1)Matrix Multiplication

```
#include<stdio.h>
#include<stdlib.h>
void main()
{
	int a[10][10],b[10][10],mul[10][10],r,c,i,j,k;
	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++)
		{
			printf("Enter the element a%d%d:", i+1, j+1);
			scanf("%d",&a[i][j]);
		}
	}
	printf("enter the second matrix element=\n");
	for(i=0;i<r;i++)
	{
		for(j=0;j<c;j++)
		{
			printf("Enter the element b%d%d:", i+1, j+1);
			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(i=0;i<r;i++)
	{
		for(j=0;j<c;j++)
```

```
                {
                        printf("%d\t",mul[i][j]);
                }
                printf("\n");
        }
}
```

Output

```
enter the number of row=2
enter the number of column=2
enter the first matrix element=
Enter the element a11:2
Enter the element a12:5
Enter the element a21:3
Enter the element a22:6
enter the second matrix element=
Enter the element b11:4
Enter the element b12:5
Enter the element b21:6
Enter the element b22:8
multiply of the matrix=
38      50
48      63
```

2)Sum of main and off diagonal of a square matrix

```
#include<stdio.h>
#include<stdlib.h>
void main()
{
        int a[10][10], sm, so, r,i,j,k;
        printf("enter the number of rows of the square matrix: ");
        scanf("%d",&r);
        printf("enter the elements of matrix :\n");
        for(i=0;i<r;i++)
        {
                for(j=0;j<r;j++)
                {
                        printf("Enter the element a%d%d:", i+1, j+1);
                        scanf("%d",&a[i][j]);
                }
        }
```

```
        printf("Elements of matrix:\n");
        for(i=0;i<r;i++)
        {
                for(j=0;j<r;j++)
                {
                        printf("%d ", a[i][j]);

                }
                printf("\n");
        }
        printf("Elements of Main diagonal are:\n");
        for(i=0;i<r;i++)
        {
                for(j=0;j<r;j++)
                {
                        if(i==j)
                        {
                                printf("%d\n", a[i][j]);
                        }
                }
        }
        printf("Elements of Off diagonal are:\n");
        for(i=0;i<r;i++)
        {
                for(j=0;j<r;j++)
                {
                        if(i==j)
                        {
                                printf("%d\n", a[i][r-j-1]);
                        }
                }
        }
        for(i=0;i<r;i++)
        {
                sm+=a[i][i];
                so+=a[i][r-i-1];
        }
        printf("\nThe sum of the main diagonal elements is = %d\n", sm);
   printf("The sum of the off diagonal elements is   = %d\n", so);
}
```

Output

```
enter the number of rows of the square matrix: 3
```

```
enter the elements of matrix :
Enter the element a11:2
Enter the element a12:3
Enter the element a13:1
Enter the element a21:4
Enter the element a22:5
Enter the element a23:6
Enter the element a31:7
Enter the element a32:8
Enter the element a33:9
Elements of matrix:
2 3 1
4 5 6
7 8 9
Elements of Main diagonal are:
2
5
9
Elements of Off diagonal are:
1
5
7

The sum of the main diagonal elements is = 16
The sum of the off diagonal elements is   = 13
```

3)Average and sum of array

```
#include <stdio.h>
int main()
{
  int marks[10], i, n, sum = 0;
  double average;
  printf("Enter number of elements: ");
  scanf("%d", &n);
  for(i=0; i < n; ++i)
  {
    printf("Enter number%d: ",i+1);
    scanf("%d", &marks[i]);
    sum += marks[i];
  }
  average = (double) sum / n;
  printf("Sum = %.2lf", (double) sum);
```

```
  printf("Average = %.2lf", average);
  return 0;
}
```

Output

```
Enter number of elements: 5
Enter number1: 5
Enter number2: 4
Enter number3: 6
Enter number4: 8
Enter number5: 2
Sum = 25.00Average = 5.00
```

