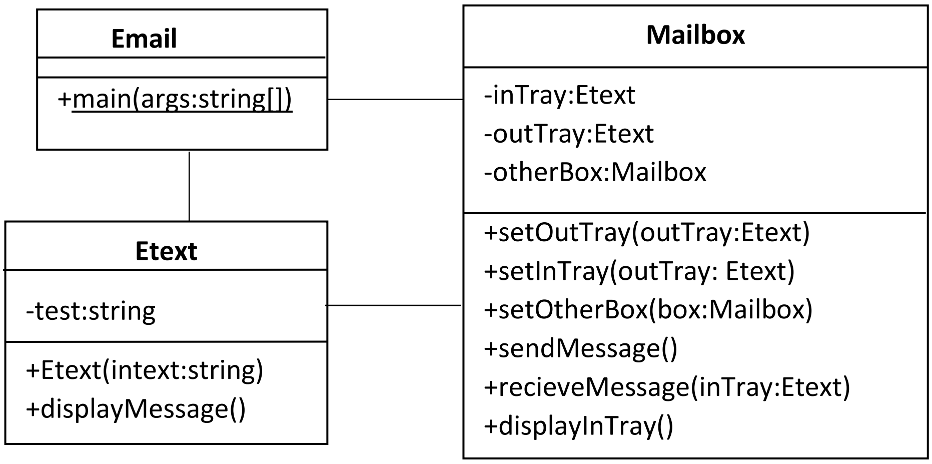
Class Association and Testing

If objects of two classes interact in some way (either by referral of either object within the same block of code or by passing of an object parameter to another object of a different class) then the two classes have a relationship called **Association**. In a class diagram such a relationship is expressed as a straight line drawn between the two classes. Using the class diagram below implement a program that simulates the transmission of an email message from one mailbox to another mailbox (ignore the fact that this would probably occur via a server sitting between the mailboxes). You should also create a third class also shown in the diagram which contains main() and which coordinates the sending and receiving of the message.



The source code for Main() is given below and should be included in your solution *without modification.*

```
Mailbox myBox = new Mailbox();                  // Create mailboxes
Mailbox yourBox = new Mailbox();
myBox.setOtherBox(yourBox);                     // Set up a two way coms link
yourBox.setOtherBox(myBox);
Etext myMessage = new Etext("From me to you");  // Create message
myBox.setOutTray(myMessage);                    // Copy message to myBox outTray
myBox.sendMessage();                            // Transfer message to yourBox inTray
yourBox.displayInTray();                        // Display message in yourBox inTray
```

Once you have a compiled version that produces a result stop development and design a test plan and results. In the prac class time available I suggest you develop two unit tests for class Mailbox based on two methods, and two integration tests. If you find it difficult to separate unit from integration tests just generically devise 4 tests. The test plan should comprise written details of the 4 tests and the Test results should be tabulated with details of specific inputs/expectations and outputs/outcomes.