1.Qudratic Equation

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

int main(){
    double a, b, c, discriminant, root1, root2, realPart, imagPart;
    printf("Enter Coffecients of a,b and c:");
    scanf("%lf%lf%lf", &a, &b, &c);
    discriminant=b*b-4*a*c;
    if(discriminant>0){
        root1=(-b + sqrt(discriminant))/(2*a);
        root2=(-b-sqrt(discriminant))/(2*a);
        printf("root 1 =%lf and root 2 = %lf", root1, root2);

    }
    else if(discriminant==0){
        root1=root2=-b/(2*a);
        printf("root1=root2=%lf", root1);

    }
    else{
        realPart=-b/(2*a);
        imagPart=sqrt(-discriminant)/(2*a);
        printf("root 1=%.2lf+%.2lfi and root 2=%.2f-%.2fi", realPart, imagPart,
realPart, imagPart);
    }
    return 0;
}
```

2.Sum and Reverse

```c
#include <stdio.h>
int main(){
    int n, sum, d, reverse;
    printf("enter a number:");
    scanf("%d", &n);
    sum=reverse=0;
    while(n!=0){
        d=n%10;
        sum+=d;
        reverse=reverse*10+d;
        n/=10;

    }
    printf("\nSum of digits = %d", sum);
    printf("\nReverse of digits = %d", reverse);
    return 0;
}
```

3.Fibonacci

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

  int i, n;
  int t1 = 0, t2 = 1;
  int nextTerm = t1 + t2;

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

  printf("Fibonacci Series: %d, %d, ", t1, t2);

  for (i = 3; i <= n; ++i) {
    printf("%d, ", nextTerm);
    t1 = t2;
    t2 = nextTerm;
    nextTerm = t1 + t2;
  }

  return 0;
}
```

4.Count of positives,negtatives,zeroes

```c
#include <stdio.h>
int main(){
    int n, arr[25],i;
     printf("Enter size of array:");
    scanf("%d", &n);
    printf("Enter elements:");
    for(i=0;i<n;i++){
    scanf("%d", &arr[i]);
    }
    int positiveCount=0,negativeCount=0,zeroCount=0;
    for(i=0;i<n;i++){
        if(arr[i]>0)
        positiveCount++;
        else if(arr[i]<0)
        negativeCount++;
        else
        zeroCount++;

    }
    printf("Number of Positive elements=%d\n", positiveCount);
    printf("Number of Negative elements=%d\n", negativeCount);
    printf("Number of  Zeroes=%d\n", zeroCount);
    return 0;

}
```

5.Evaluate

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

int main() {
  int n, i=1, x, j, fact;
  double sum = 1.0;

  printf("Enter the range of numbers: ");
  scanf("%d", &n);
  printf("Enter the value of x: ");
  scanf("%d", &x);

  while (i<=n){
    fact=1;

    for (j = 1; j <= i; j++) {
      fact *= j;
      sum += pow(x, i) / fact;
    }
  i++;
  }

  printf("The sum of series is %0.2f\n", sum);

  return 0;
}
```

6.Pyramid

```c
#include <stdio.h>

int main() {
  int n, i, j;
  printf("Enter the number of rows: ");
  scanf("%d", &n);

  for (i=0;i<=n;i++)
   {
    for (j=1; j<=(2 * n)-1; j++)
    {
      if ((j>= n-(i-1)) && (j<=n+(i-1)))
       {
        printf("*");
      }
      else
        printf(" ");

