GLA UNIVERSITY MATHURA,UP

PRACTICAL FILE

SUBMITTED TO: GURPREET KAUR

SUBMITTED BY: DEVIKA AGARWAL

UNIVERSITY ROLLNO: 2215000590

SECTION: G1

CLASS ROLLNO: 24

SUBJECT: C PROGRAMMING LAB

BRANCH: COMPUTER SCIENCE AND ENGINEERING

CONTENTS

1. C program to perform all arithmetic operations

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

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

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

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

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

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

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

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

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

11. C program to calculate total electricity bill

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

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

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

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

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

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

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

19. C program to calculate factorial of a number.

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

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

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

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

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

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

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

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

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

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

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

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

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

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

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

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

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

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

38. C program to perform scalar matrix multiplication.

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

40. C program to check sparse AND transpose matrix.

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

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

43. All Operations of String.

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

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

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

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

48. C program to add two number using pointers.

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

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

1. C PROGRAM TO PERFORM ALL ARITHMETIC OPERATIONS.

```
#include <stdio.h>

Int main()

{

    int num1, num2;

    int sum, sub, mult, mod;

    float div;


    printf("Enter any two numbers: ");

    scanf("%d%d", &num1, &num2);


    sum = num1 + num2;

    sub = num1 - num2;

    mult = num1 * num2;

    div = (float) num1 / num2;

    mod = num1 % num2;


    printf("SUM = %d\n", sum);

    printf("DIFFERENCE = %d\n", sub);

    printf("PRODUCT = %d\n", mult);

    printf("QUOTIENT = %f\n", div);

    printf("MODULUS = %d", mod);


    return 0; }
```

2. C PROGRAM TO FIND AREA OF A TRIANGLE IF BASE AND HEIGHT ARE GIVEN.

```c
#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 = %.2f sq. units", area);
    return 0;
}
```

3. . C PROGRAM TO FIND ALL ANGLES OF A TRIANGLE IF TWO ANGLES ARE GIVEN.

```c
#include <stdio.h>
int main()
{
    int a, b, c;
    printf("Enter two angles of triangle: ");
    scanf("%d%d", &a, &b);
    c = 180 - (a + b);
    printf("Third angle of the triangle = %d", c);
    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 days: ");
    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", 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 = %.2lf", num, root);
    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;
    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;
    printf("Total marks = %.2f\n", total);
    printf("Average marks = %.2f\n", average);
    printf("Percentage = %.2f", percentage);
    return 0;
}
```

7. C PROGRAM TO CHECK LEAST SIGNIFICANT BIT (LSB) OF A NUMBER USING BITWISE OPERATOR.

```
#include <stdio.h>
int main()
{
    int num;
    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);
    return 0;
}
```

8. C PROGRAM TO SWAP TWO NUMBERS 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;
b=a-b;
a=a-b;
printf("\nAfter swap a=%d b=%d",a,b);
return 0;
}
```

9. C PROGRAM TO FIND MAXIMUM BETWEEN THREE NUMBERS USING CONDITIONAL OPERATOR OR TERNARY OPERATOR.

```
# include <stdio.h>
void main()
{
    int a, b, c, big ;
    printf("Enter three numbers : ");
    scanf("%d %d %d", &a, &b, &c);

    big = a > b ? (a > c ? a : c) : (b > c ? b : c);

    printf("\nThe biggest number is : %d", big) ;

}
```

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 USING SWITCH CASE.

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

  switch (op) {
    case '+':
      printf("%.1lf + %.1lf = %.1lf", first, second, first + second);
      break;
    case '-':
      printf("%.1lf - %.1lf = %.1lf", first, second, first - second);
      break;
    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;
}
```