4) Reverse a number using function

```
#include<stdio.h>
#include<conio.h>
int Reverse(int n);
void main()
{
   int rev, num;
   printf("Enter a Positive Number: ");
   scanf("%d", &num);
   rev = Reverse(num);
   printf("The Reverse of given number %d is: %d", num, rev);

}
int Reverse(int n)
{
   int sum=0;
   while (n!=0)
   {
     sum = sum*10 + n%10;
     n /= 10;
   }
   return sum;
}
```

Output

```
Enter a Positive Number: 2526
The Reverse of given number 2526 is: 6252
```

5)Reverse a string using function

```
#include <stdio.h>
void reverseSentence();
int main()
{
   printf("Enter a sentence: ");
   reverseSentence();
   return 0;
}
void reverseSentence()
{
   char c;
   scanf("%c", &c);
   if (c != '\n')
         {
      reverseSentence();
      printf("%c", c);
   }
}
```

Output

```
Enter a sentence: Locust
tsucoL
```

6)Reverse sentence

```
#include <stdio.h>
void reverseSentence();
int main()
{
   printf("Enter a sentence: ");
   reverseSentence();
   return 0;
}
void reverseSentence()
{
   char c;
   scanf("%c", &c);
   if (c != '\n')
         {
      reverseSentence();
      printf("%c", c);
```

```
    }
}
```

Output

```
Enter a sentence: saveetha
ahteevas
```

7) Swap number using call by reference

```
#include <stdio.h>
void cyclicSwap(int *a, int *b, int *c);
int main()
{
    int a, b, c;
    printf("Enter a, b and c respectively: ");
    scanf("%d %d %d", &a, &b, &c);
    printf("Value before swapping:\n");
    printf("a = %d \nb = %d \nc = %d\n", a, b, c);
    cyclicSwap(&a, &b, &c);
    printf("Value after swapping:\n");
    printf("a = %d \nb = %d \nc = %d", a, b, c);
    return 0;
}

void cyclicSwap(int *n1, int *n2, int *n3)
{
    int temp;
    temp = *n2;
    *n2 = *n1;
    *n1 = *n3;
    *n3 = temp;
}
```

Output

```
Enter a, b and c respectively: 5 6 7
Value before swapping:
a = 5
b = 6
c = 7
Value after swapping:
a = 7
b = 5
```

c = 6

8) Reverse an array

```
void main()
{
  int values[50], n, i;
  printf("Enter number of elements: ");
  scanf("%d", &n);
  printf("Enter %d integers:\n", n);
  for( i = 0; i < n; ++i)
  {
          printf("Enter element %d:", i+1);
     scanf("%d", &values[i]);
  }
  printf("Displaying reverse integers:\n");

  for(i = n-1; i >=0; --i)
  {
    printf("%d ", values[i]);
  }
}
```

Output

```
Enter number of elements: 5
Enter 5 integers:
Enter element 1:4
Enter element 2:5
Enter element 3:6
Enter element 4:1
Enter element 5:2
Displaying reverse integers:
2 1 6 5 4
```

9)Floyd triangle

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

```
  {
    for (j = 1; j <= i; ++j)
          {
      printf("%d ", number);
      ++number;
    }
    printf("\n");
  }
  return 0;
}
```

Output

```
Enter the number of rows: 5
1
2 3
4 5 6
7 8 9 10
11 12 13 14 15
```

10)Length of sentence using pointer

```
#include <stdio.h>

int main()
{
   char str[100];
   char *ptr;
   int len = 0;

   printf("Enter a string: ");
   scanf("%s", str);

   ptr = str;
   while (*ptr != '\0')
   {
      len++;
      ptr++;
   }

   printf("Length of the string is: %d", len);

   return 0;
}
```

