C++ Assignments

Day1:

Basics Assignments

1. Print Hello World

2. Add two numbers/binary numbers/characters

3. **Calculate compound interest**

4. Calculate power of a number

Function Assignments

5. Swap two numbers

6. Calculate area of rectangle

7. **Get 4 digit number from user, write a function to calculate sum of digits of the 4-digit number:**
   - **Using loop**
   - **Using recursion**

8. Write a recursive function to obtain n numbers of a Fibonacci series
   0 1 1 2 3 5 8 13 21 34 55 89...

9. **Write a function to find the binary equivalent of this number using recursion.**

Day2:

10. Find roots of equations.

11. Check given number is odd or even number.

12. **Check if the entered char is small letter or capital letter.**

13. Write a recursive function to obtain the running sum of n numbers.

14. **Write functions for calculating area of circle, rectangle, and square and call these functions from main function.**

15. **Write a function for factorial and use the function for printing factorial series like 1 2 6 24 120……**

Day3:

16. Print below patterns and their vertical flip

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

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

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

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

Assignments on Array

17. Accept n numbers from users and store them in to array. Write a function to sort and display sorted array.

18. Accept n characters from user and print is as string.

19. Accept n numbers from user and write a function to find out average and display average.

20. Accept n numbers from user and write a function to multiply each number by 2 and display result.

21. **Accept n number from user and write functions for finding min, max, second max and second min from these numbers**

Day4:

22. Declare function and call it by reference for swapping numbers

23. **Create function for printing an array and call it from main (Student array)**

24. **Function for print factorial series with recursive**

25. Reverse string (char array) with recursion

Day5:

26. Create Student struct and add rollNo and name in it. Create Accept() function which will take an array of Students as input. Write one more function to display student data. This Display() function should receive Student array.

    Menus:

    1. Accept n numbers from users and store them in to dynamically allocated array. Write a function to sort and display sorted array.

    2. Accept n characters from user and print is as string.

    3. Accept n numbers from user and write a function to find out average and display average.

    4. Accept n numbers from user and write a function to multiply each number by 2 and display result.

Day6:

27. Create Student struct and add rollNo and name in it. Create Accept() function which will take an array of Students as input. Write one more function to display student data. This Display() function should receive Student array. Sort an array of students by rollNo.

28. **Create Employee class and add empNo, name, salary and dept in it. Crete Accept function which will take an input. Write one more function to display Employee data. Sort Employees by salary, name, empNo.**

Day7:

29. Implement constructor, parameterized constructor, Accept, Display functions in Student class. Try to create Student objects using new operator. Try creating student array and check if constructors are getting multiple times. Write constructor with initialization list for const D.M initialization. Print this pointer inside function to check if it points to invoking object.

30. **Implement constructor, parameterized constructor, Accept, Display functions in Account class.**
    Try to create Account objects using new operator.
    Provide option to add an account to array for open account and display account details.
    Print this pointer inside function to check if it points to invoking object.
    Write menu driven program for this. Find account and do transactions like withdraw and deposit.

Day8:

31. **Use static data member to generate account number in above Account assignment. Try both styles with and without pointer.**
    Use separate header and implementation files.

    Try Student, Complex and Employee class assignments with separate header and implementation.

    Use static data member in complex class to find out how many object are created. Write parameterized and no-arg ctor.

Day9:

32. Use references to write swap function.
33.
34. **Try passing const reference to factorial function and check if modification is possible.**
35. Use array of pointers in account assignment.
36. **Create an array class and implement constructor, destructor, Accept Display function.**
37. **Overload +, -, * operator for Complex number class**

Day10

38. **Write Time class that allow you to add, subtract, read and print simple Time in hh:mm:ss formats. Use function overloading in your program. Overload operator + and -**

    Overload << and >> operators for Student and Account class

    Implement association between Student and Address class. Please do it in separate header and separate cpp file

Day11:

39. Practice Player and Shape examples from class
40. **Create Employee class hierarchy with two child classes PermEmployee and ContractEmployee. Create CalcSalary function as abstract function in base class and override it in child classes. Demonstrate runtime polymorphism with the help of base class pointer reference.**

Day12:

41. **Create MobilePhone class with serialNo, modelNo, brandName and price.**
    Write menu driven program with below options
    1. Add /write MobilePhone to file
    2. Read all Mobile phones from file and store them in an array
    3. Read all Mobile phones from file and store them in an array and display sorted by price.

42. **Write a program to copy binary file to binary file. Specifically large music files.**

Day13:

43. **Write swap, add, subtract, multiplication and division functions using template.**

44. **Create and Array class and apply class template to it. Try storing int, float char data by creating different objects of an array class.**

Day14:

45. **Try STL containers Vector, Stack, List, Set and Map. Use iterators to traverse the container.**