Name : Akshay Singh

University Roll No. 2215000167

Class Roll No. 7

Section - G

1. C program to perform all arithmetic operations

```
#include<stdio.h>
int main()
{
    int a,b,c;
    float x;
    printf("\nEnter 2 Nos : ");
    scanf("%d%d",&a,&b);
    c=a+b;
    printf("\nTotal : %d",c);
    c=a-b;
    printf("\nDifference : %d",c);
    c=a*b;
    printf("\nMul : %d",c);
    x=(float)a/(float)b;
    printf("\nDiv : %0.2f",x);
    c=a%b;
    printf("\nMod : %d",c);
    return 0;
}
```

2. C program to find area of a triangle if base and height are given.

```
#include <stdio.h>
int main()
{
  float base, height, area;
  printf("Enter base of the triangle: ");
   scanf("%f", &base);
   printf("Enter height of the triangle: ");
   scanf("%f", &height);
   area = (base * height) / 2;
   printf("Area of the triangle = %f", area);
```

```
   return 0;
}
```

3. C program to find all angles of a triangle if two angles are given.

```
#include <stdio.h>
int main()
{
    int ang1, ang2, ang3;
    printf("Input two angles of triangle : ");
    scanf("%d%d",&ang1,&ang2);
    ang3 = 180 - (ang1 + ang2);
    printf("Third angle of the triangle :  %d\n", ang3);
    return 0;
 }
```

4. C program to convert days in to years, weeks and days.

```
#include <stdio.h>
int main(){
    int days, years, weeks;
     printf("Enter the days  you want convert into years ,week :  ");
     scanf("%d",&days );
    years = days/365;
    weeks = (days % 365)/7;
    days = days- ((years*365) + (weeks*7));
    printf("Years: %d\n", years);
    printf("Weeks: %d\n", weeks);
    printf("Days: %d \n", days);
    return 0;
}
```

5. C program to find power and square root of any number.

```
#include <stdio.h>
#include <math.h>
int main(){
    double num, root;
    printf("Enter any number to find square root: ");
    scanf("%lf", &num);
    root = sqrt(num);
    printf("Square root of %.2lf = %lf", num, root);
    float base, exp, result;
    printf("\nEnter a base number: ");
    scanf("%f", &base);
```

```
    printf("Enter an exponent: ");
    scanf("%f", &exp);
    result = pow(base, exp);
    printf("result = %.2f",result);
    return 0;
}
```

6. C program to calculate total, average and percentage and grades of five subjects.

```
#include <stdio.h>
int main()
{
    float eng, phy, chem, math, comp;
    float total, average, percentage;
    char grade;
    printf("Enter marks of five subjects: \n");
    scanf("%f%f%f%f%f", &eng, &phy, &chem, &math, &comp);
    total = eng + phy + chem + math + comp;
    average = total / 5.0;
    percentage = (total / 500.0) * 100;
    if ( percentage>= 90)
        grade = 'A';
    else if (percentage >= 80 && percentage < 90)
        grade = 'B';
    else if (percentage >= 70 && percentage < 80)
        grade = 'C';
    else if (percentage >= 60 && percentage < 70)
        grade = 'D';
    else
        grade = 'E';
    printf("Total marks = %f\n", total);
    printf("Average marks = %f\n", average);
    printf("Percentage = %f", percentage);
    printf("grade = %c",grade);
    return 0;
}
```

7. C program to check Least Significant Bit (LSB) and MSB of a number using bitwise operator.

```
#include <stdio.h>
#define BITS sizeof(int) * 8 // Total bits required to represent integer
int main()
{
    int num, msb;
    printf("Enter any number: ");
    scanf("%d", &num);
    if(num & 1)
        printf("LSB of %d is set (1).", num);
    else
        printf("LSB of %d is unset (0).", num)
    msb = 1 << (BITS - 1);
    if(num & msb)
        printf("MSB of %d is set (1).", num);
    else
        printf("MSB of %d is unset (0).", num);
    return 0;
}
```

8. C program to swap two numbers USING 3RD VARIABLE AND WITHOUT 3RD VARIABLE.

Using 3rd variable :

```
#include <stdio.h>
int main()
{
int var1, var2, temp;
printf("Enter two integersn");
scanf("%d%d", &var1, &var2);
printf("Before SwappingnFirst variable = %dnSecond variable = %dn", var1, var2);
temp = var1;
var1 = var2;
var2 = temp;
printf("After SwappingnFirst variable = %dnSecond variable = %dn", var1, var2);
return 0;
}
```

Without using 3rd variable :

```
#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;
}
```

9. C program to find maximum between three numbers using conditional operator AND Ternary Operator.

```
#include <stdio.h>
int main() {
    int a, b, c, max;
    printf("Enter Three Integers\n");
scanf("%d %d %d", &a, &b, &c);
max = (a > b) ? ((a > c) ? a : c) : ((b > c) ? b : c);
printf("Maximum Number is = %d\n", max);
return 0;
}
```