Output

```
Enter a string: Pokemon
Length of the string is: 7
```

11)Factorial using pointer

```
#include <stdio.h>
void factorial(int num, long long *result);
int main()
{
    int num;
    long long result = 1;
    printf("Enter a number: ");
    scanf("%d", &num);
    factorial(num, &result);
    printf("Factorial of %d is %lld\n", num, result);
    return 0;
}

void factorial(int num, long long *result)
{
    int i;
    for (i = 1; i <= num; ++i)
            {
        *result *= i;
    }
}
```

Output

```
Enter a number: 5
Factorial of 5 is 120
```

12)Largest and smallest number using function

```
#include <stdio.h>
void findMinMax(int arr[], int size, int *min, int *max);
int main()
{
    int arr[100], size, i, min, max;
    printf("Enter the number of elements: ");
    scanf("%d", &size);
```

```
    printf("Enter the elements:\n");
    for (i = 0; i < size; ++i)
            {
        scanf("%d", &arr[i]);
    }
    findMinMax(arr, size, &min, &max);
    printf("Minimum element: %d\n", min);
    printf("Maximum element: %d\n", max);
    return 0;
}

void findMinMax(int arr[], int size, int *min, int *max)
{
    int i;
    *min = arr[0];
    *max = arr[0];
    for (i = 1; i < size; ++i)
            {
        if (arr[i] < *min)
                    {
            *min = arr[i];
        }
        if (arr[i] > *max)
                    {
            *max = arr[i];
        }
    }
}
```

Output

```
Enter the number of elements: 5
Enter the elements:
5 6 4 3 2
Minimum element: 2
Maximum element: 6
```

13)Composite numbers a array

```
#include <stdio.h>
int is_composite(int n)
{
    if (n < 2)
            {
```

```
        return 0;
    }
    for (int i = 2; i * i <= n; i++)
            {
        if (n % i == 0)
                    {
            return 1;
        }
    }
    return 0;
}

int main()
{
    int n;
    printf("Enter the size of the array: ");
    scanf("%d", &n);

    int arr[n];
    printf("Enter the elements of the array:\n");
    for (int i = 0; i < n; i++)
            {
        scanf("%d", &arr[i]);
    }

    printf("The composite numbers in the array are: ");
    for (int i = 0; i < n; i++)
            {
        if (is_composite(arr[i]))
                    {
            printf("%d ", arr[i]);
        }
    }
    printf("\n");
    return 0;
}
```

Output

```
Enter the size of the array: 5
Enter the elements of the array:
5 6 4 9 8
The composite numbers in the array are: 6 4 9 8
```

14)Compile and Execute the C program to calculate Arithmetic Operators Functions such as Pow(x,n), Add(x,n), Sub(x,n), Mul(x,n), Div(x,n), where x and n are the two operands. Get the input and choice from the user.

```
#include <stdio.h>
#include <math.h>
int main()
{
   int choice;
   float x, n, result;
   printf("Enter value of number 1: ");
   scanf("%f", &x);
   printf("Enter value of number 2: ");
   scanf("%f", &n);
   printf("\nChoose the arithmetic operation:\n");
   printf("1. Power\n2. Addition\n3. Subtraction\n4. Multiplication\n5. Division\n");
   printf("Enter choice (1-5): ");
   scanf("%d", &choice);
   switch(choice)
         {
      case 1:
         result = pow(x, n);
         printf("%.2f ^ %.2f = %.2f", x, n, result);
         break;
      case 2:
         result = x + n;
         printf("%.2f + %.2f = %.2f", x, n, result);
         break;
      case 3:
         result = x - n;
         printf("%.2f - %.2f = %.2f", x, n, result);
         break;
      case 4:
         result = x * n;
         printf("%.2f * %.2f = %.2f", x, n, result);
         break;
      case 5:
         result = x / n;
         printf("%.2f / %.2f = %.2f", x, n, result);
         break;
      default:
         printf("Invalid choice");
   }
```

```
    return 0;
}
```

