●●Prime number

```
#include<stdio.h>
int main(){
int n,i,m=0,flag=0;
printf("Enter the number to check prime:");
scanf("%d",&n);
m=n/2;
for(i=2;i<=m;i++)
{
if(n%i==0)
{
printf("Number is not prime");
flag=1;
break;
}
}
if(flag==0)
printf("Number is prime");
return 0;
 }
```

●●PALINDROME

```
#include<stdio.h>
```

```
int main()
{
int n,r,sum=0,temp;
printf("enter the number=");
scanf("%d",&n);
temp=n;
while(n>0)
{
r=n%10;
sum=(sum*10)+r;
n=n/10;
}
if(temp==sum)
printf("palindrome number ");
else
printf("not palindrome");
return 0;
}
```

●●Insertion sort

```
#include <stdio.h>
```

```
void insert(int a[], int n) /* function to sort an aay with insertion sort */

{

   int i, j, temp;

   for (i = 1; i < n; i++) {

      temp = a[i];

      j = i - 1;



      while(j>=0 && temp <= a[j])  /
* Move the elements greater than temp to one position ahead from their current position*/

      {

         a[j+1] = a[j];

         j = j-1;

      }

      a[j+1] = temp;

   }

}



void printArr(int a[], int n) /* function to print the array */

{

   int i;

   for (i = 0; i < n; i++)

      printf("%d ", a[i]);

}
```

```
int main()
{
    int a[] = { 12, 31, 25, 8, 32, 17 };
    int n = sizeof(a) / sizeof(a[0]);
    printf("Before sorting array elements are - \n");
    printArr(a, n);
    insert(a, n);
    printf("\nAfter sorting array elements are - \n");
    printArr(a, n);

    return 0;
}
```

●●Quick sort

```
#include <stdio.h>
/* function that consider last element as pivot,
place the pivot at its exact position, and place
smaller elements to left of pivot and greater
elements to right of pivot.  */
int partition (int a[], int start, int end)
{
```

```
    int pivot = a[end]; // pivot element
    int i = (start - 1);

    for (int j = start; j <= end - 1; j++)
    {
        // If current element is smaller than the pivot
        if (a[j] < pivot)
        {
            i++; // increment index of smaller element
            int t = a[i];
            a[i] = a[j];
            a[j] = t;
        }
    }
    int t = a[i+1];
    a[i+1] = a[end];
    a[end] = t;
    return (i + 1);
}

/* function to implement quick sort */
void quick(int a[], int start, int end) /
```

```
* a[] = array to be sorted, start = Starting index, end = Ending index */
{
    if (start < end)
    {
        int p = partition(a, start, end); //p is the partitioning index
        quick(a, start, p - 1);
        quick(a, p + 1, end);
    }
}

/* function to print an array */
void printArr(int a[], int n)
{
    int i;
    for (i = 0; i < n; i++)
        printf("%d ", a[i]);
}
int main()
{
    int a[] = { 24, 9, 29, 14, 19, 27 };
    int n = sizeof(a) / sizeof(a[0]);
    printf("Before sorting array elements are - \n");
```

```
    printArr(a, n);

    quick(a, 0, n - 1);

    printf("\nAfter sorting array elements are - \n");

    printArr(a, n);



    return 0;

}
```

●●bubble sort

```
#include<stdio.h>

 void print(int a[], int n) //function to print array elements

    {

    int i;

    for(i = 0; i < n; i++)
    {

        printf("%d ",a[i]);

    }

    }

 void bubble(int a[], int n) // function to implement bubble sort

{

   int i, j, temp;

   for(i = 0; i < n; i++)

    {
```

```
    for(j = i+1; j < n; j++)
      {
         if(a[j] < a[i])
         {
            temp = a[i];
            a[i] = a[j];
            a[j] = temp;
         }
      }
   }
 }
void main ()
{
   int i, j,temp;
   int a[5] = { 10, 35, 32, 13, 26};
   int n = sizeof(a)/sizeof(a[0]);
   printf("Before sorting array elements are - \n");
   print(a, n);
   bubble(a, n);
   printf("\nAfter sorting array elements are - \n");
   print(a, n);
}
```

