1. Write a C program to find cube of any number using function.
2. Write a C program to find diameter, circumference and area of circle using functions.
3. Write a C program to find maximum and minimum between two numbers using functions.
4. Write a C program to check whether a number is even or odd using functions.
5. Write a C program to check whether a number is prime, Armstrong or perfect number using functions.
6. Write a C program to find all prime numbers between given interval using functions.
7. Write a C program to print all strong numbers between given interval using functions.
8. Write a C program to print all Armstrong numbers between given interval using functions.
9. Write a C program to print all perfect numbers between given interval using functions.
10. Write a C program to find power of any number using function.
11. Write a C program to print all natural numbers between 1 to n using function.
12. Write a C program to print all even or odd numbers in given range using function.
13. Write a C program to find sum of all natural numbers between 1 to n using function.
14. Write a C program to find sum of all even or odd numbers in given range using function.
15. Write a C program to find reverse of any number using function.
16. Write a C program to check whether a number is palindrome or not using function.
17. Write a C program to find sum of digits of a given number using function.
18. Write a C program to find factorial of any number using function.
19. Write a C program to generate nth Fibonacci term using function.
20. Write a C program to find GCD (HCF) of two numbers using function.
21. Write a C program to find LCM of two numbers using function.
22. Write a C program to display all array elements using function.
23. Write a C program to find sum of elements of array using function.
24. Write a C program to find maximum and minimum elements in array using function.
25. Write a function **PrintMenu** which prints following menu on the screen, takes an option(choice) from the user and returns that option. Prototype of PrintMenu is **int PrintMenu();**

|  |
| --- |
| Options:  0 ------------ To Quit  1 ------------ To print Solid Square  2 ------------ To print Solid Rectangle  3 ------------ To print Multiple Solid Rectangles  4 ------------ To print Multiple Solid Rectangles using function call.  5 ------------ Find Min and Max  6 ------------ Print Prime Numbers  Select your option: |

1. Write a function **PrintDiamond** which takes **noOfLines** (always an odd number) as input and prints a solid diamond. For example if the user inputs 5, then your program should print:

|  |
| --- |
| \*  \*\*\*  \*\*\*\*\*  \*\*\*  \* |

1. Problem 40: Write a function that takes the coefficients of a two-degree polynomial and outputs whether the polynomial has real roots, repeated roots or complex roots. If the roots are repeated, then return the roots else if roots are distinct return either of the roots.

If the discriminant , the roots are real,

Else if the discriminant,, the roots are repeated.

Else, If the discriminant,, the roots are Imaginary.

Also,

**Problem 14**: Write a function that outputs a multiply sign:

  \*            \*  
    \*      \*  
        \*  
    \*      \*  
 \*            \*

Call the two functions from main and make sure they are working correctly.

**Problem 15**: You can create a rotating fan by first printing the plus sign (of problem 2) and then clear the screen and print the multiply sign (of problem 2). It should be done repeatedly to show the rotation of a fan.

NOTE: The screen changes too quickly. To slow down the animation you can use the Sleep function in between. Sleep(500), for example will halt your program for 500 milliseconds. Include <windows.h> to use the Sleep function.

NOTE: Sleep is with capital S. Also, the include file windows should be written with .h extension.

Congratulations! Now you have created your first animation

**Problem 16**: We have 3 stones. The weight of each stone is an integer value. The total weight of these stones is 20. What are the possible weights of the three stones?For example one possibility is 1,1,18. You don’t have to repeat the numbers in a combination, e.g., 1,1,18 can be treated in a similar manner as 18,1,1. The other possibility is 1,5,14,. Implement a function:

OutputWeightCombinations

that outputs all possible combinations. What should be its parameters? Now generalize your function, so that it takes as parameter the sum of the weight of stones and outputs the possible combinations. How many for/while loops do you require?

**Problem 17**: Write a function randomWalk that takes as input the initial position of a robot moving on the screen. The function should place a character at the initial position and then randomly decide which direction to move next and place a character at the new position. The robot has to keep walking randomly till the space key is pressed. You can use the CheckKeyPressed function of myconsole.h library.

To generate a random number use the rand() function of the math library (so include <math.h>). The rand function will generate a random number within a large range. However, you can convert this large number to a number from 0-3 by taking the modulus with 4.

If the random number generated is 0, you could move the robot one step up, if it is 1 then you could move the robot left, for 2 move the robot right and for 3 move the robot one step down. To watch this as an animation use the Sleep() function within your loop.

**Problem 18**: Write a function plotLine that inputs the initial and final coordinates and plots a line between the two points. In main function input the two coordinates from the user and then display the line by calling plotLine.

**Problem 19**: Modify your cpp file. Write a function **PrintMenu** which prints following menu on the screen, takes an option(choice) from the user and returns that option. Prototype of PrintMenu is

**int PrintMenu();**

|  |
| --- |
| Options:  0 ------------ To Quit  1 ------------ To print Solid Square  2 ------------ To print Solid Rectangle  3 ------------ To print Multiple Solid Rectangles  4 ------------ To print Multiple Solid Rectangles using function call.  5 ------------ Find Min and Max  6 ------------ Print Prime Numbers  Select your option: |

**Problem 20**: Write a function **PrintSolidSquare** which takes a character ‘symbol’ and a number ‘side’ from user and prints a square filled with symbol. For example, if user enters the symbol ‘%’ and side = 5, your output will be as shown below. Functions prototype will be

**void PrintSolidSquare( int, char );**

**Problem 21**: Write a function **PrintSolidRectangle** which takes two integers ‘Length’ and ‘Width’ and a character ‘symbol’ and prints a rectangle filled with symbols. Remember that length of a rectangle is always greater than its width so you have to add *Data Validation* in your function. Think of an appropriate prototype for your function. Sample output for Length = 7, width = 5 and Symbol = @ is given below

|  |
| --- |
| @@@@@  @@@@@  @@@@@  @@@@@  @@@@@  @@@@@  @@@@@ |

**Problem 22**: Write a function **PrintMultipleSolidRectangles** which takes three integers i.e. ‘TotalNumberOfRectangles’, ‘Lenght’, ‘Width’ and a character ‘Symbol’ and prints multiple solid rectangles of same size. Think of an appropriate prototype for your function. You cannot call your ‘**PrintSolidRectangle**’ function in this function. Sample output for ‘TotalNumberOfRectangles = 3’, ‘Length = 4’, ‘Width = 3’ and ‘Symbol = \*’ is shown below:

|  |
| --- |
| Rectangle 1:  \*\*\*  \*\*\*  \*\*\*  \*\*\*  Rectangle 2:  \*\*\*  \*\*\*  \*\*\*  \*\*\*  Rectangle 3:  \*\*\*  \*\*\*  \*\*\*  \*\*\* |