Output

```
Enter value of number 1: 5
Enter value of number 2: 10

Choose the arithmetic operation:
1. Power
2. Addition
3. Subtraction
4. Multiplication
5. Division
Enter choice (1-5): 2
5.00 + 10.00 = 15.00
```

15) Number Pattern

```
1
2 2
3 3 3
```

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

Output

```
Enter the number of rows: 5
1
2 2
```

3 3 3
4 4 4 4
5 5 5 5 5

16)Negative numbers in an array of numbers

```
#include <stdio.h>
int main()
{
    int arr[100], n;
    int count = 0;
    printf("Enter the number of elements: ");
    scanf("%d", &n);
    for (int i = 0; i < n; i++)
            {
        printf("Enter the %d element: ", i+1);
        scanf("%d", &arr[i]);
    }
    for (int i = 0; i < n; i++)
            {
        if (arr[i] < 0)
                    {
            count++;
        }
    }
    printf("The number of negative numbers in the array is %d\n", count);
    return 0;
}
```

Output

```
Enter the number of elements: 5
Enter the 1 element: 2
Enter the 2 element: -3
Enter the 3 element: 4
Enter the 4 element: -5
Enter the 5 element: 6
The number of negative numbers in the array is 2
```

17)Write a program in C to add two numbers using pointers

```
#include <stdio.h>

void add(int *a, int *b, int *result)
```

```
{
   *result = *a + *b;
}
int main()
{
   int num1, num2, sum;
   int *p1, *p2, *p3;
   printf("Enter first number: ");
   scanf("%d", &num1);
   printf("Enter second number: ");
   scanf("%d", &num2);
   p1 = &num1;
   p2 = &num2;
   p3 = &sum;
   add(p1, p2, p3);
   printf("The sum of %d and %d is %d\n", num1, num2, sum);
   return 0;
}
```

Output

```
Enter first number: 5
Enter second number: 5
The sum of 5 and 5 is 10
```

18)Leap year using date format input

```
#include <stdio.h>
int main()
{
   int date, month, year;
   printf("Enter date in DD/MM/YYYY format: ");
   scanf("%d/%d/%d", &date, &month, &year);
   if (year % 4 == 0)
         {
      printf("%d/%d/%d is a leap year.", date, month, year);
   }
         else
         {
      printf("%d/%d/%d is not a leap year.", date, month, year);
   }
   return 0;
}
```

Output

```
Enter date in DD/MM/YYYY format: 22/11/2004
22/11/2004 is a leap year.
```

19)Star Pattern

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

Output

```
Enter the number of rows: 4
*
* *
* * *
* * * *
```

20)Leap Year

```
#include <stdio.h>
int main()
{
   int age;
   printf("Enter your age: ");
   scanf("%d", &age);
   if (age >= 18)
         {
     printf("You are eligible to vote!\n");
   }
```

```
        else
        {
      int years_left = 18 - age;
      printf("You are not eligible to vote. You need to wait %d more year(s).\n", years_left);
   }
   return 0;
}
```

Output

```
Enter your age: 7
You are not eligible to vote. You need to wait 11 more year(s)
```

21)Compile and Execute the C program to check whether the number divisible by 2, then print the given number even otherwise odd. Check and display the output on the screen.

```
#include <stdio.h>

int main()
{
   int num;
   printf("Enter a number: ");
   scanf("%d", &num);
   if (num % 2 == 0)
        {
      printf("%d is even.\n", num);
   }
        else
        {
      printf("%d is odd.\n", num);
   }
   return 0;
}
```

Output

```
Enter a number: 2545
2545 is odd.
```

22)Write a C program to print the number of vowels in the given statement

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

```
#include <ctype.h>

int main()
{
    char str[100];
    int vowels = 0;
    printf("Enter a string: ");
    fgets(str, sizeof(str), stdin);

    for (int i = 0; str[i] != '\0'; i++)
          {
        char c = str[i];

        if (c == 'a' || c == 'e' || c == 'i' || c == 'o' || c == 'u' || c == 'A'|| c == 'E'|| c == 'I'|| c == 'O'|| c == 'U')
                    {
            vowels++;
        }
    }

    printf("Number of vowels: %d\n", vowels);

    return 0;
}
```