10. C program to check alphabet, digit or special character using Conditional operator.

```
#include <stdio.h>
int main()
{
 char ch;
 printf("Enter any character: ");
 scanf("%c", &ch);
if((ch >= 'a' && ch <= 'z') || (ch >= 'A' && ch <= 'Z'))
  {
printf("'%c' is alphabet.", ch);
  }
 else if(ch >= '0' && ch <= '9')
  {
    printf("'%c' is digit.", ch);
  }
  else
  {
    printf("'%c' is special character.", ch);
  }
  return 0;
}
```

11. C program to calculate total electricity bill.

```
#include <stdio.h>
int main()
{
   int unit;
   float amt, total_amt, sur_charge;
  printf("Enter total units consumed: ");
   scanf("%d", &unit);
  if(unit <= 50)
   {
      amt = unit * 0.50;
   }
   else if(unit <= 150)
   {
      amt = 25 + ((unit-50) * 0.75);
   }
   else if(unit <= 250)
   {
      amt = 100 + ((unit-150) * 1.20);
   }
   else
   {
      amt = 220 + ((unit-250) * 1.50);
   }

   sur_charge = amt * 0.20;
   total_amt  = amt + sur_charge;

   printf("Electricity Bill = Rs. %.2f", total_amt);
```

```
   return 0;
}
```

12. C program to create Simple Calculator AND Days of week using switch case.

```
#include <stdio.h>

int main()
{
    char ch;
    int num1,num2;

    printf("Choose the operator(+,-,*,/,%%): ");
    scanf("%c",&ch);

    printf("Enter two numbers: ");
    scanf("%d %d",&num1,&num2);

    switch(ch)
    {
      case '+':
        printf("%d + %d =\t%d\n",num1,num2,num1+num2);
        break;
      case '-':
        printf("%d - %d =\t%d\n",num1,num2,num1-num2);
        break;
      case '*':
        printf("%d * %d =\t%d\n",num1,num2,num1*num2);
        break;
      case '/':
        printf("%d / %d =\t%d\n",num1,num2,num1/num2);
        break;
      case '%':
        printf("%d %% %d =\t%d\n",num1,num2,num1%num2);
```

```
    break;
  default:
    printf("Error! Invalid Operator.");
 }

int week;

printf("Enter week number(1-7): ");
scanf("%d", &week);

switch(week)
{
   case 1:
      printf("Monday");
      break;
   case 2:
      printf("Tuesday");
      break;
   case 3:
      printf("Wednesday");
      break;
   case 4:
      printf("Thursday");
      break;
   case 5:
      printf("Friday");
      break;
   case 6:
      printf("Saturday");
      break;
   case 7:
      printf("Sunday");
      break;
   default:
```

```
        printf("Invalid input! Please enter week number between 1-7.");
    }

    return 0;
}
```

13. C program to check vowel or consonant using switch case.

```
#include <stdio.h>
int main() {
char c;
printf("Enter an Alphabet\n");
scanf("%c", &c);
switch(c) {
case 'a':
case 'A':
case 'e':
case 'E':
case 'i':
case 'I':
case 'o':
case 'O':
case 'u':
case 'U':
      printf("%c is VOWEL", c);
break;
default: printf("%c is CONSONANT", c);
}
return 0;
}
```

14. C program to check positive negative or zero using switch case.

```
#include<stdio.h>
int main()
{
   printf("1. Check Zero \n");
   printf("2. Check Positive \n");
   printf("3. Check Negative \n");

   int choice;
   printf("Enter your choice : ");
   scanf("%d",&choice);

   int num;
   printf("Enter Number : ");
   scanf("%d", &num);
   switch(choice)
   {
      case 1 :
      {
         if(num == 0)
         {
            printf("It is zero");
         }
         else{
            printf("Number isn't zero");
         }
         break;
      }
      case 2:
      {
```

```
        if(num > 0)
        {
            printf("Number is Positive");
        }
        else
        {
            printf("Number not is Positive");
        }
        break;
    }
    case 3:
    {
        if(num < 0)
        {
            printf("Number is Neagtive");
        }
        else
        {
            printf("Number not is Neagtive");
        }
        break;
    }

    default :
    {
        printf("Enter a valid choice !");
    }
  }

  return 0;
}
```

15. C program to check whether a triangle is Equilateral, Isosceles or Scalene.

```
#include <stdio.h>

int main()
{
   int side1, side2, side3;

   printf("Enter three sides of triangle: ");
   scanf("%d%d%d", &side1, &side2, &side3);

   if(side1==side2 && side2==side3)
   {
      printf("Equilateral triangle.");
   }
   else if(side1==side2 || side1==side3 || side2==side3)
   {
      printf("Isosceles triangle.");
   }
   else
   {
      printf("Scalene triangle.");
   }

   return 0;
}
```

16. C program to print all natural numbers AND sum of it from 1 to n.

```
#include<stdio.h>
int main()
{
                                        int Number, i,sum=0;
                                       printf("\n Please Enter  Value  : ");
                                        scanf("%d", &Number);
                                        printf("\n List of Natural Numbers from 1 to %d are \n", Number);
                                        for(i = 1; i <= Number; i++)
                                        {
                                        printf(" %d \t", i);
                                        }
     printf("Enter upper limit: ");
   scanf("%d", &Number);
  for(i=1; i<=Number; i++)
   {
      sum += i;
   }
 printf("Sum of first %d natural numbers = %d", Number, sum);
 return 0;
}
```