13. C PROGRAM TO CHECK VOWEL OR CONSONANT USING SWITCH CASE.

```
#include <stdio.h>
int main()
{
    char ch;
    printf("Enter any alphabet: ");
    scanf("%c", &ch);
    switch(ch)
    {
        case 'a':
            printf("Vowel");
            break;
        case 'e':
            printf("Vowel");
            break;
        case 'i':
            printf("Vowel");
            break;
        case 'o':
            printf("Vowel");
            break;
        case 'u':
            printf("Vowel");
            break;
        case 'A':
            printf("Vowel");
            break;
        case 'E':
            printf("Vowel");
            break;
        case 'I':
            printf("Vowel");
            break;
        case 'O':
            printf("Vowel");
            break;
        case 'U':
            printf("Vowel");
            break;
        default:
            printf("Consonant");
    }
    return 0;}
```

14. C PROGRAM TO CHECK POSITIVE NEGATIVE OR ZERO USING SWITCH CASE.

```
#include <stdio.h>

int main()

{

    int num;

    printf("Enter any number: ");

    scanf("%d", &num);

    switch (num > 0)

    {

        case 1:

        printf("%d is positive.", num);

        break;

        case 0:

            switch (num < 0)

            {

                case 1:

                    printf("%d is negative.", num);

                    break;

                case 0:

                    printf("%d is zero.", num);

                    break;

            }

        break;

    }

    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 i, n, sum=0;
    printf("Enter upper limit: ");
    scanf("%d", &n);
    for(i=1; i<=n; i++)
    {
        sum += i;
    }
    printf("Sum of first %d natural numbers = %d", n, 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 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 i, num;
    printf("Enter number to print table: ");
    scanf("%d", &num);
    for(i=1; i<=10; i++)
    {
        printf("%d * %d = %d\n", num, i, (num*i));
    }
    return 0;
}
```

19. C PROGRAM TO CALCULATE FACTORIAL OF A NUMBER.

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

#define BASE 10

int main()

{

    long long num, n;

    int i, lastDigit;

    int freq[BASE];

    printf("Enter any number: ");

    scanf("%lld", &num);

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

    {

        freq[i] = 0;

    }

    n = num;

    while(n != 0)

    {

        lastDigit = n % 10;

        n /= 10;

        freq[lastDigit]++;

    }

    printf("Frequency of each digit in %lld is: \n", num);

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

    {

        printf("Frequency of %d = %d\n", i, freq[i]);

    }

    return 0; }
```

22. C PROGRAM TO FIND HCF(GCD) AND LCM OF TWO NUMBERS.

```
#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;
}
```

23. C PROGRAM TO PRINT ALL PRIME NUMBERS BETWEEN 1 TO N.

```
#include<stdio.h>
int main(){
    int num,i,count,n;
    printf("Enter max range: ");
    scanf("%d",&n);
    for(num = 1;num<=n;num++){
        count = 0;
        for(i=2;i<=num/2;i++){
            if(num%i==0){
                count++;
                break;
            }
        }
        if(count==0 && num!= 1)
        printf("%d ",num);
    }
    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, b, c, i, terms;

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

    a = 0;

    b = 1;

    c = 0;

    printf("Fibonacci terms: \n");

    for(i=1; i<=terms; i++)

    {

        printf("%d, ", c);

        a = b;

        b = c;

        c = a + b;

    }

    return 0;

}
```

26. C PROGRAM TO PRINT ARMSTRONG NUMBERS FROM 1 TO N.

```
#include <stdio.h>
#include <math.h>
int main()
{
    int num, lastDigit, digits, sum, i, end;
    printf("Enter upper limit: ");
    scanf("%d", &end);
    printf("Armstrong number between 1 to %d are: \n", end);
    for(i=1; i<=end; i++)
    {
        sum = 0;
        num = i;
        digits = (int) log10(num) + 1;
        while(num > 0)
        {
            lastDigit = num % 10;
            sum = sum + ceil(pow(lastDigit, digits));
            num = num / 10;
        }
        if(i == sum)
        {
            printf("%d, ", i);
        }
    }
    return 0;
}
```

27. C PROGRAM TO PRINT ALL PERFECT NUMBERS BETWEEN 1 TO N.

