ALL IMP PROGRAMS : C

```
/*
#include<stdio.h>
int main(){                                   //n=3       ***
        int n,i,j;                                         ***
        printf("Enter n");                                  ***
        scanf("%d", &n);
        for(i = 1; i<=n ; i++){
                for(j=1; j<=n ; j++){
                        printf("*");
                }
        printf("\n");
 }
 return 0;
}

*/

/*#include<stdio.h>                                //hollow rectangle
int main(){
        int n,i,j;
        printf("Enter n");
        scanf("%d", &n);
        for(i = 1; i<=n ; i++){
                for(j=1; j<=n ; j++){
                        if(i==1||j==1||i==n||j==n){
```

```
			printf("*");
		}
		else{
			printf(" ");
		}
		}
	printf("\n");
 }
 return 0;
}

*/

//  HALF PYRAMID

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

// n = 5
*
**
***
****
*****
*/


// inverted half pyramid
/*
#include<stdio.h>
int main(){
    int n,i,j;
    printf("Enter n");
    scanf("%d", &n);
    for(i = n;i>=1; i--){
        for(j =1 ; j<=i; j++){
            printf("*");
        }
    printf("\n");
```

```
        }
    return 0;
}

//n = 3
***
**
*

*/

/*
#include<stdio.h>
int main(){
    int i,j,n;
    printf("Enter n");
    scanf("%d",&n);
    for(i=1; i<=n;i++){
        for(j=1; j<=i ; j++){
            printf("%d", j);
        }
        printf("\n");
    }
}
```

```
*/

//o/p :
/*
1
12
123
*/

/*
#include<stdio.h>
int main(){
        int i,j,n;
        printf("Enter n");
        scanf("%d",&n);
        for(i=1; i<=n;i++){
                int p = (char)('A'+i-1);
                for(j=1; j<=i ; j++){
                        printf("%c", p);
                }
                printf("\n");
        }
}

//o/p :
```

```
A
BB
CCC
DDDD
EEEEE
*/

/*
#include<stdio.h>
int main(){
        int i,j,n;
        printf("Enter n");
        scanf("%d", &n);
        for(i=1; i<=n; i++){
                for(j =1; j<=n; j++){
                                int p =(char)('A'+j-1);
                                printf("%c", p);
                }
                printf("\n");
        }
}

//o/p :
ABC
ABC
ABC
```

```
*/

/*
#include<stdio.h>
int main(){
        int i,j,n;
        printf("Enter n");
        scanf("%d", &n);
        for(i=1; i<=n; i++){
                for(j =1; j<=n; j++){
                                int p =(char)('A'+i-1);
                                printf("%c", p);
                }
                printf("\n");
        }
}
*/

//inverted no. pyramid
/*
#include<stdio.h>
int main(){
        int i,j,n;
        printf("Enter n");
        scanf("%d", &n);
        for(i=n; i>=1; i--){
               for(j =1; j<=i; j++){
                              printf("%d", i);
               }
               printf("\n");
       }
}
*/


//Full pyramid of *
/*
#include <stdio.h>
int main() {
  int i, space, rows, k = 0;
  printf("Enter the number of rows: ");
  scanf("%d", &rows);
  for (i = 1; i <= rows; ++i, k = 0) {
    for (space = 1; space <= rows - i; ++space) {
      printf("  ");
    }
    while (k != 2 * i - 1) {
      printf("* ");                              //BE VERY CAREFUL OF ("*") AND ("* ")
      ++k;
    }
    printf("\n");
  }
  return 0;
}
*/
//  OOOOOOOOOOOOOOORRRRRRRRRRRRRRRRR
/*
#include<stdio.h>
int main(){
        int i,j,n;
        printf("Enter n");
        scanf("%d", &n);
        for(i=1;i<=n;i++){
                int spaces;
                for(spaces=1; spaces<=n-i; spaces++){
                        printf(" ");
                }
                for(j=1; j<=i;j++){
                        printf("*");
                }
                int x;
                for(x=i-1;x>=1;x--){
                        printf("*");
                }
                printf("\n");
        }
        return 0;
}
```

```
*/
/*
#include<stdio.h>
int main(){
        int i,j,n;
        printf("Enter n");
        scanf("%d",&n);
        for(i=1; i<=n;i++){
                int spaces;
                for(spaces=1; spaces<=n-i;spaces++){
                        printf(" ");
                }
                for(j=1; j<=i; j++){
                        printf("%d",j);
                }
                int x;
                for(x=i-1; x>=1; x--){
                        printf("%d",x);
                }
                printf("\n");
        }
        return 0;
}
*/
/* o/p:
    1
   121
  12321
 1234321
123454321
*/
/*
#include<stdio.h>
int main(){
	int i,j,n;
	printf("ENTER n");
	scanf("%d",&n);
	for(i=1; i<=n;i++){
		for(j=i; j<=n;j++){
			printf(" ");
		}
		int k;
		for(k=1; k<i*2;k++){
			printf("%d",k);
		}
		printf("\n");
	}
}
*/
//o/p
/*
```

```
n=5
      1
    123
   12345
  1234567
 123456789

 if j=1 instead of j=i;

     1
     123
     12345
     1234567
     123456789

*/

//BUTTERFLY PATTREN :
/*
#include <stdio.h>
int main()
{
    int x,i,j;
    printf("Enter x\n");
    scanf("%d",&x);
    int spaces = 2 * x - 2;
```

```
for( i = 1; i <= x;i++)
{
  for(j = 0;j < i;j++)
  {
    printf("*");
  }
  for( j=0;j<spaces;j++)
  {
    printf(" ");
  }
  for( j =0;j<i;j++)
  {
    printf("*");
  }
  printf("\n");
  spaces-=2;
}
spaces = 0;
for( i = x; i > 0;i--)
{
  for( j = 0;j < i;j++)
  {
    printf("*");
  }
  for( j=0;j<spaces;j++)
  {
      printf(" ");
     }
     for( j =0;j<i;j++)
     {
      printf("*");
     }
     printf("\n");
     spaces+=2;
   }


}

*/
// to print name ,phone no. of user
/*
#include<stdio.h>
int main(){
	char a[20];
	long long int b;
	printf("Enter your name");
	scanf(" %[^\n]s", &a);			// or we can use gets(a); instead of
writing scanf .