17. C program to print all even numbers AND sum of it from 1 to n.

```
#include <stdio.h>
int main()
{
   int i, n, sum=0;
   printf("Enter any number: ");
   scanf("%d", &n);
   for(i=2; i<=n; i+=2)
   {
      sum += i;
   }
   printf("Sum of all even numbers from 1 to %d: %d", n, sum);

   printf(" \n Enter upper limit: ");
   scanf("%d", &n);

   for(i=2; i<=n; i+=2)
   {
      sum += i;
   }

   printf("Sum of all even number between 1 to %d = %d", n, sum);

   return 0;
}
```

18. C program to print multiplication table of a number.

```
#include <stdio.h>
int main()
{
   int n, i;
   printf("Enter a number: ");
   scanf("%d", &n);
   printf("Multiplication table of %d:\n ", n);
   printf(" \n");
   for (i = 1; i <= 10; i++)
      printf("%d x %d = %d\n", n, i, n * i);
   return 0;
}
```

19. C program to calculate factorial of a number.

```
#include <stdio.h>
int  main(){
  int i,f=1,num;

  printf("Input the number : ");
  scanf("%d",&num);

  for(i=1;i<=num;i++)
     f=f*i;

  printf("The Factorial of %d is: %d\n",num,f);
  return 0;
}
```

20. C program to check whether a number is palindrome or not.

```
#include <stdio.h>
int main() {
  int n, reversed = 0, remainder, original;
    printf("Enter an integer: ");
    scanf("%d", &n);
    original = n;

    while (n != 0) {
        remainder = n % 10;
        reversed = reversed * 10 + remainder;
        n /= 10;
    }

    if (original == reversed)
        printf("%d is a palindrome.", original);
    else
        printf("%d is not a palindrome.", original);

    return 0
}
```

21. C program to count frequency of digits in a given number.

```
#include<stdio.h>
int main( )
{
  int n, i, arr[50], inc=0,  j, count, m;
  printf("enter the number");
  scanf("%d",&n);
  m=n;
 while(m>0)
  {
     i=m%10;
     m=m/10;
     arr[inc]=i;
     inc++;
  }

  for(i=0; i<=9 ; i++)
   {
      count=0;
      for(j=0; j<inc ; j++)
       {
           if(arr[j] == i)
              count++;
       }
      printf("frequency of %d is %d",n,count);
   }
return 0;
}
```

22. C program to find HCF(GCD) AND LCM of two numbers.

```
#include <stdio.h>
 void main()
{
   int num1, num2, gcd, lcm, remainder, numerator, denominator;

   printf("Enter two numbers:\n");
   scanf("%d %d", &num1, &num2);

   numerator = (num1>num2)?num1:num2;
   denominator = (num1<num2)?num1:num2;
   remainder = numerator % denominator;

   while (remainder != 0)
   {
      numerator   = denominator;
      denominator = remainder;
      remainder   = numerator % denominator;
   }
   gcd = denominator;
   lcm = num1 * num2 / gcd;
   printf("GCD of %d and %d = %d\n", num1, num2, gcd);
   printf("LCM of %d and %d = %d\n", num1, num2, lcm);
}
```

23. C program to print all prime numbers between 1 to n.

```
#include<stdio.h>

int main(){
int N, i, j, Prime, n;

printf("To print all prime numbers between 1 to N\n");
printf("Enter the value of N\n");
scanf("%d",&N);

printf("Prime numbers between %d to %d\n", 1, N);
for(i = 2; i <= N; i++) {
 Prime = 0;
for(j = 2; j <= i/2; j++){
if(i % j == 0){
Prime = 1;
  break;
}

}

if(Prime==0 && N!= 1)
printf("%d ",i);
}
return 0;

}
```

24. C program to print all Strong Numbers between 1 to n.

```
#include <stdio.h>
int main()
{
   int i, j, cur, lastDigit, end;
   long long fact, sum;
   printf("Enter upper limit: ");
   scanf("%d", &end);
   printf("All Strong numbers between 1 to %d are:\n", end);
   for(i=1; i<=end; i++)
   {
      cur = i;
      sum = 0;
      while(cur > 0)
      {
         fact = 1ll;
         lastDigit = cur % 10;
         for( j=1; j<=lastDigit; j++)
         {
            fact = fact * j;
         }
         sum += fact;
         cur /= 10;
      }
      if(sum == i)
      {
         printf("%d, ", i);
      }
   }
   return 0;
```

```
}
```

