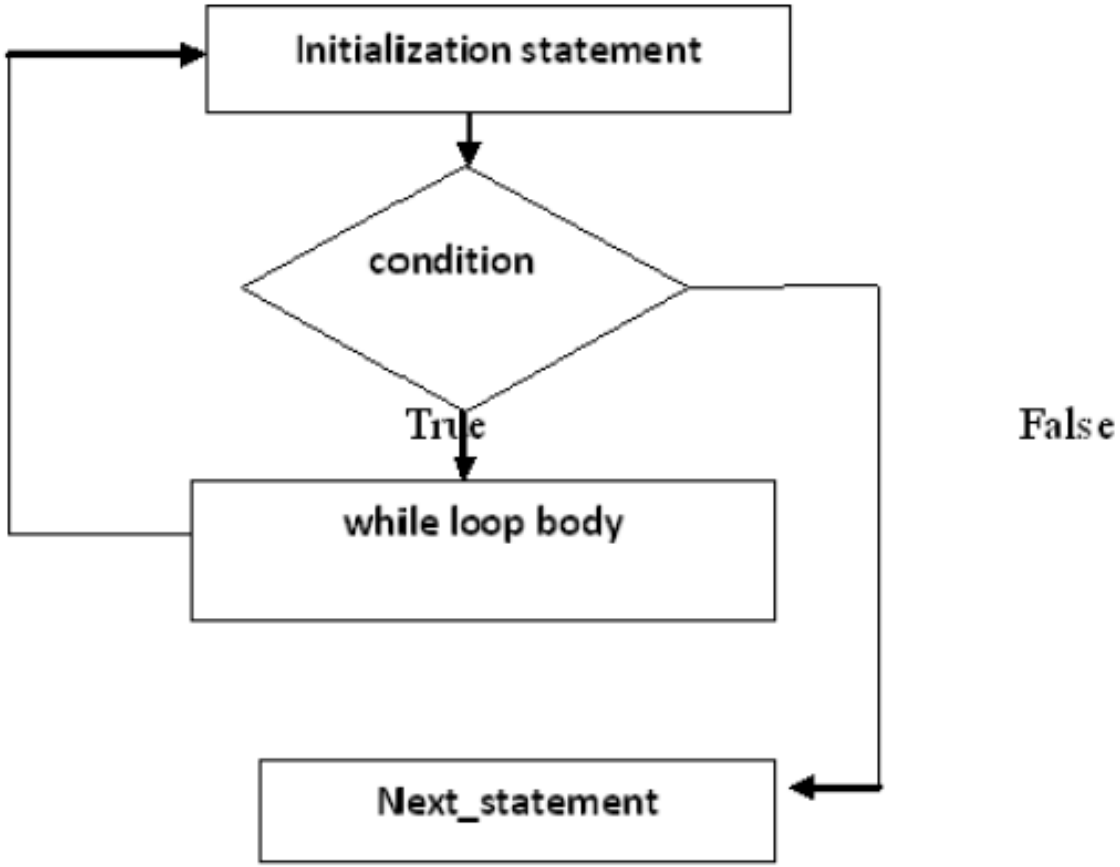
Control Structures in C

Conditional Branching Statements in C:

1. Simple if statement
2. if… else statement
3. Nested if…else statement
4. else…if ladder
5. Switch statement
6. While loop

Syntax:

```
While(condition)
{
Statements;
}
```



Control Structures in C

Conditional Branching Statements in C:

1. Simple if statement
2. if… else statement
3. Nested if…else statement
4. else…if ladder
5. Switch statement
6. While loop

Output= 1 2 3 4

WHILE: PRINT N NATURAL NUMBERS

```
#include<stdio.h>
#include<conio.h>
int main()
{
int i,n;
printf("enter the range\n");
 scanf("%d",&n);
 i=1;
 while(i<=n)
 {
 printf("%d",i);
 i=i+1;
 }
 getch();
 }
```

Control Structures in C

Conditional Branching Statements in C:

1. Simple if statement
2. if… else statement
3. Nested if…else statement
4. else…if ladder
5. Switch statement
6. While loop

Example: C program to Calculate sum of digits using while loop.

