Quiz #1 (10 marks)

On CourseWeb: Course Documents ↪ Week 4 Materials ↪ Quiz 1

Submission Instructions: Submit your code as a single zip file at the CourseWeb page for the quiz. Name your zip file as `<Pitt username>.zip` (e.g., `ksm73.zip`). Remember to include all source-code files including the ones supplied to you. Do not enclose your files inside a folder.

General Guidelines

- Build your solution incrementally. On short intervals, make sure that the following commands produce no compilation errors.

  ```
  javac *.java
  java Quiz1Test
  ```

- Do not change anything in `BagInterface.java` and `Quiz1Test.java`.

- Before submission, make sure that the following commands produce no error and give the output that all tests are passed. In grading your submissions, a couple new test cases will be used as well.

  ```
  javac *.java
  java Quiz1Test
  ```

- **You only have one submission attempt.**

Question

1. Download the following three files from the Quiz page on CourseWeb.
   `BagInterface.java`
   `ArrayBag.java`
   `Quiz1Test.java`.

2. Complete the implementation of the following two methods inside the file ArrayBag.java.

   -
     ```
     blueboolean splitInto(BagInterface<T> first, BagInterface<T> second) {
         ...
     }
     ```

     The method will split and add the contents of the bag into two bags that are passed in as arguments. **The bag on which the method is applied doesn't change.** If there are an odd number of items, put the extra item into the first bag. The method will return a boolean value. If either bag overflows, return false. Otherwise, return true. Note that while you will directly access the array of the bag that the method is applied to, you can only use the methods from BagInterface on the arguments.

   -

     ```
     blueboolean addAll(BagInterface<T> toAdd){
     ...
     }
     ```

The method will add all of the items from the argument into the bag. The method will return a boolean value indicating an overflow. If adding the items would cause the bag to overflow, do nothing and return false. Otherwise, add the items and return true. Note that while you will directly access the array of the bag that the method is applied to, you can only use the methods from BagInterface on the argument.

Grading Criteria

All test cases are passed including the unseen ones	10
Some unseen test cases are passed	9
No unseen test cases are passed	5-7
Some seen test cases are not passed	6
Code doesn't compile	0-5