●●Merge sort

```
#include <stdio.h>

#define max 10

int a[11] = { 10, 14, 19, 26, 27, 31, 33, 35, 42, 44, 0 };
int b[10];

void merging(int low, int mid, int high) {
  int l1, l2, i;

  for(l1 = low, l2 = mid + 1, i = low; l1 <= mid && l2 <= high; i++) {
    if(a[l1] <= a[l2])
      b[i] = a[l1++];
    else
      b[i] = a[l2++];
  }

  while(l1 <= mid)
    b[i++] = a[l1++];

  while(l2 <= high)
    b[i++] = a[l2++];

  for(i = low; i <= high; i++)
    a[i] = b[i];
}

void sort(int low, int high) {
  int mid;

  if(low < high) {
    mid = (low + high) / 2;
    sort(low, mid);
    sort(mid+1, high);
    merging(low, mid, high);
  } else {
    return;
  }
}

int main() {
```

```
  int i;

  printf("List before sorting\n");

  for(i = 0; i <= max; i++)
    printf("%d ", a[i]);

  sort(0, max);

  printf("\nList after sorting\n");

  for(i = 0; i <= max; i++)
    printf("%d ", a[i]);
}
```

●●selection Sort

```
#include <stdio.h>

// function to swap the the position of two elements
void swap(int *a, int *b) {
 int temp = *a;
 *a = *b;
 *b = temp;
}

void selectionSort(int array[], int size) {
 for (int step = 0; step < size - 1; step++) {
  int min_idx = step;
  for (int i = step + 1; i < size; i++) {

    // To sort in descending order, change > to < in this line.
    // Select the minimum element in each loop.
    if (array[i] < array[min_idx])
      min_idx = i;
  }

  // put min at the correct position
  swap(&array[min_idx], &array[step]);
 }
}

// function to print an array
void printArray(int array[], int size) {
```

```
  for (int i = 0; i < size; ++i) {
    printf("\n%d  ", array[i]);
  }
  printf("\n");
}

// driver code
int main() {
  int data[] = {20, 12, 10, 15, 2};
  int i;
printf("elementsof given array.....\n");
for(i=0;i<5;i++)
{
   printf("\n%d",data[i]);
}
  int size = sizeof(data) / sizeof(data[0]);
  selectionSort(data, size);
  printf("\nSorted arr""ay in Acsending Order:\n");
  printArray(data, size);
}
```

●●sum of all number

```
#include <stdio.h>
void main()
{
int i,n,sum=0;
printf("Enter the number : ");
scanf("%d",&n);
printf("\nThe square natural upto %d number  are :",n);
for(i=1;i<=n;i++)
{
printf("%d ",i*i);
sum+=i*i;
}
printf("\nThe Sum of Square Natural Number upto %d number = %d \n",n,sum);
}
```

●●ASCII VALUES

```
//Taking input from user for displaying ASCII value

#include <stdio.h>

int main()
```

```
{
  char ch;    // variable declaration

   printf("Enter a character:\n");

   scanf("%c",&ch);  // user input

   printf("\n The ascii value of the ch variable is : %d", ch);

   return 0;

}
```

●●fibonacci 1) with recurssion

```
#include<stdio.h>
void printfibo(int a, int b, int n)
{
 int x = a+b;
  if(x<n)
{
 printf("\n%d ", x);
 printfibo(b, x, n);
}
}
void main()
{
   int n;
   printf("enter the number upto you want fibonacci series:");
   scanf("%d", &n);
   printf("\n0 \n1 ");
   printfibo(0,1,n);
}
```

●●Fibonacci 2) without recurssion :

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

```
   printf("enter the number upto you want fibonacci series:");
   scanf("%d", &n);
   for(i=1; i<=n; i++)
   {
      printf("%d\n", c);
      a=b;
      b=c;
      c=a+b;
   }
}
```

●●factorial

```
#include<stdio.h>
int main()
{
    int no,f=1;
    printf("Enter any number\n");
    scanf("%d",&no);
    for(int i=1;i<=no;i++)
    {
        f=f*i;
    }
    printf("Factorial=%d",f);
}
```

●●reverse number

```
#include<stdio.h>

 int main()

{

int n, reverse=0, rem;

printf("Enter a number: ");

  scanf("%d", &n);

  while(n!=0)

  {
```

```
    rem=n%10;

    reverse=reverse*10+rem;

    n/=10;

  }

  printf("Reversed Number: %d",reverse);

return 0;

}
```

●●Armstrong number

```
#include<stdio.h>
 int main()
{
int n,r,sum=0,temp;
printf("enter the number=");
scanf("%d",&n);
temp=n;
while(n>0)
{
r=n%10;
sum=sum+(r*r*r);
n=n/10;
}
if(temp==sum)
printf("armstrong  number ");
else
printf("not armstrong number");
return 0;
}
```

●●sum of digits

```
#include<stdio.h>

 int main()

{

int n,sum=0,m;
```

```
printf("Enter a number:");
scanf("%d",&n);
while(n>0)
{
m=n%10;
sum=sum+m;
n=n/10;
}
printf("Sum is=%d",sum);
return 0;
}
```

