Week 6 Activities

This Week's Objectives:

Students will be able to:

Describe, analyze, and be able to develop a stack or a queue using arrays

Explain the use of an iterator class or interface

Explain the use of, and develop an iterator operation

This Week's Assignment(s):

Read the objectives carefully, make sure you can answer any question provided (before class)

Make sure you understand what each is asking.

Use references as needed, including the textbook, to respond to each objective as needed.

Finally, make sure you can meet all the objectives by the end of the week.

If you can't, come see one of us for help

You are responsible for all of the objectives throughout the course; they are cumulative

All quizzes and examinations will be a part of, or directly related to, the stated objectives

You may be quizzed or tested on any stated objective from the week it was introduced until the end of the semester

It would be a good idea to make and keep notes for all of the objectives you have studied

Next Week's Objectives (Tentative, Subject to Modification):

Students will be able to:

- Demonstrate competence in all objectives provided in the course to this point

- Be prepared to write any identified methods from the first five projects

- Prepare for the course mid-term on 5 March

ASSIGNMENT REQUIREMENTS

This assignment is due by 6:00 pm on the date specified as the title of, and at the beginning of, this assignment page. Any assignments turned in after 6:00 pm but received before 11:59 pm on the same date will be accepted with a 50% reduction of the graded score. No assignments will be accepted at or after 12:00 midnight of the due date. Note that the Blackboard Learn clock will be considered the standard; you would be wise to turn your assignments in well before these specified deadline times. Technical and/or time-related issues related to uploading assignments are not likely to be accepted as reasons to recover credit.

ASSIGNMENT OBJECTIVES

- Students will review array operations by creating an iterator class inherited from an ArrayClass and a stack class that uses an ArrayClass

- Students will use multiple classes and provided code to develop a working solution

- Students will review use of Java arrays, and practice basic algorithmic activities with arrays

- Students will review the pass-by-copy practice of Java, along with managing potential aliasing issues in method development

- Students will practice development, implementation, and usage of supporting/utility methods; students will not use Java utilities such as length, size, etc., unless they are specifically permitted

- Students will practice implementing methods in such a way that code is rarely, if ever, repeated by implementing supporting or utility methods in their development activities

- Students will demonstrate analytic and diagnostic competence with testing and verifying all components of the given assignment

ASSIGNMENT OVERVIEW

- Students will develop an iterator class and a stack class using an ArrayClass as an engine either in an ISA format or a HASA format

ASSIGNMENT SPECIFICATIONS

- Students must develop the code to create the two classes. Note that the ArrayClass must be fully functional and correct; credit will be reduced if any part of the ArrayClass is not fully and correctly operational

- Students will also document all classes and methods using the Javadoc commenting process; comments are not required to be exactly the same as found in the supporting document (below) but they should be semantically equivalent

- Students will upload the ArrayClass.java, IteratorClass.java, and StackClass.java files to this assignment; any other uploaded files will result in a reduction of the project grade, and in some cases, may cause a complete loss of credit (e.g., uploading *.class, or other unusable files)

- Other Constraints:

* The General Usage Rubric will be applied to all code developed by the students; verify that your code does not incur any credit reduction due to these standards

* All methods must be placed in the class files in alphabetical order so methods can be found by graders; methods not found will not be graded

* Students are given a maximum of three attempts to upload their work to this BBLearn assignment page

ASSIGNMENT DOCUMENTS

Provided display samples:

```
displayIteratorList() - shown here:

Iterator List:
27, 34, [56], 42, 61

==================================================

displayStack() - shown here:

Stack Display:
Bottom of Stack -|27, 34, 56, 42, 61<- Top of Stack
```