```
#include <stdio.h>
int main()
{
    int i, j, end, sum;
    printf("Enter upper limit: ");
    scanf("%d", &end);
    printf("All Perfect numbers between 1 to %d:\n", end);
    for(i=1; i<=end; i++)
    {
        sum = 0;
        for(j=1; j<i; j++)
        {
            if(i % j == 0)
            {
                sum += j;
            }
        }
        if(sum == i)
        {
            printf("%d, ", i);
        }
    }
    return 0;

}
```

28. C PROGRAM TO FIND POWER OF ANY NUMBER USING FOR LOOP.

```
#include <stdio.h>
int main()
{
int base, exponent;
long long power = 1;
int i;
printf("Enter base: ");
scanf("%d", &base);
printf("Enter exponent: ");
scanf("%d", &exponent);
for(i=1; i<=exponent; i++)
{
power = power * base;
}
printf("%d ^ %d = %lld", base, exponent, power);
return 0;
}
```

29. C PROGRAM TO PRINT ASCII VALUES OF ALL CHARACTERS.

```
#include <stdio.h>

int main() {

    char c;

    printf("Enter a character: ");

    scanf("%c", &c);

    printf("ASCII value of %c = %d", c, c);

    return 0;

}
```

30. C PROGRAM TO PRINT PASCAL TRIANGLE UP TO N ROWS.

```
#include <stdio.h>

int getFactorial(int n);

int main(){
    int row, rows, i, value;
    printf("Enter Number of Rows of Pascal Triangle\n");
    scanf("%d", &rows);
   for(row = 0; row < rows; row++){
        for(i = row; i <= rows; i++)
        printf("  ");
        for(i = 0; i <= row; i++){
            value = getFactorial(row)/(getFactorial(i)*getFactorial(row-i));
            printf("%4d", value);
        }
        printf("\n");
    }
    return 0;
}
int getFactorial(int N){
    if(N < 0){
        printf("Invalid Input: factorial not defined for negative numbers\n");
        return 0;
    }
    int nFactorial = 1, counter;
    for(counter = 1; counter <= N; counter++){
        nFactorial = nFactorial * counter;
    }
    return nFactorial;
}
```

31. C PROGRAM TO FIND SUM OF ALL ELEMENTS OF ARRAY.

```
#include<stdio.h>
int main()
{
int arr[100], size, i, sum = 0;
printf("Enter array size\n");
scanf("%d",&size);
printf("Enter array elements\n");
for(i = 0; i < size; i++)
scanf("%d",&arr[i]);
for(i = 0; i < size; i++)
sum = sum + arr[i];
printf("Sum of the array = %d\n",sum);
return 0;
}
```

32. C PROGRAM TO COPY ONE ARRAY TO ANOTHER ARRAY.

```
#include <stdio.h>
int main()
{
    int arr1[] = {1, 2, 3, 4, 5};
    int length = sizeof(arr1)/sizeof(arr1[0]);
    int arr2[length];
    for (int i = 0; i < length; i++) {
        arr2[i] = arr1[i];
    }
    printf("Elements of original array: \n");
    for (int i = 0; i < length; i++) {
        printf("%d ", arr1[i]);
    }
    printf("\n");
    printf("Elements of new array: \n");
    for (int i = 0; i < length; i++) {
        printf("%d ", arr2[i]);
    }
    return 0;   }
```

33. C PROGRAM TO INSERT AN ELEMENT IN ARRAY AT SPECIFIED POSITION.

```
    # include < stdio.h >

    int  main( )

    {
int  a[20], i, n, ele, pos ;
printf(" Enter the Numbers of elements: ") ;
scanf("%d ",& n) ;
printf("\n Enter the elements of array : \n") ;
for (  i = 1 ; i < = n ; i++)
scanf("%d ",& a[i]) ;
printf("\n Array enter by user are :\n") ;
for (  i = 1 ; i < = n ; i++)
printf("%d \t",a[i]) ;

printf("\n Enter the position you want to enter :") ;
scanf("%d ",& pos) ;
printf("\n Enter the element you want to enter :") ;
scanf("%d ",& ele) ;

for (  i = 1 ; i < = n ; i++)
{
if ( i > pos )
a[i] = a[i-1] ;
else
{
if ( i == pos )
a[i] = ele ;
else
a[i] = a[i] ;
}
}
printf("\n Array After Inserting element :\n") ;
for (  i = 1 ; i < = n ; i++)
pcanf("%d \t",a[i]) ;
return ( 0 ) ;
}
```

34. C PROGRAM TO DELETE AN ELEMENT IN ARRAY AT SPECIFIED POSITION.