Output

```
Enter a string: Saveetha School of Engineering
Number of vowels: 12
```

23)Write a program using function to calculate the simple interest. Suppose the customer is a senior citizen. He is being offered 12 percent rate of interest; for all other customers, the ROI is 10 percent.

```
#include <stdio.h>
float calculateSI(float principal, float time, char customerType);
int main()
{
    float principal, rate, time, si;
    char customerType;
    printf("Enter the principal amount: ");
    scanf("%f", &principal);
    printf("Enter the time period in years: ");
```

```
    scanf("%f", &time);
    printf("Is customer senior citizen (y/n): ");
    scanf(" %c", &customerType);
    si = calculateSI(principal, time, customerType);
    printf("The simple interest is: %.2f\n", si);
    return 0;
}

float calculateSI(float principal, float time, char customerType)
{
    float rate;
    if (customerType == 'Y'|| customerType=='y')
            {
        rate = 12.0;
    }
            else
            {
        rate = 10.0;
    }
    return (principal * rate * time) / 100.0;
}
```

Output

```
Enter the principal amount: 25000
Enter the time period in years: 5
Is customer senior citizen (y/n): Y
The simple interest is: 15000.00
```

24)Compile and Execute the C program to change all the digits of a number to bring the digit at the last position to the first position and vice-versa using loop. Get the input from user

```
#include <stdio.h>
int main()
{
    int num, reverse_num = 0;
    printf("Enter a number: ");
    scanf("%d", &num);
            while (num != 0)
            {
        reverse_num = reverse_num * 10 + num % 10;
        num /= 10;
    }
    printf("Reverse Number: %d", reverse_num);
```

```
    return 0;
}
```

Output

```
Enter a number: 25142
Reverse Number: 24152
```

25)Write a C program to find the number of student users in the college, get the total users, staff users details as input. Note for every 3 staff user there is one Non teaching staff user assigned by default.

```
#include <stdio.h>

int main()
{
    int num_student_users, num_total_users, num_staff_users, num_nt_staff_users;
    printf("Enter the number of student users: ");
    scanf("%d", &num_student_users);
    printf("Enter the total number of users: ");
    scanf("%d", &num_total_users);
    printf("Enter the number of staff users: ");
    scanf("%d", &num_staff_users);
    num_nt_staff_users = num_staff_users / 3;
    printf("Number of student users: %d\n", num_student_users);
    printf("Number of total users: %d\n", num_total_users);
    printf("Number of staff users: %d\n", num_staff_users);
    printf("Number of non-teaching staff users: %d\n", num_nt_staff_users);
    return 0;
}
```

Output

```
Enter the number of student users: 123
Enter the total number of users: 333
Enter the number of staff users: 240
Number of student users: 123
Number of total users: 333
Number of staff users: 240
Number of non-teaching staff users: 80
```

26)Decimal to binary conversion

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

```
int main()
{
   int decimal_num, binary_num = 0, base = 1, rem;
   printf("Enter a decimal number: ");
   scanf("%d", &decimal_num);
   while (decimal_num > 0)
   {
      rem = decimal_num % 2;
      binary_num += rem * base;
      decimal_num /= 2;
      base *= 10;
   }

   printf("Binary equivalent: %d\n", binary_num);

   return 0;
}
```