25. C program to print Fibonacci series up to n terms.

```
#include<stdio.h>
int main()
{
   int a = 0, b = 1;
   int c;
   int n;
   printf("How many terms you want to print ?\n");
   scanf("%d",&n);

   if(n == 1)
   {
      printf("%d", a);
   }
   else if(n == 2)
   {
      printf("%d\n", a);
      printf("%d", b);
   }
   else if(n > 2)
   {
      printf("%d\n", a);
      printf("%d\n", b);
      while((n - 2) > 0)
      {
         c = a + b;
         printf("%d\n",c);
         a = b;
         b = c;
         n--;
```

```
    }
  }
  return 0;
}
```

26. C program to print Armstrong numbers from 1 to n AND Check a given number is Armstrong numbers or not.

```
#include <stdio.h>
int main()
{
        int n, r, sum = 0, temp;
        printf("enter the number = ");
        scanf("%d", &n);
        int copy = n;
        temp = n;
        while (n > 0)
        {
                r = n % 10;
                sum = sum + (r * r * r);
                n = n / 10;
        }
          if (temp == sum)
                printf("%d is an armstrong number\n", temp);
          else
                printf("%d is not armstrong number\n\n", temp);
  printf("Armstrong numbers in given range are: ");
  for (int i = 1; i <= copy; i++)
  {
        int temp1 = i;
        sum = 0;
        while (temp1 != 0)
        {
                r = temp1 % 10;
                sum = sum + (r * r * r);
                temp1 = temp1 / 10;
        }
        if (sum == i)
                printf("%d \n", i);
```

```
  }
  printf("\n");
  return 0;
}
```

27. C program to print all Perfect numbers between 1 to n AND Check a given number is Perfect numbers or not.

```
#include <stdio.h>
int main()
{
   int n, rem, sum = 0, i;
   printf("Enter a number\n");
   scanf("%d", &n);
   for (i = 1; i < n; i++)
   {
      rem = n % i;
      if (rem == 0)
      {
         sum = sum + i;
      }
   }
   if (sum == n)
      printf("%d is a Perfect Number\n",n);
   else
      printf("%d is not a Perfect Number\n",n);
   printf("Perfect numbers in the given range are : ");
   for(int x = 2; x <= n; x++)
   {
      sum = 0;
      for(int j = 1; j < x; j++)
      {
         rem = x % j;
         if(rem == 0) {
            sum += j;
         }
      }
      if(x == sum) {
         printf("%d \n",x);
```

```
        }
    }
    return 0;
}
```

28. C program to find power of any number using for loop.

```
#include<stdio.h>
int main()
{
    int base, power, result = 1;
    printf("Enter Base Value (b) : ");
    scanf("%d", &base);
    printf("Enter power value (p) : ");
    scanf("%d", &power);

    for(int i = 0; i < power; i++)
        result *= base;

    printf("Answer = %d",result);
}
```

29. C program to print ASCII values of all characters.

```
#include <stdio.h>
int main()
{
    printf("All ASCII Values : \n");
    for(int i = 0; i <= 255; i++)
    {
        printf("%c \t\t %d \n", i, i);
    }
    return 0;
}
```

30. C program to print Pascal triangle up to n rows.

```
#include <stdio.h>
int factorial(int n)
{
   int f;
   for (f = 1; n > 1; n--)
      f *= n;

   return f;
}
int ncr(int n, int r)
{
   return factorial(n) / (factorial(n - r) * factorial(r));
}
int main()
{
   int n, i, j;
   printf("Enter Number of rows (n) : ");
   scanf("%d", &n);
   for (i = 0; i <= n; i++)
   {
      for (j = 0; j <= n - i; j++)
         printf("  ");

      for (j = 0; j <= i; j++)
         printf(" %3d", ncr(i, j));

      printf("\n");
   }
   return 0;
```

```
}
```

31. C program to find sum of all elements of array.

```
#include<stdio.h>
#define SIZE 50
 int main()
{
   int A[SIZE],i,n,sum=0;
   printf("Enter the size of Array : ");
   scanf("%d",&n);
   printf("Enter Array Elements : \n");
   for ( i = 0; i < n; i++)
   {
      scanf("%d",&A[i]);
   }

   for ( i = 0; i < n; i++)
   {
      sum += A[i];
   }
   printf("Sum of all Elements of Array = %d \n",sum);
   return 0;
}
```

32. C program to copy one array to another array.

```
#include <stdio.h>
int main()
{
   int size;
   printf("Enter size of array : ");
   scanf("%d", &size);
   int arr[size];

   printf("Enter Array Elements : \n");
   for (int i = 0; i < size; i++)
   {
      scanf("%d", &arr[i]);
   }
   printf("\nOriginal Array Elements : ");
   for (int i = 0; i < size; i++)
   {
      printf("%d  ", arr[i]);
   }

   int copyArr[size];
   for (int i = 0; i < size; i++)
   {
      copyArr[i] = arr[i];
   }
   printf("\nCopy Array : ");
   for (int i = 0; i < size; i++)
   {
      printf("%d  ", copyArr[i]);
   }
```

```
    return 0;
}
```