```
#include <stdio.h>

#define MAX_SIZE 100

int main()
```

```
{
    int arr[MAX_SIZE];
    int i, size, pos;
    printf("Enter size of the array : ");
    scanf("%d", &size);
    printf("Enter elements in array : ");
    for(i=0; i<size; i++)
    {
        scanf("%d", &arr[i]);
    }
    printf("Enter the element position to delete : ");
    scanf("%d", &pos);
    if(pos < 0 || pos > size)
    {
        printf("Invalid position! Please enter position between 1 to %d", size);
    }
    else
    {
        for(i=pos-1; i<size-1; i++)
        {
            arr[i] = arr[i + 1];
        }
        size--;
        printf("\nElements of array after delete are : ");
        for(i=0; i<size; i++)
        {
            printf("%d\t", arr[i]);
        }
    }
    return 0;
}
```

35. C Program To Search Element In Array Using Linear Search.

```
#include <stdio.h>
#include <conio.h>
int main(){
    int inputArray[100], elementCount, counter, num;
    printf("Enter Number of Elements in Array\n");
    scanf("%d", &elementCount);
    printf("Enter %d numbers \n", elementCount);
    for(counter = 0; counter < elementCount; counter++){
        scanf("%d", &inputArray[counter]);
    }
    printf("Enter a number to serach in Array\n");
    scanf("%d", &num);
    for(counter = 0; counter < elementCount; counter++){
        if(inputArray[counter] == num){
            printf("Number %d found at index %d\n", num, counter);
            break;
        }
    }
    if(counter == elementCount){
     printf("Number %d Not Present in Input Array\n", num);
    }
    getch();
    return 0;
}
```

36. C PROGRAM TO CHECK MSB OF A NUMBER USING BITWISE OPERATOR.

```
#include <stdio.h>
#define BITS sizeof(int) * 8
int main()
{
    int num, msb;
    printf("Enter any number: ");
    scanf("%d", &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;
}
```

37. C PROGRAM TO COUNT TOTAL NUMBER OF DUPLICATE ELEMENTS IN AN ARRAY.

```
#include <stdio.h>
#define MAX_SIZE 100
```

```
int main()
{
    int arr[MAX_SIZE];
    int i, j, size, count = 0;
    printf("Enter size of the array : ");
    scanf("%d", &size);
    printf("Enter elements in array : ");
    for(i=0; i<size; i++)
    {
        scanf("%d", &arr[i]);
    }
    for(i=0; i<size; i++)
    {
        for(j=i+1; j<size; j++)
        {
            if(arr[i] == arr[j])
            {
                count++;
                break;
            }
        }
    }
    printf("\nTotal number of duplicate elements found in array = %d", count);
    return 0;
}
```

38. C PROGRAM TO PERFORM SCALAR MATRIX MULTIPLICATION.

```c
#include <stdio.h>
#define SIZE 3
int main()
{
    int A[SIZE][SIZE];
    int num, row, col;
    printf("Enter elements in matrix of size %dx%d: \n", SIZE, SIZE);
    for(row=0; row<SIZE; row++)
    {
        for(col=0; col<SIZE; col++)
        {
            scanf("%d", &A[row][col]);
        }
    }
    printf("Enter any number to multiply with matrix A: ");
    scanf("%d", &num);
    for(row=0; row<SIZE; row++)
    {
        for(col=0; col<SIZE; col++)
        {
            A[row][col] = num * A[row][col];
        }
    }
    printf("\nResultant matrix c.A = \n");
    for(row=0; row<SIZE; row++)
    {
        for(col=0; col<SIZE; col++)
        {
            printf("%d ", A[row][col]);
        }
        printf("\n");
    }
    return 0;
}
```

39. C PROGRAM TO FIND SUM OF MAIN DIAGONAL ELEMENTS OF A MATRIX.