	printf("Enter your no.");
	scanf("%lld",&b);
	printf("Your name : %s \n Your no. : %lld", a,b);
	return 0;
```

```
}
*/

// prime or not
/*
#include<stdio.h>
int main(){
    int n ,i,flag=0;
    printf("Enter n");
    scanf("%d",&n);
    if(n==0 || n==1){
        flag =1;
    }
    for(i=1; i<=n/2; i++){
        if(n%i == 0){
            flag =1;
            break;
        }
    }
    if(flag>0){
        printf("NOT PRIME");
    }else{
        printf("PRIME");
    }
}
*/
```

```
// find prime factors


/*
#include<stdio.h>
int main(){
          int n,i;
          int count =0;
          printf("Enter n");
          scanf("%d",&n);
          for(i = 2; i <= n; i++) {            //Loop to check the factors.
   while(n % i == 0) {             //While the input is divisible to "i" which is initially
2.
       printf("%d ", i);             //Print the factor.
       n = n / i;                 //Divide the num by "i" which is initially 2 to change the
value of num.
       }
   }
 }
*/



//       HCF/GCD                          LCM = N1N2/GCD
/*
#include<stdio.h>


int main()
```

```
{
    int num1 = 36, num2 = 60, hcf = 1;
    int i;
    for(i = 1; i <= num1 || i <= num2; i++) {
        if(num1 % i == 0 && num2 % i == 0)
            hcf = i;
    }

    printf("The HCF: %d", hcf);
    int lcm = num1*num2/hcf;
    printf("The LCM is : %d", lcm);
    return 0;
}
*/

//Factorial
/*
#include<stdio.h>
int main(){
        int i=1;
        int fact =1;
        int n;
        printf("Enter n");
        scanf("%d",&n);
        while(i<=n){
                fact = fact*i;
                i++;
        }
        printf("%d",fact);
}
*/

// BINARY TO DECIMAL:
/*
#include<stdio.h>
#include<math.h>
int main(){
        int n,n1;
        printf("Enter decimal no.");
        scanf("%d",&n);
        int j = 0;
        int ans =0;
        while(n>0) {
           int n1 = n%10;
              n = n/10;
           ans = ans + pow(2,j)*n1;
          j++;
        }
 printf("%d",ans);
}
*/
/*
```

```
// DECIMAL TO BINARY
#include<stdio.h>
int main(){
        long long ans = 0;
        int i = 0;
        int n;
        printf("Enter decimal form of the no.");
        scanf("%d",&n);
        while(n>0) {
          int r = n%2;
          ans = ans + pow(10,i)*r;
          n = n/2;
          i++;
          printf("%lld",ans);
}
return 0;
}
*/
//INVERTED M SHAPED PATTREN
/*
#include<stdio.h>

int main()
{
int n, i,  j, k;
printf("Enter the number of rows to show number pattern: ");
scanf("%d",&n);
for(i =1; i <= n; i++)
{
for(j =1; j <= n; j++)
{
if(j<= i)
printf("%d",j);
else
printf(" ");
}
for(j = n; j>= 1;j--)
{
if(j <= i)
printf("%d",j);
else
printf(" ");
}
printf("\n");
}
return 0;
}
*/

// n =5
/*
1       1
```

```
12      21

123    321

1234  4321

1234554321

*/


//DIAMOND PATTREN (nO. FORM) ::::::::::::::::::::::::::::::::IMPORTANT:::::::::::::::::::::::::

/*

#include<stdio.h>

int main()

{

int n, x, y, k;

printf("Enter the number of rows to show number paatern: ");

scanf("%d",&n);

for(x = 1; x <= n; x++)

{

for(y = x; y <n; y++)

{

printf(" ");

}

for(k = 1; k < (x*2); k++)

{

printf("%d",k);

}

printf("\n");

}
```

```
for(x = n-1; x >= 1; x--)
{
for(y = n; y >  x; y--)
{
printf(" ");
}
for(k = 1; k < (x*2); k++)
{
printf("%d",k);
}
printf("\n");
}
return 0;
}
*/

//VERTICAL DIAMOND
/*::::::::::::::::::::::::::::::::::::::::::::::::::::::::IMP QUS::::::::::::::::::::::::::::::::::::::::
#include<stdio.h>
int main()
{
int  n, x, y;
printf("Enter the number of rows to show number pattern: ");
scanf("%d",&n);
for(x = 1; x < n; x++)
{
```

```
for( y = 1; y <= x; y++)
printf("%d",y);
printf("\n");
}
for( x = n; x >= 0; x--)
{
for( y = 1; y <= x; y++)
printf("%d",y);
printf("\n");
}
return 0;
}
*/

// SWAP 2 NOS. WITH 2 VARIABLE[NOT USING TEMP]

/*
a=10 , b=20

a=a+b;//a=30 (10+20)
b=a-b;//b=10 (30-20)
a=a-b;//a=20 (30-10)

*/

//ARMSTRONG NO.