    }
    printf("\n");
  }

  return 0;
}
```

7.Armstrong

```c
#include <stdio.h>

int main() {
  int low, high, originalnum, sum, digit;

  printf("Enter the lower and upper limits: ");
  scanf("%d %d", &low, &high);

  for (int i = low; i <= high; i++) {
    originalnum = i;
    sum = 0;
    while (originalnum > 0) {
      digit = originalnum % 10;
      sum += digit * digit * digit;
      originalnum /= 10;
    }

    if (sum == i) {
      printf("%d\n", i);
    }
  }

  return 0;
}
```

8.Calculator

```
#include <stdio.h>
#include <math.h>
int main(){
    int choice;
    float n1,n2,X;

printf("1.Addition\n2.Substraction\n3.Multiplication\n4.Division\n5.Power\n6.Square Root\n7.Sine(X)\n8.Cosine(X)\n9.exit\n");
    printf("Enter your choice: ");
    scanf("%d", &choice);

    switch(choice){
        case 1:
               printf("enter two numbers:");
               scanf("%f%f", &n1, &n2);
               printf("The sum of two numbers is %f", n1+n2);
               break;
        case 2:
               printf("enter two numbers:");
               scanf("%f%f", &n1, &n2);
               printf("The diffrence of two numbers is %f", n1-n2);
               break;
        case 3:
               printf("enter two numbers:");
               scanf("%f%f", &n1, &n2);
               printf("The product of two numbers is %f", n1*n2);
               break;
        case 4:
               printf("enter two numbers:");
```

```
            scanf("%f%f", &n1, &n2);
            printf("The quotient of two numbers is %f", n1/n2);
            break;
        case 5:
            printf("enter number and power:");
            scanf("%f%f", &n1, &n2);
            printf("the power = %f", pow(n1, n2));
            break;
        case 6:
            printf("Enter value of  X:");
            scanf("%f", &X);
            printf("The square root of X = %f", sqrt(X));
            break;
         case 7:
            printf("Enter value of  X:");
            scanf("%f", &X);
            printf("The Sine value of X = %f", sin(X));
            break;
        case 8:
            printf("Enter value of  X:");
            scanf("%f", &X);
            printf("The Cosine value of X = %f", cos(X));
            break;
        case 9:
        break;

        default:
            printf("Invalid Choice");
}
```

```
}
```

9.Number pattern

```
#include <stdio.h>

int main(){
    int l, t, n;
    printf("Enter a number:");
    scanf("%d", &n);
    for(l=1,t=n/10;t!=0;t/=10,l*=10);
    printf("\nThe pattern\n");
    for(t=n;t!=0;t%=l,l/=10)
    printf("\n%d", t);
    return 0;

}
```

10.Leap Year

```c
#include <stdio.h>

int main(){
    int startYear, endYear;
    printf("Enter the start year:");
    scanf("%d", &startYear);
     printf("Enter the end year:");
    scanf("%d", &endYear);
    printf("Leap years between %d and %d\n", startYear, endYear);
    for(int year=startYear;year<=endYear;year++)
    {
        if((year%4==0 && year%100!=0) || (year%400 == 0)){
            printf("%d\n", year);
        }
    }
    return 0;
}
```

11.Base conversion

```
#include<stdio.h>
#include<conio.h>
int main()
{
void convert(int,int);
int a,b;
printf("\n enter the number:");
scanf("%d",&a);
printf("\n enter the base:");
scanf("%d",&b);
convert(a,b);
getch();
}
void convert(int a,int b)
{
if(a==0)
return;
int x=a%b;
convert(a/b,b);
if(x<10)
printf("%d",x);
else
printf("%c",55+x);
}
```

12.Standard deviation

```c
#include <stdio.h>
#include <math.h>
int main(){
    int n, i;
    float mean, sd, sum=0, arr[25];
    printf("enter size of array:");
    scanf("%d", &n);
    printf("Enter elements in array: ");
    for(i=0;i<n;i++){
        scanf("%f", &arr[i]);
        sum = sum + arr[i];

    }
    mean=sum/n;
    for(i=0;i<n;i++){
        sum=sum+pow(arr[i]-mean, 2);
        sd=sqrt(sum/n);
    }
    printf("Standard Deviation is %f", sd);
    return 0;


}
```

13.Location

```
#include <stdio.h>
int main(){
    int num, a[25], size, i;
    printf("enter size of array:");
    scanf("%d", &size);
    printf("Enter elements in array: ");
    for(i=0;i<size;i++){
        scanf("%d", &a[i]);