Control Structures in C

Conditional Branching Statements in C:

1. Simple if statement
2. if… else statement
3. Nested if…else statement
4. else…if ladder
5. Switch statement
6. While loop

s= 0	a = 123	s=s+(a%10)	a=a/10
0	123	3	12
3	12	3+2=5	1
5	1	5+1=6	0
stop			

Example: C program to Calculate sum of digits using while loop.

```
#include<stdio.h>
void main()
{
   int a, s;
   printf("Enter value of a: ");
   scanf("%d",&a);
   s = 0;
   while(a > 0)
   {
      s = s + (a%10);
      a = a / 10;
   }
   printf("Sum of digits: %d",s);
   getch();
}
```

Print the Fibonacci series up to given “range”

```
#include<stdio.h>
int main()
 {
    int a=0, b=1, range, c, sum=0;

    printf("Enter the range of Fibonacci series: ");
    scanf("%d",&range);

    printf("The fibonacci series is: \t");

    while( a <= range )
    {

        printf("%d\t",a);
        c = a + b;
        a = b;
        b = c;
    }
}
```

a = 0	b = 1	c
0	1	1
1	1	2
1	2	3
2	3	5
3	5	8
5	8	13
8	13	21
13	21	34

output for range = 9?

0 1 1 2 3 5 8

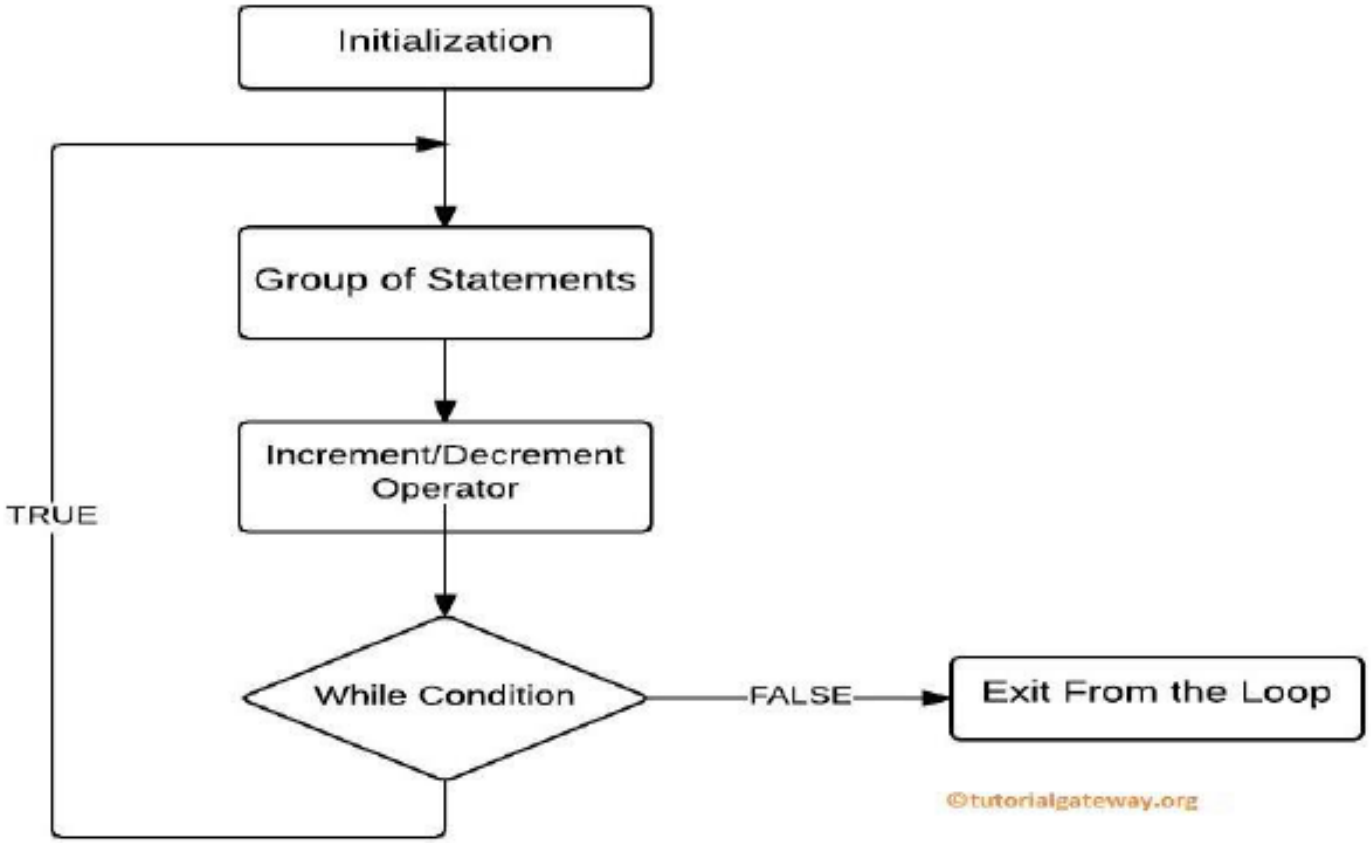
Control Structures in C

Conditional Branching Statements in C:

1. Simple if statement
2. if… else statement
3. Nested if…else statement
4. else…if ladder
5. Switch statement
6. While loop
7. Do- while loop

Syntax:

```
Initialization statement;
do
{    statement(s);
 } while(<condition>);
next statement;
```



Control Structures in C

Conditional Branching Statements in C:

1. Simple if statement
2. if… else statement
3. Nested if…else statement
4. else…if ladder
5. Switch statement
6. While loop
7. Do- while loop

Example: C program to calculate factorial value using do while.

Control Structures in C

Conditional Branching Statements in C:

1. Simple if statement
2. if… else statement
3. Nested if…else statement
4. else…if ladder
5. Switch statement
6. While loop
7. Do- while loop

Int gives factorial upto 9 only

Long int gives factorial upto 19 only

Long long int gives factorial upto 49 on

Example: C program to calculate factorial value using do while.

```
#include<stdio.h>
#include<conio.h>
void main()
{
  long int i,n,fact=1; /*variable declaration */
  clrscr();
  printf("Enter the value of n \n");
  scanf("%ld", &n);
  i=1;
  do
  {
    fact*=i;
    i++;
  }
  while(i<=n);
   printf("Factorial = %ld\n",fact);
  getch();
  }
```

```
#include<stdio.h>
int main()
{
   long long n,i,fact=1;
   printf ("Enter the number upto which you want factc
   scanf ("%lld",&n);
   i=n;
   do
   {
      fact=fact*i;
      i--;

   }
   while(i>=1);

   printf("The factorial is : %lld",fact);
   return 0;
}
```

Control Structures in C

Conditional Branching Statements in C:

1. Simple if statement
2. if… else statement
3. Nested if…else statement
4. else…if ladder
5. Switch statement
6. While loop
7. Do- while loop

Example: Write a C program to print the sum of all even and odd numbers up to n.

```
#include<stdio.h>
void main()
{
    int n,s1=0,s2=0,i;
    printf("Enter Number : ");
    scanf("%d",&n);
    i=1;
    do
    {
        if(i%2==0)
            s1=s1+i;
        else
            s2=s2+i;
     i++;
    }while(i<=n);
    printf("\nSum of Even Numbers :  %d\n",s1);
    printf("\nSum of Odd Numbers  :  %d\n",s2);
    getch();
}
```

Control Structures in C

Conditional Branching Statements in C:

1. Simple if statement
2. if… else statement
3. Nested if…else statement
4. else…if ladder
5. Switch statement
6. While loop
7. Do- while loop

Difference between while and do while loops:

While	Do While
The loop which continues until the statement holds true and repeats constantly.	The loop which holds true for specific instructions.
Only one statement for all the package to work	Requires separate statement for all the while conditions.

While (condition)
{
statement;
}.
	Do
	{
	statements;
	} while (condition);

Entry control.	Exit control.
Takes less time to execute but and the code is shorter.	Takes more time to execute and code becomes longer.

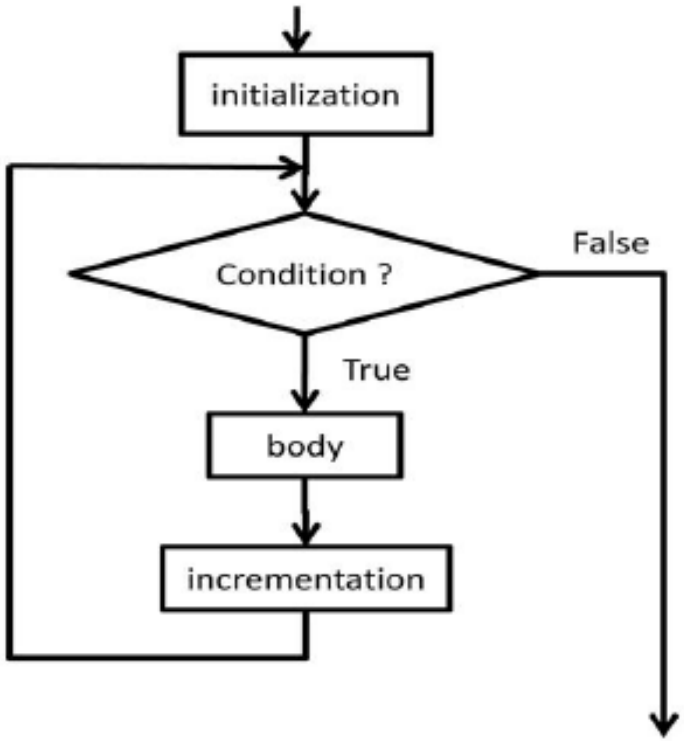
Control Structures in C

Conditional Branching Statements in C:

1. Simple if statement
2. if… else statement
3. Nested if…else statement
4. else…if ladder
5. Switch statement
6. While loop
7. Do- while loop
8. For statement

Flowchart of for loop:

for(initialization; condition; incrementation)
body;



Syntax of for loop:

```
for (initialization; condition test; increment or decrement)
{
    //Statements to be executed repeatedly
}
```

Control Structures in C

Conditional Branching Statements in C:

1. Simple if statement
2. if… else statement
3. Nested if…else statement
4. else…if ladder
5. Switch statement
6. While loop
7. Do- while loop
8. For statement

Print first “n” numbers using for loop

Control Structures in C

Conditional Branching Statements in C:

1. Simple if statement
2. if… else statement
3. Nested if…else statement
4. else…if ladder
5. Switch statement
6. While loop
7. Do- while loop
8. For statement

```
#include<stdio.h>
#include<conio.h>
void main()
{
int i,n;
printf("enter the value");
scanf("%d",&n);
for(i=1;i<=n;i++)
{
printf("%d\n",i);
}
getch();
}
```

Print the first N terms of Fibonacci series

```
#include<stdio.h>
int main()
 {
    int a=0, b=1, num, c, sum=0;

    printf("Enter the number of terms: ");
    scanf("%d",&num);

    printf("The fibonacci series is: \t");

    for(i=1; i<=num; i++) /* prints series for n number of terms */
    {
       printf("%d\t",a);
       sum = sum + a;
       c = a + b;
       a = b;
       b = c;
    }
}
```

output for num = 9?

0 1 1 2 3 5 8 13 21

Control Structures in C

Conditional Branching Statements in C:

1. Simple if statement
2. if… else statement
3. Nested if…else statement
4. else…if ladder
5. Switch statement
6. While loop
7. Do- while loop
8. For statement

Example: C Program to find Factorial of a Number.

```
#include <stdio.h>
#include<conio.h>
void  main()
{
   int n, i;
   unsigned long long factorial = 1;
   printf("Enter an integer: ");
   scanf("%d",&n);
   // show error if the user enters a negative integer
   if (n < 0)
      printf("Error! Factorial of a negative number doesn't exist.");
   else
   {
      for(i=1; i<=n; ++i)
      {
         factorial *= i;              // factorial = factorial*i;
      }
      printf("Factorial of %d = %llu", n, factorial);
   }
    getch();
}
```

Various forms of for loop in C:

1) Initialization part can be skipped from loop as shown below, the counter variable is declared before the loop.

```
int num=10;
for (;num<20;num++)
```

Note: Even though we can skip initialization part but semicolon (;) before condition is must, without which you will get compilation error.

2) Like initialization, you can also skip the increment part as we did below. In this case semicolon (;) is must after condition logic. In this case the increment or decrement part is done inside the loop.

```
for (num=10; num<20; )
{
//Statements
num++;
}
```

Various forms of for loop in C:

3) This is also possible. The counter variable is initialized before the loop and incremented inside the loop.

```
int num=10;
for (;num<20;)
{
//Statements
num++;
}
```

4) As mentioned above, the counter variable can be decremented as well. In the below example the variable gets decremented each time the loop runs until the condition num>10 returns false.

```
for(num=20; num>10; num--)
```

Nested For Loop in C:

```
#include <stdio.h>
void main()
{
  for (int i=0; i<2; i++)
  {
   for (int j=0; j<4; j++)
   {
     printf("%d, %d\n",i ,j);
   }
  }
  getch();
}
```

Output ?

```
0 0
0 1
0 2
0 3

1 0
1 1
1 2
1 3
```

Nested For Loop in C:

Step 5: write logic for sorting elements in the array

```
for(i=0;i<n-1;i++)
    {
       for(j=0;j<n-1-i;j++)
       {
          if(arr[j]>arr[j+1])
          {
             temp=arr[j];
             arr[j]=arr[j+1];
             arr[j+1]=temp;
          }
       }
    }
```