●●decimal to binary

```
include<stdio.h>
#include<stdlib.h>
int main(){
int a[10],n,i;
system ("cls");
printf("Enter the number to convert: ");
scanf("%d",&n);
for(i=0;n>0;i++)
{
a[i]=n%2;
```

```
n=n/2;

}

printf("\nBinary of Given Number is=");

for(i=i-1;i>=0;i--)

{

printf("%d",a[i]);

}

return 0;

}
```

○●●○easy level

```
●●#include<stdio.h>
int main()
{
for(int i=1;i<=5;i++)
 {
    for(int j=1;j<=5;j++)
    {
       if(i>=j)
      printf("* ");
      else
      printf("  ");
    }
     printf("\n");
 }
}
```

```
### Output ###
*
* *
* * *
* * * *
* * * * *
```

```
●●#include<stdio.h>
int main()
{
for(int i=1;i<=5;i++)
 {
    for(int j=1;j<=5;j++)
    {
       if(i<=j)
      printf("* ");
      else
      printf("  ");
    }
     printf("\n");
 }
}
```

```
### Output ###
* * * * *
  * * * *
    * * *
      * *
        *
```

```
●●#include<stdio.h>
int main()
{
for(int i=1;i<=5;i++)
 {
    for(int j=1;j<=5;j++)
    {
       if((i+j)<=6)//change only this line in all program
      printf("* ");
      else
      printf("  ");
    }
     printf("\n");
 }
}
```

```
### Output ###
* * * * *
* * * *
```

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

●●
```
#include<stdio.h>
int main()
{
    for(int i=1;i<=5;i++)
    {
        for(int j=1;j<=5;j++)
        {
            if((i+j)>=6)
            printf("%d ",(i));
            else
            printf(" ");
        }
        printf("\n");
    }
}
```

Output

```
     1
   2  2
  3  3  3
 4  4  4  4
5  5  5  5  5
```

●●
```
#include<stdio.h>
int main()
{
    for(int i=1;i<=5;i++)
    {
        for(int j=1;j<=5;j++)
        {
            if(i>=j)
            printf("%c ",(j+64));
            else
            printf("  ");
        }
        printf("\n");
    }
}
```

Output
A
A B
A B C
A B C D
A B C D

●●#include<stdio.h>
int main()
{
    for(int i=1;i<=5;i++)
    {
        for(int j=1;j<=5;j++)
        {
            if(i>=j)
            printf("%d ",j);
            else
            printf(" ");
        }
        printf("\n");
    }
}

Output
1
1 2
1 2 3
1 2 3 4
1 2 3 4

●●include<stdio.h>
int main()
{
    for(int i=1;i<=5;i++)
    {
        for(int j=1;j<=5;j++)
        {
            if(i>=j)
            printf("%d ",i);
            else
            printf(" ");
        }
        printf("\n");

```
    }
}
```

Output

```
1
2  2
3  3  3
4  4  4  4
5  5  5  5
```

●●

```
#include<stdio.h>
int main()
{
for(int i=1;i<=5;i++)
 {
    for(int j=1;j<=5;j++)
    {
       if((i+j)>=6)//change only this line in all program
      printf("* ");
      else
      printf("  ");
    }
     printf("\n");
 }
}
```

Output

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

●●

```
#include<stdio.h>
 int main()
{
switch(printf("hello world")){}
return 0;
}
```

Print hello world without using semicolon

●●Swap two numbers without using third number

```
#include<stdio.h>
 int main()
{
int a=10, b=20;
printf("Before swap a=%d b=%d",a,b);
a=a+b;//a=30 (10+20)
b=a-b;//b=10 (30-20)
a=a-b;//a=20 (30-10)
printf("\nAfter swap a=%d b=%d",a,b);
return 0;
}
```

●●c function without main function

```
#include<stdio.h>
 #define start main
void start() {
   printf("Hello");
}
```

●●multiplication of array

```
#include<stdio.h>
#include<stdlib.h>
int main(){
int a[10][10],b[10][10],mul[10][10],r,c,i,j,k;
system("cls");
printf("enter the number of row=");
scanf("%d",&r);
printf("enter the number of column=");
scanf("%d",&c);
printf("enter the first matrix element=\n");
for(i=0;i<r;i++)
{
for(j=0;j<c;j++)
{
scanf("%d",&a[i][j]);
}
}
printf("enter the second matrix element=\n");
```

```
for(i=0;i<r;i++)
{
for(j=0;j<c;j++)
{
scanf("%d",&b[i][j]);
}
}

printf("multiply of the matrix=\n");
for(i=0;i<r;i++)
{
for(j=0;j<c;j++)
{
mul[i][j]=0;
for(k=0;k<c;k++)
{
mul[i][j]+=a[i][k]*b[k][j];
}
}
}
//for printing result
for(i=0;i<r;i++)
{
for(j=0;j<c;j++)
{
printf("%d\t",mul[i][j]);
}
```