}
printf("Enter element to search:");
scanf("%d", &num);

for(i=0;i<size;i++)
{
    if(a[i]==num){
        printf("Element is at the position %d", i+1);
        break;
    }
    if(i==size){
        printf("Element not found");
    }
}
return 0;
}
```

14.Append array

```c
#include <stdio.h>
int main(){
    int arr1size=5, arr2size=5, arr_resultsize, i, j;
    int a[5]={1,2,3,4,5};
    int b[5]={6,7,8,9,10};
    arr_resultsize = arr1size + arr2size;
    int c[arr_resultsize];
    for(i=0;i<arr1size;i++){
        c[i]=a[i];
    }
     for(i=0, j=arr1size; j<arr_resultsize && i< arr2size;i++,j++){
        c[j]=b[i];
    }
    for(i=0;i<arr_resultsize;i++){
        printf("%d", c[i]);
    }
    return 0;
}
```

15.Currency denomination

```c
#include <stdio.h>
int main() {
    int a[9]={500,200,100,50,20,10,5,2,1},m,temp,i;

    printf("enter the amount: ");
    scanf("%d",&m);
    temp=m;
    for(i=0;i<9;i++)
    {
        printf("\n%d notes:%d",a[i],temp/a[i]);
        temp=temp%a[i];
    }

    return 0;
}
```

16.Transpose of Matrix

```c
#include <stdio.h>

int main() {
  int m, n, i, j;
  int matrix[10][10], transpose[10][10];

  printf("Enter rows and columns: ");
  scanf("%d %d", &m, &n);

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

  // Transpose the matrix.
  for (i = 0; i < m; i++) {
    for (j = 0; j < n; j++) {
      transpose[j][i] = matrix[i][j];
    }
  }

  // Print the transpose of the matrix.
  printf("Transpose of the matrix:\n");
  for (i = 0; i < n; i++) {
    for (j = 0; j < m; j++) {
      printf("%d ", transpose[i][j]);
```

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

  return 0;
}
```

17.diagonal matrix

```c
#include <stdio.h>

int main() {
    int size;

    printf("Enter the size of the matrix: ");
    scanf("%d", &size);

    int matrix[size][size];

    for (int i = 0; i < size; i++) {
        for (int j = 0; j < size; j++) {
            if (i < j) {
                matrix[i][j] = 1;
            } else if (i > j) {
                matrix[i][j] = -1;
            } else {
                matrix[i][j] = 0;
            }
        }
    }

    printf("Resultant Matrix:\n");
    for (int i = 0; i < size; i++) {
        for (int j = 0; j < size; j++) {
            printf("%d ", matrix[i][j]);
        }
        printf("\n");
```

```
    }

    return 0;
}
```

18.Matrix Multiplication

```c
#include <stdio.h>

int main() {
    int rows1, cols1, rows2, cols2;

    printf("Enter the number of rows and columns for the first matrix: ");
    scanf("%d %d", &rows1, &cols1);

    printf("Enter the number of rows and columns for the second matrix: ");
    scanf("%d %d", &rows2, &cols2);

    if (cols1 != rows2) {
        printf("Matrix multiplication is not possible.\n");
        return 1;
    }

    int matrix1[rows1][cols1];
    int matrix2[rows2][cols2];
    int result[rows1][cols2];

    printf("Enter elements for the first matrix:\n");
    for (int i = 0; i < rows1; i++) {
        for (int j = 0; j < cols1; j++) {
            scanf("%d", &matrix1[i][j]);
        }
    }

    printf("Enter elements for the second matrix:\n");
```

```c
    for (int i = 0; i < rows2; i++) {
        for (int j = 0; j < cols2; j++) {
            scanf("%d", &matrix2[i][j]);
        }
    }

    // Perform matrix multiplication
    for (int i = 0; i < rows1; i++) {
        for (int j = 0; j < cols2; j++) {
            result[i][j] = 0;
            for (int k = 0; k < cols1; k++) {
                result[i][j] += matrix1[i][k] * matrix2[k][j];
            }
        }
    }

    printf("Resultant Matrix after multiplication:\n");
    for (int i = 0; i < rows1; i++) {
        for (int j = 0; j < cols2; j++) {
            printf("%d ", result[i][j]);
        }
        printf("\n");
    }

    return 0;
}
```