Output

```
Enter a decimal number: 192
Binary equivalent: 11000000
```

27)Write a program to print the all Odd numbers and number of even numbers in between M and N

```
#include <stdio.h>

int main()
{
   int M, N;
         int even_count = 0, odd_count=0;
   printf("Enter M and N separated by a space: ");
   scanf("%d %d", &M, &N);
   printf("Odd numbers between %d and %d: ", M, N);
   for (int i = M; i <= N; i++)
         {
      if (i % 2 != 0)
                  {
         printf("%d ", i);
         ++odd_count;
      }
   }
   printf("\n");
```

```
    printf("Even numbers between %d and %d: ", M, N);
    for (int i = M; i <= N; i++)
            {
        if (i % 2 == 0)
                    {
            printf("%d ", i);
            ++even_count;
        }
    }
    printf("\nNumber of odd numbers between %d and %d: %d", M, N, odd_count);
    printf("\nNumber of even numbers between %d and %d: %d", M, N, even_count);

    return 0;
}
```

Output

```
Enter M and N separated by a space: 25 35
Odd numbers between 25 and 35: 25 27 29 31 33 35
Even numbers between 25 and 35: 26 28 30 32 34
Number of odd numbers between 25 and 35: 6
Number of even numbers between 25 and 35: 5
```

28)Pythagoras theorem

```
#include <stdio.h>
#include <math.h>
int main()
{
    double a, b, c;
    printf("Enter the length of the first side (a): ");
    scanf("%lf", &a);
    printf("Enter the length of the second side (b): ");
    scanf("%lf", &b);
    c = sqrt(a*a + b*b);
    printf("The length of the hypotenuse (c) is: %.2lf\n", c);
    return 0;
}
```

Output

```
Enter the length of the first side (a): 6
Enter the length of the second side (b): 8
```

The length of the hypotenuse (c) is: 10.00

29)Bubble Sort

```
#include <stdio.h>
void bubbleSort(int arr[], int n);
int main()
{
   int arr[100], num;
   printf("Enter the number of elements:");
   scanf("%d", &num);
   for(int i=0; i<num; i++)
   {
         printf("Enter the %d element: ", i+1);
         scanf("%d", &arr[i]);
         }
   printf("Array before sorting: ");
   for (int i = 0; i < num; i++)
         {
      printf("%d ", arr[i]);
   }
   bubbleSort(arr, num);
   printf("\nArray after sorting: ");
   for (int i = 0; i < num; i++)
         {
      printf("%d ", arr[i]);
   }
   return 0;
}

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

```
}
```
Output

```
Enter the number of elements:5
Enter the 1 element: 9
Enter the 2 element: 6
Enter the 3 element: 8
Enter the 4 element: 7
Enter the 5 element: 4
Array before sorting: 9 6 8 7 4
Array after sorting: 4 6 7 8 9
```

30)Vowels and consonants

```
#include <stdio.h>
#include <string.h>
#include <ctype.h>
int main()
{
   char string[100];
   int i, vowels = 0, consonants = 0;
   printf("Enter a string: ");
   fgets(string, sizeof(string), stdin);
   for (i = 0; i < strlen(string); i++)
         {
      if (isalpha(string[i]))
                  {
         if (tolower(string[i]) == 'a' || tolower(string[i]) == 'e' || tolower(string[i]) == 'i' ||
tolower(string[i]) == 'o' || tolower(string[i]) == 'u')
                              {
            vowels++;
         }
         else
                              {
            consonants++;
         }
      }
   }
   printf("Number of vowels: %d\n", vowels);
   printf("Number of consonants: %d\n", consonants);
   return 0;
}
```

Output

```
Enter a string: Saveetha School of engineering
Number of vowels: 12
Number of consonants: 15
```

31)Removing vowel from a string

```
#include <stdio.h>
#include <string.h>
int main()
{
   char string[100], newString[100];
   int i, j = 0;
   printf("Enter a string: ");
   fgets(string, sizeof(string), stdin);
   for (i = 0; i < strlen(string); i++)
         {
      if (string[i] != 'a' && string[i] != 'e' && string[i] != 'i' && string[i] != 'o' && string[i] != 'u' &&
string[i] != 'A' && string[i] != 'E' && string[i] != 'I' && string[i] != 'O' && string[i] != 'U')
                  {
         newString[j] = string[i];
         j++;
      }
   }
   newString[j] = '\0';
   printf("Original string: %s", string);
   printf("String after removing vowels: %s", newString);

   return 0;
}
```

Output

```
Enter a string: Saveetha school of engineering
Original string: Saveetha school of engineering
String after removing vowels: Svth schl f ngnrng
```

32) Anagram Check

angel = glean

arc = car