```
#include <stdio.h>
#define SIZE 3 // Matrix size
int main()
{
    int A[SIZE][SIZE];
    int row, col, sum = 0;
    printf("Enter elements in matrix of size %dx%d: \n", SIZE, SIZE);
    for(row=0; row<SIZE; row++)
    {
        for(col=0; col<SIZE; col++)
        {
            scanf("%d", &A[row][col]);
        }
    }
    for(row=0; row<SIZE; row++)
    {
        sum = sum + A[row][row];
    }
    printf("\nSum of main diagonal elements = %d", sum);
    return 0;
}
```

40. C PROGRAM TO CHECK A MATRIX IS SPARSE MATRIX OR NOT.

```
#include <stdio.h>
#include <conio.h>
int main(){
    int rows, cols, row, col, count=0;
    int matrix[50][50];
    printf("Enter Rows and Columns of Matrix\n");
    scanf("%d %d", &rows, &cols);
    printf("Enter Matrix of size %dX%d\n", rows, cols);
    for(row = 0; row < rows; row++){
        for(col = 0; col < cols; col++){
            scanf("%d", &matrix[row][col]);
        }
    }
    for(row = 0; row < rows; row++){
        for(col = 0; col < cols; col++){
            if(matrix[row][col] == 0){
             count++;
            }
        }
    }
    if(count > (rows*cols)/2){
        printf("Input Matrix is a Sparse Matrix\n");
    } else {
        printf("Input Matrix is Not a Sparse Matrix\n");
    }
    getch();
    return 0;
}
```

41. C PROGRAM TO CHECK WHETHER A MATRIX IS IDENTITY MATRIX OR NOT.

```
#include<stdio.h>
int main()
{
        int i, j, rows, columns, a[10][10],Flag = 1;
        printf("\n Please Enter Number of rows and columns  :  ");
        scanf("%d %d", &i, &j);
    printf("\n Please Enter the Matrix Elements \n");
        for(rows = 0; rows < i; rows++)
        {
                for(columns = 0; columns < j; columns++)
        {
                scanf("%d", &a[rows][columns]);
        }
        }
    for(rows = 0; rows < i; rows++)
        {
```

```
        for(columns = 0; columns < j; columns++)
    {
        if(a[rows][columns] != 1 && a[columns][rows] != 0)
        {
            Flag = 0;
            break;
            }
        }
    }
    if(Flag == 1)
    {
        printf("\n The Matrix that you entered is an Identity Matrix ");
    }
    else
    {
        printf("\n The Matrix that you entered is Not an Identity Matrix ");
    }
    return 0;
}
```

42. C PROGRAM TO MERGE TWO SORTED ARRAY IN ASCENDING ORDER.

```
#include <stdio.h>
int main(){
  int s1, s2, s3;
  printf("\n Enter the size of 1st array  ");
  scanf("%d", & s1);
  printf("\n Enter the size of 2nd array ");
```

```
scanf("%d", & s2);
s3 = s1 + s2;
printf("\n Enter the elements of 1st array\n");
int arr1[s1], arr2[s2], arr3[s3];
for (int i = 0; i < s1; i++) {
  scanf("%d", & arr1[i]);
  arr3[i] = arr1[i];
}
int k = s1;
printf("\nEnter the elements of 2nd array \n");
for (int i = 0; i < s2; i++)
{
  scanf("%d", & arr2[i]);
  arr3[k] = arr3[i];
  k++;
}
printf("\nThe merged array before sorting : \n\t");
for (int i = 0; i < s3; i++)
  printf("%d ", arr3[i]);
  printf("\n The merged array after sorting\n\t");
for (int i = 0; i < s3; i++){
    int tem;
    for (int j = i + 1; j < s3; j++) {
    if (arr3[i] > arr3[j]) {
      tem = arr3[i];
      arr3[i] = arr3[j];
```

```
        arr3[j] = tem;
      }
    }
  }
  for (int i = 0; i < s3; i++)
  {
    printf(" %d ", arr3[i]);
  }
}
```

43. C PROGRAM TO SWAP TWO NUMBERS USING 3RD VARIABLE.