33. C program to insert an element in array at specified position.

```
#include <stdio.h>
#include <iomanip>
const int size = 50;
int main()
{
    int len;
    printf("Enter length of array : ");
    scanf("%d", &len);
    int arr[len];
    printf("\nEnter Array Elements : ");
    for (int i = 0; i < len; i++)
    {
        scanf("%d", &arr[i]);
    }
    for (int i = 0; i < len; i++)
    {
        printf("%d  ", arr[i]);
    }
    int pos, element;

    printf("\nEnter position where you want to insert element : ");
    scanf("%d", &pos);

    if (pos > len + 1)
    {
        printf("\nInvalid Position!!");
        exit(0);
    }
    else
```

```
    {
        printf("\nEnter element : ");
        scanf("%d", &element);
        int curPos = len;
        while (curPos >= pos)
        {
            arr[curPos] = arr[curPos - 1];
            curPos--;
        }
        arr[pos - 1] = element;
        len = len + 1;
    }
    printf("\nArray after insertion : ");
    for (int i = 0; i < len; i++)
    {
        printf("%d  ", arr[i]);
    }
    return 0;
}
```

34. C program to delete an element in array at specified position.

```
#include <stdio.h>
int main()
{
    int array[100], position, c, n;
    printf("Enter number of elements in array\n");
    scanf("%d", &n);
    printf("Enter %d elements\n", n);
    for (c = 0; c < n; c++)
        scanf("%d", &array[c]);
    printf("Enter the location where you wish to delete element\n");
    scanf("%d", &position);
    if (position >= n + 1)
        printf("Deletion not possible.\n");
```

```
    else
    {
        for (c = position - 1; c < n - 1; c++)
            array[c] = array[c + 1];

        printf("Resultant array is\n");
        for (c = 0; c < n - 1; c++)
            printf("%d\n", array[c]);
    }
    return 0;
}
```

35. C program to search element in array using Linear Search.

```
#include<stdio.h>
#define SIZE 50
 int main()
{
    int A[SIZE],i,n,x,a=0;
    printf("Enter the size of Array : ");
    scanf("%d",&n);
    printf("Enter Array Elements :\n");
    for ( i = 0; i < n; i++)
    {
        scanf("%d",&A[i]);
    }
    for ( i = 0; i < n; i++)
    {
        printf("%d ",A[i]);
    }
    printf("\n Enter the number to search in Array : ");
    scanf("%d",&x);
    for ( i = 0; i < n; i++)
    {
        if (x==A[i])
```

```
    {
        printf("\nElement(%d) found at position %d",x,i+1);
        a=1;
        break;
    }
  }
  if(a==0)
  {
    printf("\nElement Not Found!!");
  }
  return 0;
}
```

36. C program to find second largest number and Sorting Using Bubble sort in an array.

```
#include <stdio.h>
#define SIZE 50
int main()
{
  int A[SIZE], i, j, temp, n;
  printf("Enter the size of array : \n");
  scanf("%d", &n);
  printf("Enter Array Elements :\n");
  for (i = 0; i < n; i++)
  {
    scanf("%d", &A[i]);
  }
  printf("\n\n");
  printf("Array Elements : ");
  for (i = 0; i < n; i++)
  {
    printf("%d ", A[i]);
  }
  printf("\n\n");
  for (i = 0; i < n - 1; i++)
```

```
    {
        for (j = i + 1; j < n; j++)
        {
            if (A[i] > A[j])
            {
                temp = A[i];
                A[i] = A[j];
                A[j] = temp;
            }
        }
    }
    printf("\n\nArray after sorting : ");
    for (i = 0; i < n; i++)
    {
        printf("%d  ", A[i]);
    }
    printf("2nd Biggest Number = %d", A[n - 2]);
    return 0;
}
```

37. C program to count total number of duplicate elements in an array.

```
#include<stdio.h>
#define SIZE 50
 int main()
{
    int A[SIZE],Frequency[SIZE],i,j,n,count;
    printf("Enter the size of array : ");
    scanf("%d",&n);
    printf("\nEnter array elements : \n");
    for(i=0;i<n;i++)
    {
        scanf("%d",&A[i]);
        Frequency[i] = -1;
    }
```

```
   for ( i = 0; i < n; i++)
   {
      count = 1;
      for ( j = i+1; j < n; j++)
      {
         if(A[i]==A[j])
         {
            count++;
            Frequency[j] = 0;
         }
      }
      if (Frequency[i] != 0)
      {
         Frequency[i] = count;
      }

   }
   for ( i = 0; i < n; i++)
   {
      if (Frequency[i] != 0)
      {
         printf("\nFrequency of %d is = %d",A[i],Frequency[i]);
      }

   }
   return 0;
}
```

38. C program to perform scalar matrix multiplication.

```
#include <stdio.h>
int main()
{
   int row, col, multiplier, sum = 0;
```

```
printf("Rows of Array : ");
scanf("%d", &row);
printf("Columns of Array : ");
scanf("%d", &col);

int A[row][col];
printf("Enter Array Elements : \n");
for (int i = 0; i < row; i++)
{
   for (int j = 0; j < col; j++)
   {
      scanf("%d", &A[i][j]);
   }
}
for (int i = 0; i < row; i++)
{
   for (int j = 0; j < col; j++)
   {
      printf("%d ", A[i][j]);
   }
   printf("\n");
}

printf("Scalar Multiplier : ");
scanf("%d", &multiplier);

for (int i = 0; i < row; i++)
{
   for (int j = 0; j < col; j++)
   {
      A[i][j] = multiplier * A[i][j];
   }
}
printf("Updated Array after scalar multiplication : \n");
```

```
    for (int i = 0; i < row; i++)
    {
        for (int j = 0; j < col; j++)
        {
            printf("%d ", A[i][j]);
        }
        printf("\n");
    }
    return 0;
}
```

