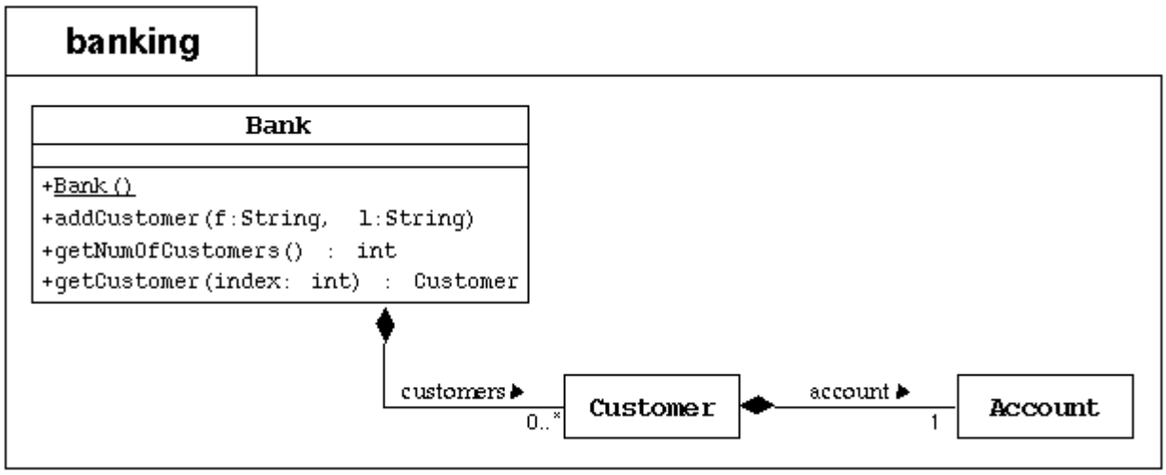
Course Number	*CSci 120*
Descriptive Title	*Object-oriented Programming*
Programming Language	*Java*

Problem Set Number	*5*
Problem Number	*2*
Activity Title	*Use Arrays to Represent Associations*

Objective

In this exercise you will use arrays to implement the multiplicity on the association between a bank and its customers.



Directions

To the `banking` package, you will add the `Bank` class as modeled by the UML diagram above. A bank object simply keeps track of an association between itself and its customers. We will implement this aggregate association with an array of `Customer` objects. We will also need to keep an integer attribute that keeps track of how many customers currently exist in the bank.

1. Create the `banking` directory. Copy the previous Banking project files in this package directory.
2. Add two attributes to the `Bank` class: `customers` (an array of `Customer` objects) and `numberOfCustomers` (an integer that keeps track of the next `customers` array index).
3. Add a public constructor that initializes the `customers` array with some appropriate maximum size (at least bigger than 5).
4. 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.
5. Add the `getNumOfCustomers` accessor method, which returns the `numberOfCustomers` attribute.
6. Add the `getCustomer` method. This method returns the customer associated with the given `index` parameter.
7. Compile and run the `TestBanking` 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
```