```
#include<stdio.h>
int main() {
  double first, second, temp;
  printf("Enter first number: ");
  scanf("%lf", &first);
  printf("Enter second number: ");
  scanf("%lf", &second);
  temp = first;
  first = second;
  second = temp;
  printf("\nAfter swapping, first number = %.2lf\n", first);
  printf("After swapping, second number = %.2lf", second);
  return 0;
}
```

44. C PROGRAM TO CHECK WHETHER A STRING IS PALINDROME OR NOT WITHOUT COMPARE FUNCTION OF STRING

```
#include<stdio.h>
int main()
{
    char string[40];
    int length=0, flag=1,i;
    printf("Enter string:\n");
    gets(string);
    for(i=0;string[i]!='\0';i++)
    {
        length++;
    }
    for(i=0;i< length/2;i++)
    {
        if( string[i] != string[length-1-i] )
        {
            flag=0;
            break;
        }
    }
    if(flag==1)
    {
        printf("PALINDROME");
    }
    else
    {
        printf("NOT PALINDROME");
    }
    return 0;
}
```

45. C PROGRAM TO COUNT FREQUENCY OF EACH CHARACTER IN A STRING.

```
#include <stdio.h>
#include <string.h>
int main()
{
    char s[1000];
    int  i,j,k,count=0,n;
    printf("Enter  the string : ");
    gets(s);
    for(j=0;s[j];j++);
    n=j;
            printf(" frequency count character in string:\n");
            for(i=0;i<n;i++)
    {
        count=1;
            if(s[i])
            {
                for(j=i+1;j<n;j++)
                  {
                        if(s[i]==s[j])
                {
                count++;
                s[j]='\0';
                        }
                  }
                  printf(" '%c' = %d \n",s[i],count);
            }
            }
            return 0;
}
```

46. C PROGRAM TO FIND DIAMETER, CIRCUMFERENCE AND AREA OF A CIRCLE USING FUNCTIONS.

```
#include <stdio.h>
#include <math.h>
double getDiameter(double radius);
double getCircumference(double radius);
double getArea(double radius);
int main()
{
    float radius, dia, circ, area;
    printf("Enter radius of circle: ");
    scanf("%f", &radius);
    dia  = getDiameter(radius);
    circ = getCircumference(radius);
    area = getArea(radius);
    printf("Diameter of the circle = %.2f units\n", dia);

    printf("Circumference of the circle = %.2f units\n", circ);

    printf("Area of the circle = %.2f sq. units", area);

    return 0;

}

double getDiameter(double radius)

{

    return (2 * radius);

}

double getCircumference(double radius)

{

    return (2 * M_PI * radius);

}

double getArea(double radius)

{

    return (M_PI * radius * radius);

}
```

47. C PROGRAM TO CHECK PRIME, ARMSTRONG AND PERFECT NUMBERS USING FUNCTIONS.

```
#include <stdio.h>

#include <math.h>

int isPrime(int num);

int isArmstrong(int num);

int isPerfect(int num);

int main()

{
```

```
int num;

printf("Enter any number: ");

scanf("%d", &num);

if(isPrime(num))

{

    printf("%d is Prime number.\n", num);

}

else

{

    printf("%d is not Prime number.\n", num);

}

if(isArmstrong(num))

{

    printf("%d is Armstrong number.\n", num);

}

else

{

    printf("%d is not Armstrong number.\n", num);

}

if(isPerfect(num))

{

    printf("%d is Perfect number.\n", num);