39. C program to find sum of main diagonal elements of a matrix.

```
#include <stdio.h>
int main()
{
    int row, col, sum = 0;
    printf("Rows of Array : ");
    scanf("%d", &row);
    printf("Columns of Array : ");
    scanf("%d", &col);

    int A[row][col];
    printf("Enter Array Elements : \n");
    for (int i = 0; i < row; i++)
    {
        for (int j = 0; j < col; j++)
        {
            scanf("%d", &A[i][j]);
        }
    }
    for (int i = 0; i < row; i++)
    {
        for (int j = 0; j < col; j++)
        {
```

```
            printf("%d ", A[i][j]);
        }
        printf("\n");
    }
    for (int i = 0; i < row; i++)
    {
        for (int j = 0; j < col; j++)
        {
            if (i == j)
            {
                sum += A[i][j];
            }
        }
    }
    printf("Diagonal Sum = %d", sum);
    return 0;
}
```

40. C program to check sparse AND transpose matrix.

```
#include <stdio.h>
int main()
{
        int i, j, r, c, count = 0;
        printf("Enter no of rows and columns in matrix \n");
        scanf("%d %d", &r, &c);
        int a[r][c];
        printf("Enter the value of martix ELEMENTS : ");

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

```
for (i = 0; i < r; i++)
{
    for (j = 0; j < c; j++)
    {
            if (a[i][j] == 0)
            {
                    count++;
            }
    }
}

int size = r * c;
printf("Rows : %d \n",r);
printf("Columns : %d \n",c);
printf("Size : %d \n",size);
for (i = 0; i < r; i++)
{
    for (j = 0; j < c; j++)
    {
            printf("%d   ", a[i][j]);
    }
    printf("\n");
}

if (count > (size / 2))
{
    printf("Sparse matrix.");
}
else
{
    printf("no");
}
return 0;
```

```
}
```

41. C program to check whether a matrix is Identity matrix or not.

```
#include <stdio.h>
int main()
{
   int row, col, sum = 0;
   printf("Rows of Array : ");
   scanf("%d", &row);
   printf("Columns of Array : ");
   scanf("%d", &col);

   int A[row][col];
   printf("Enter Array Elements : \n");
   for (int i = 0; i < row; i++)
   {
      for (int j = 0; j < col; j++)
      {
         scanf("%d", &A[i][j]);
      }
   }
   for (int i = 0; i < row; i++)
   {
      for (int j = 0; j < col; j++)
      {
         printf("%d ", A[i][j]);
      }
      printf("\n");
   }
   int flag = 1;
   for (int i = 0; i < row; i++)
   {
```

```
        for (int j = 0; j < col; j++)
        {
            if ((i == j && A[i][j] == 1))
            {
                flag = 1;
            }
            else if (i != j && A[i][j] == 0)
            {
                flag = 1;
            }
            else
            {
                flag = 0;
                break;
            }
        }
    }
    if (flag == 0)
    {
        printf("False");
    }
    else
    {
        printf("True");
    }
    return 0;
}
```

42. C program to merge two sorted array in ascending order.

```
#include <stdio.h>
int main()
{
    int len1, len2;
    printf("Size of array 1 (l1) : ");
```

```
scanf("%d", &len1);
printf("Size of array 2 (l2) : ");
scanf("%d", &len2);

int sLen = (len1 > len2) ? (len2) : (len1);
int a1[len1];
int a2[len2];
int result[len1 + len2];
printf("Enter Array1 Elements : \n");
for (int i = 0; i < len1; i++)
{
   scanf("%d", &a1[i]);
}
printf("Enter Array2 Elements : \n");
for (int i = 0; i < len2; i++)
{
   scanf("%d", &a2[i]);
}

int i = 0, j = 0, k = 0;

while (i < len1 && j < len2)
{
   if (a1[i] < a2[j])
      result[k++] = a1[i++];
   else
      result[k++] = a2[j++];
}

while (i < len1)
   result[k++] = a1[i++];

while (j < len2)
   result[k++] = a2[j++];
```

```
    printf("\nMerged Sorted Array : ");
    for (int i = 0; i < len1 + len2; i++)
    {
        printf("%d  ", result[i]);
    }
    return 0;
}
```

43. All Operations of String.

