PROGRAMMING ASSIGNMENT 1

Due: Tuesday, January 20th at 11:59pm

This assignment will provide you experience working with

- Arrays as a set
- Classes with UML class diagrams (reading and understanding only)
- Testing with JUnit (adding tests)
- Submission to your repository via gitHub should contain
    - Folder named AssignmentOne containing
        - CardSet.java
        - CardSetTest.java updated with your tests

In this assignment you are given two implemented classes (Card.java and CardSetTest.java). You job is to implement the CardSet.java class according to the UML diagram below and instructions below.

- Internally, CardSet should contain an array of Cards. This can be referred to as a set or hand.
- There can be no duplicate cards in your set.
- You must implement the following methods:
    - `public CardSet(int size)  Constructor for the new CardSet. size is the maximum size of the set. It will have this many "slots" for  Cards, and should be initialized to contain all null pointers.`
    - `public int count()  Returns the number of cards currently in your hand.`
    - `public void add(Card card) Adds the passed card to the CardSet. It should search the slots for the first null pointer, and place that card there. For example, if the hand currently has the Jack of Spades, a null pointer, the 10 of Diamonds, and a null pointer, after adding the 2 of Clubs it'll have the Jack, the 2, the 10, and a null pointer. If there is no space to add the card, it should print back an error message.`
    - `public Card discard(int whichCard)  Discards the card whose index is the whichCard argument, replacing it with a null pointer.  So for the above  example, if we then discard Card #2, it'll get rid of the 10. This function should return the card that was  discarded.`
    - `public void print()  Prints out every card in the hand.`
    - `public int findHighRank()  Returns the rank of the highest card in the hand. If that card is an ace, it should return 14.`
    - `public int findLowRank()  Returns the rank of the lowest card in the hand. If that card is an ace (very rare), it should return 14.`
    - `public int findHighPairRank()  Returns the rank of the highest pair of cards with identical rank. Again, return 14 for aces. If there is no  pair, return -1.`

- public boolean containsPair() Returns true only if there is a pair of cards with identical rank.
- public boolean isFlush() Returns true only if the cards are all of the same suit.
- public boolean isStraight() Returns true only if the cards' ranks are sequential numbers (though they may be out of order). For this function, an ace may be either low or high.
- Extra Credit:
  - public boolean isStraightFlush() Returns true only if the CardSet is both a straight and a flush.
  - You must test for this (valid and invalid result) in your CardSetTest.java to receive credit.

Your code will be tested using CardSetTest.java, with additional tests written by the graders. You should modify CardSetTest.java to include additional tests that you do. Please see grading rubric for number of tests required and other grading criteria.