```

```
/*
#include<stdio.h>
int main(){
        int n,a;
        int sum =0;
        printf("Enter n");
        scanf("%d",&n);
        int b =n;
        while(n>0){
        a = n%10;
        sum = sum + pow(a,3);
        n = n/10;
   }
 if(sum == b){
        printf("ARMSTRONG NO.");
 }else
 {
        printf("NOT A ARMSTRONG NO.");
 }
        return 0;
}

*/

//PRINT ALL ARMSTRONG NOS. BETWEEN 1 TO 500;
```

```
/*::::::::::::::::::::::::::::::::::::::::::::::::::::::::::::::IMP::::::::::::::::::::::::::::::::::::::::::::
#include<stdio.h>



int main()
{
   int n,sum,i,t,a;
   printf("\n\n\nThe Armstrong numbers in between 1 to 500 are : \n\n\n");

   for(i = 1; i <= 500; i++)
   {
      t = i;  // as we need to retain the original number
      sum = 0;
      while(t != 0)
      {
         a = t%10;
         sum += a*a*a;
         t = t/10;
      }

      if(sum == i)
      printf("\n\t\t\t%d", i);
   }
   return 0;
}
```

```
*/

//MULTIPLICATION TABLE
/*
#include<stdio.h>
int main(){
        int i,n;
        printf("Enter n");
        scanf("%d",&n);
        for(i=0; i<=n ; i++){
                int c = n*i;
                printf("%d \n",c);
        }
}
*/

//TO FIND RANGE FROM ENTERED NOS.

/*

#include<stdio.h>
int main()
{
   int small, big, range, num, limit;

   printf("Enter the limit\n");
```

```
scanf("%d", &limit);


printf("Enter %d numbers\n", limit);
scanf("%d", &num);


small = big = num;


limit = limit - 1;


while(limit)
{
   scanf("%d", &num);


   if(num > big)
   {
      big = num;
   }


   if(num < small)
   {
      small = num;
   }


   limit--;
}
```

```
    range = big - small;


    printf("Small Number = %d\nBig Number = %d\n", small, big);
    printf("Range is %d\n", abs(range));


    return 0;
}

*/

////////////////////////////////////////////////////////////

/*



VARIANCE  =  ((OriginalValue – Mean)² + (OriginalValue – Mean)² +…. )     /
Total number of items



MEAN = Sum of each individual/total number of items


STANDARD DEVIATION = ROOT OF VARIANCE



Compound Interest = P(1 +(r/100))^n  - P


n = no. of times interest got compounded anually
```

```
*/

//REVERSE NO.
/*
#include<stdio.h>
int main(){
        int n,a;
        int b=0;
        printf("Enter n");
        scanf("%d",&n);
        while(n>0){
                a = n%10;
                b = b*10 + a;
                n = n/10;
        }
        printf("%d",b);
}

*/

//LARGEST AMONG 3:
/*
#include<stdio.h>
int main()
{
	int a,b,c;

	printf("Enter a,b,c");

	scanf("%d %d %d", &a,&b,&c);

	(a>b&&a>c)?printf("a is greater"):(b>c)?printf(" b is greater"):printf(" c is greater");


}

*/




//LEAP YEAR

/*

#include<stdio.h>

int main(){

	int n;

	printf("Enter n");

	scanf("%d",&n);

	if((n % 4 == 0) && (n % 100 != 0) || (n % 400 == 0)){

		printf("LEAP YR");

	}else{

		printf("NOT A LEAP YR");

	}

}

*/




// PALINDROME NO. ::    IF NO. AFTER REVERSING IS SAME AS THAT OF INITIAL
// Fibonnaci series:


/*
#include<stdio.h>
int main(){
        int n,a,b,c;
        int i =2;
        printf("Enter n");
        scanf("%d",&n);
        a=0;
        b=1;
        printf("\n%d %d ",a,b);
        for(i=2;i<n;i++){
                c = a + b;
                printf(" %d",c);
                a = b;
                b = c;
        }
}
*/
```