```
#include <stdio.h>
#include <string.h>
int main()
{
    char string1[20] = {'C', 'L', 'a', 'n', 'g', 'u', 'a', 'g', 'e'};
    printf("String : %s", string1);
    printf("Length of string is: %d\n", strlen(string1)); // 1

    char string2[20];

    strcpy(string2, string1); // 2
    printf("Value of second string is: %s\n", string2);

    char s1[10] = {'N', 'a', 'm', 'a', 's', 't', 'e', '\0'};
    char s2[10] = {'W', 'o', 'r', 'l', 'd', '\0'};

    if (strcmp(s1, s2) == 0) // 3
        printf("\nStrings are equal");
    else
        printf("\nStrings are not equal");

    strcat(s1, s2); // 4
    printf("\nValue of first string is: %s", s1);
```

```
    char newStr[20];
    printf("\nEnter String : ");
    gets(newStr);
    printf("Original String : %s\n", newStr);
    printf("Reverse String : %s\n", strrev(newStr)); // 5

    printf("\nUpper String is: %s", strupr(newStr));
    printf("\nLower String is: %s", strlwr(newStr));
    return 0;
}
```

44. C program to check whether a string is palindrome or not without Compare Function of String.

```
#include <stdio.h>
#include <string.h>
int main()
{
    char string[100];
    printf("Enter String : ");
    gets(string);
    int l = 0;
    int h = strlen(string) - 1;
    while (h > l)
    {
        if (string[l++] != string[h--])
        {
            printf("%s is not a palindrome\n", string);
            return 0;
        }
    }
    printf("%s is a palindrome\n", string);
    return 0;
}
```

45. C program to count frequency of each character in a string.

```
#include <stdio.h>
#include <string.h>
int main()
{
   char S[100];
   printf("Enter String : ");
   gets(S);
   int i = 0;
   int freq[26] = {0};
   while (S[i] != '\0')
   {
      freq[S[i] - 'a']++;
      i++;
   }
   for (int i = 0; i < 26; i++)
   {
      if (freq[i] != 0)
      {
         printf("%c - %d\n",
               i + 'a', freq[i]);
      }
   }
}
```

46. C program to find diameter, circumference and area of a circle using functions.

```
/* C-Program to Calculate Diameter, Circumference & Area of Circle */
#include <stdio.h>
float calcArea(float r)
{
   float Area = (22 * r * r) / 7;
   return Area;
}

float calcCircumference(float r)
{
   float Circum = (2 * 22 * r) / 7;
   return Circum;
}

float calcDiameter(float r)
{
   float Dia = 2 * r;
   return Dia;
}
int main()
{
   float r, d, C, Area; /* r = Radius, d = Diameter, C = Circumference */
   printf("Enter Radius of Circle (r) = ");
   scanf("%f", &r);

   printf("\nDiameter = %6.2f", calcDiameter(r));
   printf("\nCircumference = %6.2f", calcCircumference(r));
   printf("\nArea = %6.2f sq.cm", calcArea(r));
   return 0;
```

```
}
```

47. C program to check prime, armstrong and perfect numbers using functions.

```
#include <stdio.h>
#include <math.h>
void checkPrime(int n)
{
   int flag = 0;

   if (n < 2)
   {
      printf("\n%d is not a prime no.", n);
   }
   else if (n == 2)
   {
      printf("%d is a prime no. ", n);
   }
   else
   {
      if (n % 2 == 0)
      {
         printf("%d is not a prime no.", n);
      }
      else
      {
         for (int i = 3; i < n / 2; i += 2)
         {
            if (n % i == 0)
            {
               flag = 1;
               break;
            }
```

```
        }
        if (flag == 0)
            printf("%d is a prime no. ", n);
        else
            printf("%d is not a prime no.", n);
      }
   }
}

void checkArmstrong(int n)
{
   int n1, n2, rem, num = 0, length = 0;

   n1 = n;
   n2 = n;

   while (n2 > 0)
   {
      n2 = n2 / 10;
      length++;
   }

   while (n1 != 0)
   {
      rem = n1 % 10;
      num = num + pow(rem, length);
      n1 = n1 / 10;
   }

   if (num == n)
   {
      printf("\n%d is Armstrong Number.", n);
   }
   else
```

```
    {
        printf("\n%d is not a Armstrong Number.", n);
    }
}
void checkPerfect(int n)
{
    int x, i = 1, rem, sum = 0;
    x = n;
    while (i < n)
    {
        rem = x % i;
        if (rem == 0)
        {
            sum += i;
        }
        i++;
    }
    if (sum == n)
    {
        printf("\n%d is a perfect number.\n", n);
    }
    else
    {
        printf("\n%d isn't a perfect number.\n", n);
    }
}

int main()
{
    int num;
    printf("Enter number (num) : ");
    scanf("%d", &num);

    checkPrime(num);
```

```
    checkArmstrong(num);
    checkPerfect(num);

    return 0;
}
```

48. C program to add two number using pointers.

```
#include <stdio.h>

int main()
{
    int first, second, *p, *q, sum;

    printf("Enter two integers to add\n");
    scanf("%d%d", &first, &second);

    p = &first;
    q = &second;

    sum = *p + *q;

    printf("Sum of entered numbers = %d\n", sum);

    return 0;
}
```

49. Swap 2 numbers using Call by Value AND Call by reference.