}

else

{

    printf("%d is not Perfect number.\n", num);
    }
    return 0;
}
int isPrime(int num)
{
    int i;
    for(i=2; i<=num/2; i++)
    {
        if(num%i == 0)
        {
            return 0;
        }
    }
    return 1;
}
int isArmstrong(int num)
{
    int lastDigit, sum, originalNum, digits;
    sum = 0;
    originalNum = num;
    digits = (int) log10(num) + 1;
    while(num > 0)
    {
        lastDigit = num % 10;
        sum = sum + round(pow(lastDigit, digits));
        num = num / 10;
    }
    return (originalNum == sum);
}
int isPerfect(int num)
{
    int i, sum, n;
    sum = 0;
    n = num;
    for(i=1; i<n; i++)
    {
        if(n%i == 0)
        {
            sum += i;
        }
    }
    return (num == sum);
}
```

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 the numbers = %d\n", sum);

    return 0;
}
```

49. SWAP 2 NUMBERS USING CALL BY VALUE .

```
#include <stdio.h>
void swap(int , int);
int main()
{
    int a = 10;
    int b = 20;
    printf("Before swapping the values in main a = %d, b = %d\n",a,b);
    swap(a,b);
    printf("After swapping values in main a = %d, b = %d\n",a,b);
}
void swap (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);
}
```

50. C PROGRAM TO COPY AN ARRAY TO ANOTHER ARRAY AND REVERSE AN ARRAY USING POINTERS.

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

```
#define MAX_SIZE 100

void printArr(int *arr, int size);

int main()

{

    int arr[MAX_SIZE];

    int size;

    int *left = arr;

    int *right;

    printf("Enter size of array: ");

    scanf("%d", &size);

    right = &arr[size - 1];

    printf("Enter elements in array: ");

    while(left <= right)

    {

        scanf("%d", left++);

    }

    printf("\nArray before reverse: ");

    printArr(arr, size);

    left = arr;

    while(left < right)

    {

        *left   ^= *right;

        *right  ^= *left;

        *left   ^= *right;

        left++;

        right--;
    }
    printf("\nArray after reverse: ");
    printArr(arr, size);
    return 0;
}
void printArr(int * arr, int size)
{
    int * arrEnd = (arr + size - 1);
    while(arr <= arrEnd)
    {
        printf("%d, ", *arr);
        arr++;
    }
}
```

51. WAP IN C TO PRINT THE FOLLOWING PATTERN.

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

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

52. WAP IN C TO PRINT THE FOLLOWING PATTERN.

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

```
#include <stdio.h>
int main()
{
    int i, j, N;
    printf("Enter number of rows: ");
    scanf("%d", &N);
    for(i=1; i<=N; i++)
    {
        for(j=1; j<=N; j++)
        {
            if(i==1 || i==N || j==1 || j==N)
            {
                printf("*");
            }
            else
            {
                printf(" ");
            }
        }
        printf("\n");
    }
    return 0;
}
```

53. WAP IN C TO PRINT THE FOLLOWING PATTERN.

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

```
#include <stdio.h>
int main()
{
    int i, j, N;
    printf("Enter number of rows: ");
    scanf("%d", &N);
    for(i=1; i<=N; i++)
    {
        for(j=1; j<=N; j++)
        {
            if(i==1 || i==N || j==1 || j==N || i==j || j==(N - i + 1))
            {
                printf("*");
            }
            else
            {
                printf(" ");
            }
        }
        printf("\n");
    }
    return 0;
}
```

54. WAP IN C TO PRINT THE FOLLOWING PATTERN:

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

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

55. WAP IN C TO PRINT THE FOLLOWING PATTERN:

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

```
#include <stdio.h>

int main()

{

    int i, j, rows;

    printf("Enter rows : ");

    scanf("%d", &rows);

    for(i=1; i<=rows; i++)

    {

        for(j=1; j<=rows-i; j++)

        {

            printf(" ");

        }

        for(j=1; j<=rows; j++)

        {
            if(i==1 || i==rows || j==1 || j==rows)
                printf("*");
            else
                printf(" ");
        }
        printf("\n");
    }
    return 0;
}
```