

BLG332E

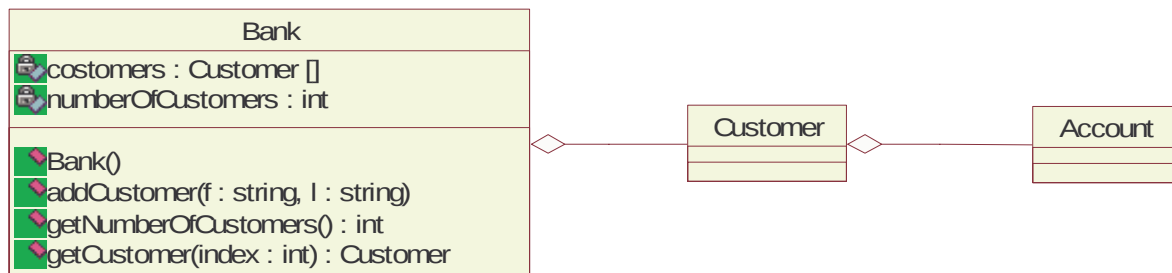
Object Oriented Programming

Practice Session 4

Exercise 7

In this exercise, you will add the Bank class as modeled by the UML figure. A bank object keeps track of an association between itself and its customers. You implement this aggregate association with an array of Customer objects. You will also need to keep an integer attribute that keeps track of how many customers currently exist in the bank

Task 1: Change your working directory to `chap03/exercise7`



Task 2: Copy the previous project files (`Customer.cpp`, `Customer.h`, `Account.h`, `Account.cpp`) to the working directory.

Task 3: Create a class `Bank` with two attributes: `customers` and `numberOfCustomers`.

- Add a public constructor that initializes the `customers` array with some appropriate maximum size (at least bigger than 5).
- Add the `addCustomer` method. This method must construct a new `Customer` object from the parameters (first name, last name) and place it on the `customers` array. It must also increment the `numberOfCustomers` attribute.
- Add the `getNumberOfCustomers` accessor method, which return the `numberOfCustomers` attribute.
- Add the `getCustomer` method. This method returns the customer associated with the given `index` parameter.

Task 4: Read the `TestBanking.cpp` code.

Task 5: Compile `Account.cpp`, `Customer.cpp`, `Bank.cpp` and `TestBanking.cpp`.

Task 6: Run the program. You should see the following output:

```
Customer [1] is Simms, Jane
Customer [2] is Bryant, Owen
Customer [3] is Soley, Tim
Customer [4] is Soley, Maria
```