```
#include <stdio.h>
void swapByVal(int, int);
void swapByRef(int *, int *);

int main()
{
    int a = 10;
    int b = 20;
```

```
    int c = 70;
    int d = 80;

    printf("Before swapping the values in main a = %d, b = %d\n", a, b); // printing the value of a and b in main
    swapByVal(a, b);
    printf("After swapping values in main a = %d, b = %d\n", a, b); // The value of actual parameters do not change by changing the formal parameters in call by value, a = 10, b = 20

    printf("Before swapping the values in main a = %d, b = %d\n", c, d); // printing the value of a and b in main
    swapByRef(&c, &d);
    printf("After swapping values in main a = %d, b = %d\n", c, d); // The values of actual parameters do change in call by reference, a = 10, b = 20
}

void swapByVal(int a, int b)
{
    int temp;
    temp = a;
    a = b;
    b = temp;
    printf("After swapping values in function a = %d, b = %d\n", a, b); // Formal parameters, a = 20, b = 10
}

void swapByRef(int *a, int *b)
{
    int temp;
    temp = *a;
    *a = *b;
    *b = temp;
    printf("After swapping values in function a = %d, b = %d\n", *a, *b); // Formal parameters, a = 20, b = 10
}
```

50. C program to copy an array to another array AND reverse an array using pointers.

```
#include <stdio.h>
void swap(int *a, int *b)
{
   int temp = *a;
   *a = *b;
   *b = temp;
}

void reverse(int arr[], int len)
{
   int *ptr1 = arr,
      *ptr2 = arr + len - 1;
   while (ptr1 < ptr2)
   {
      swap(ptr1, ptr2);
      ptr1++;
      ptr2--;
   }
}
void print(int *arr, int len)
{
   int *length = arr + len,
      *pos = arr;
   printf("arr = ");
   for (pos = arr; pos < length; pos++)
      printf("%d ", *pos); // cout << *pos << " ";
}
void copy(int arr[], int copyArr[], int len)
{
```

```
    for (int i = 0; i < len; i++)
    {
        copyArr[i] = arr[i];
    }
}
int main()
{
    int size;
    printf("Enter size of array : ");
    scanf("%d", &size);
    int arr[size];
    for (int i = 0; i < size; i++)
    {
        scanf("%d", &arr[i]);
    }

    printf("\nOriginal : ");
    print(arr, size);

    int copyArr[size];
    copy(arr, copyArr, size);
    printf("\nCopied Array : ");
    print(copyArr, size);

    printf("\nReverse : ");
    reverse(arr, size);
    print(arr, size);
    return 0;
}
```

Pattern Questions

1) Square Pattern

```
#include <stdio.h>
int main()
{
   int rows;
   printf("No. of rows (r) : ");
   scanf("%d", &rows);
   printf("Square Pattern\n");
   for (int i = 0; i < rows; i++)
   {
      for (int j = 0; j < rows; j++)
      {
         printf("* ");
      }
      printf("\n");
   }
   return 0;
}
```

2) Right Triangle Star Pattern

```
#include <stdio.h>
int main()
{
   int rows;
   printf("No. of rows (r) : ");
   scanf("%d", &rows);
   printf("Right Triangle Star Pattern\n");
   for (int i = 0; i < rows; i++)
   {
      for (int j = 0; j <= i; j++)
      {
         printf("* ");
      }
      printf("\n");
   }
   return 0;
}
```

3) Hollow Mirrored Right Triangle Star Pattern

```
#include <stdio.h>
int main()
{
   int rows;
```

```
    printf("No. of rows (r) : ");
    scanf("%d", &rows);
    printf("Hollow Mirrored Right Triangle Star Pattern\n");
    int space = rows;
    for (int i = 0; i < rows; i++)
    {
        for (int k = 0; k < space; k++)
        {
            printf("  ");
        }
        for (int j = 0; j < 2 * i + 1; j++)
        {
            printf("* ");
        }
        printf("\n");
        space--;
    }
    return 0;
}
```

4) Number Pattern 1

```
#include <stdio.h>
int main()
{
    int rows;
    printf("No. of rows (r) : ");
    scanf("%d", &rows);
    printf("Square Pattern\n");
    for (int i = 0; i < rows; i++)
    {
        for (int j = 0; j < rows; j++)
        {
            printf("1 ");
        }
        printf("\n");
    }
    return 0;
}
```

```
No. of rows (r) : 5
Square Pattern
1 1 1 1 1
1 1 1 1 1
1 1 1 1 1
1 1 1 1 1
1 1 1 1 1
```

5) Number Pattern 10

```
#include <stdio.h>
int main()
{
   int rows;
   printf("No. of rows (r) : ");
   scanf("%d", &rows);
   printf("Square Pattern\n");
   for (int i = 0; i < rows; i++)
   {
      for (int j = 0; j < rows; j++)
      {
         printf("%d ", i + 1);
      }
      printf("\n");
   }
   return 0;
}
```

```
No. of rows (r) : 5
Square Pattern
1 1 1 1 1
2 2 2 2 2
3 3 3 3 3
4 4 4 4 4
5 5 5 5 5
```