Object Oriented Programming
Lab Task#10

Note:
- Copied task will be awarded zero marks.
- Note that these lab task marks could be graded through a viva in lab.
- Submit the .cpp file for each task in google classroom and rename it with your roll no and name.
  For example: Ali_Ahmed_22p9023_task1.cpp.
- Lab Tasks will be graded in Lab.
- properly comment your code for user understanding purpose.
- Marks will be deducted if above instructions are not followed.

Task:1

Scenario: You are developing a contact management software that allows users to keep track of their contacts. Each contact in the software consists of a person's name, phone number, email address, and date of birth. You want to implement this system using object-oriented programming principles, with a composition relationship. Here, the class Contact is composed of n numbers of contacts(array of contacts). And the total count of contacts(size of array), each contact is composed of the object of date of birth class plus some other data members like name, phone_no, and email. Also design a method in the ContactBook class to print the names of all contacts in the book who have a birthday in a specific month.

Note: Your program must fulfill the criteria of composition.

Task:2

Scenario: You are working on a project that involves complex numbers in C++. You have created a class called "Complex" that represents a complex number. You want to implement a friend function that adds two complex numbers together and returns the result. Define a "Complex" class with the following private data members: int real: the real part of the complex number int imag: the imaginary part of the complex number Define a friend function called "addComplex" that takes two "Complex" objects as arguments and returns a new "Complex" object representing the sum of the two complex numbers. The friend function should have the following signature: Complex addComplex(Complex c1, Complex c2) In the "Complex" class, define a public member function called "display" that displays the complex number in the format a + bi, where a is the real part and b is the imaginary part. In the main function, create two "Complex" objects and add them together using the "addComplex" function. Display the result using the "display" function.