19.Recursion

```c
#include<stdio.h>
long int multiplyNumbers(int n);
int main() {
    int n;
    printf("Enter a positive integer: ");
    scanf("%d",&n);
    printf("Factorial of %d = %ld", n, multiplyNumbers(n));
    return 0;
}

long int multiplyNumbers(int n) {
    if (n>=1)
        return n*multiplyNumbers(n-1);
    else
        return 1;
}
```

20.Prime number

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

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

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

    printf("Prime numbers in the group:\n");
    for (int i = 0; i < n; i++) {
        bool isPrime = true;
        if (numbers[i] <= 1) {
            isPrime = false;
        } else {
            for (int j = 2; j * j <= numbers[i]; j++) {
                if (numbers[i] % j == 0) {
                    isPrime = false;
                    break;
                }
            }
        }
        if (isPrime) {
```

```
            printf("%d ", numbers[i]);
        }
    }

    return 0;
}
```

21.Short form

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

int main()
{
    char str[100];
    int i;
    printf("enter a string : ");
    gets(str);
    printf("\nShort form is ");
    printf("%c",str[0]);
    for(i=0;str[i]!='\0';i++)
    {
       if(str[i]==' ' && str[i+1]!=' '){
            printf("%c",str[i+1]);
        }
    }
}
```

22.Vowels

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

int main() {
    char str[100];
    int count = 0;

    printf("Enter a string: ");
    gets(str);

    for (int i = 0; str[i] != '\0'; i++) {
        if (str[i] == 'a' || str[i] == 'e' || str[i] == 'i' || str[i] == 'o' || str[i] == 'u' ||
            str[i] == 'A' || str[i] == 'E' || str[i] == 'I' || str[i] == 'O' || str[i] == 'U') {
            count++;
        }
    }

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

    return 0;
}
```

23.Marklist

```c
#include <stdio.h>

struct Student {
    char name[50];
    int rollNumber;
    int marks;
};

int main() {
    int n;

    printf("Enter the number of students: ");
    scanf("%d", &n);

    struct Student students[n];

    for (int i = 0; i < n; i++) {
        printf("Enter name, roll number and marks for student %d:\n", i + 1);
        scanf("%s %d %d", students[i].name, &students[i].rollNumber,
&students[i].marks);
    }

    printf("Mark List:\n");
    printf("Name\tRoll Number\tMarks\n");
    for (int i = 0; i < n; i++) {
        printf("%s\t%d\t\t%d\n", students[i].name, students[i].rollNumber,
students[i].marks);
    }
```

```
    return 0;
}
```

24.Length of string

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

int main() {

    char str[100], * ptr;
    int count;

    printf("Enter any string: ");
    gets(str);


    ptr = str;


    count = 0;

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

    printf("The length of the string is: %d", count);

    return 0;
```

```
}
```

25.Two files

```c
#include <stdio.h>
#include<stdlib.h>
int main(){
    int i, n, value;
    printf("Enter number of elements:  ");
    scanf("%d", &n);
    FILE *fptro, *fptre;
    fptre = (fopen("even.txt", "w"));
    fptro = (fopen("odd.txt", "w"));
    if(fptre == NULL)
    {
        printf("!ERROR");
        exit(1);
    }
    for(i=0;i<n;++i){
        scanf("%d", &value);
        if(value%2==0)
        fprintf(fptre,"%d\n", value);
        else
        fprintf(fptro,"%d\n", value);
    }
    fclose(fptre);
    fclose(fptro);
    return 0;
}
```