```
#include <stdio.h>
#include <string.h>
int main()
{
   char string1[100], string2[100];
   int freq1[26] = {0}, freq2[26] = {0}, i;
   printf("Enter first string: ");
   fgets(string1, sizeof(string1), stdin);
   printf("Enter second string: ");
   fgets(string2, sizeof(string2), stdin);
   for (i = 0; i < strlen(string1); i++)
         {
      if (string1[i] >= 'a' && string1[i] <= 'z')
                  {
         freq1[string1[i] - 'a']++;
      }
   }
         for (i = 0; i < strlen(string2); i++)
         {
      if (string2[i] >= 'a' && string2[i] <= 'z')
                  {
         freq2[string2[i] - 'a']++;
      }
   }
         for (i = 0; i < 26; i++)
         {
      if (freq1[i] != freq2[i])
                  {
         printf("The strings are not anagrams.\n");
         return 0;
      }
   }
   printf("The strings are anagrams.\n");
   return 0;
}
```

Output

```
Enter first string: eat
Enter second string: ate
The strings are anagrams.
```

33)Printing fruits names

```
#include <stdio.h>
#include <string.h>
#define MAX_FRUITS 10
int main()
{
    char fruits[MAX_FRUITS][20];
    int num_fruits;
    printf("Enter the number of fruits you want to enter (max %d): ", MAX_FRUITS);
    scanf("%d", &num_fruits);
    if (num_fruits > MAX_FRUITS)
            {
        printf("Error: Maximum number of fruits exceeded.\n");
        return 1;
    }    printf("Enter the names of the fruits:\n");
    for (int i = 0; i < num_fruits; i++)
            {
        scanf("%s", fruits[i]);
    }    printf("Fruits: ");
    for (int i = 0; i < num_fruits; i++)
            {
        printf("\"%s\"", fruits[i]);
        if (i != num_fruits - 1)
                    {
            printf(", ");
        }
    }
    printf("\n");

    return 0;
}
```

Output

```
Enter the number of fruits you want to enter (max 10): 5
Enter the names of the fruits:
apple orange pine grapes banana
Fruits: "apple", "orange", "pine", "grapes", "banana"
```

34)Fibonacci series between intervals

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

```
    int a = 0, b = 1, c, m, n;
    printf("Enter two numbers M and N: ");
    scanf("%d %d", &m, &n);
    printf("Fibonacci series between %d and %d: ", m, n);
    while (b < n)
          {
       if (b >= m)
                   {
          printf("%d ", b);
       }
       c = a + b;
       a = b;
       b = c;
    }
    printf("\n");
    return 0;
}
```

Output

```
Enter two numbers M and N: 2 100
Fibonacci series between 2 and 100: 2 3 5 8 13 21 34 55 89
```

35)Check for armstrong number

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

int main()
{
    int num, original_num, remainder, n = 0, result = 0, power;
    printf("Enter a positive integer: ");
    scanf("%d", &num);
    original_num = num;
    while (original_num != 0)
          {
       original_num /= 10;
       ++n;
    }
    original_num = num;
    while (original_num != 0)
          {
       remainder = original_num % 10;
       power = round(pow(remainder, n));
```

```
        result += power;
        original_num /= 10;
    }
    if (result == num)
            {
        printf("%d is an Armstrong number.\n", num);
    } else
            {
        printf("%d is not an Armstrong number.\n", num);
    }
    return 0;
}
```

Output

```
Enter a positive integer: 371
371 is an Armstrong number.
```