A02 Randomized Queues and Deques Part 1

MISSING 30 Possible Points

| 9/30/2022

Attempt 1

IN PROGRESS
Next Up: Submit Assignment

Add Comment

Unlimited Attempts Allowed

Details

Fundamentals

Assignment: Randomized Queues and Deques Part 1

Learning Objectives

- Use Java generics and create generic classes.
- Select the appropriate data structure(s) to solve a problem given a set of programming specifications.
- Determine the performance characteristics of a software application.
- Implement elementary data structures using arrays and linked lists
- Implement iterators.

Overview

This assignment should be completed with a partner

In this assignment, you will write a generic data type for a deque and a randomized queue.

The assignment is broken up into two parts. In this first part of the assignment, you are asked to implement one out of two classes.

Instruction

Find a partner unless you got one assigned already. Try to have a different one for each of the team assignments. This allows you to work with multiple classmates, strengthen your teamwork skills, and build a network of friends and colleagues that can support you as you transition to a 4-year college or the workplace.

Assignment Instructions:

The assignment instructions are based on an assignment of Princeton's algorithm course. It comes with additional resources: a checklist that includes FAQ and a video that helps you get started.
http://www.cs.princeton.edu/courses/archive/fall14/cos226/assignments/queues.html (http://www.cs.princeton.edu/courses/archive/fall14/cos226/assignments/queues.html)

This first part of the assignment requires the implementation of the class **RandomizedQueue**. It should be created in a **package called a02**.

In order to earn full points, the class needs to pass most of the JUnit tests on CodePost (see rubric). You can submit as often as you like but only up until the specified deadline.
Before you submit via Canvas, take a screen-shot of the CodePost website that shows your bruinmail, the class that is tested, and the number of passing/failing tests.

Additional Resources:

Previous (https://slcc.instructure.com/courses/817632/modules/items/18752951)

Submit Assignment

Next (https://slcc.instructure.com/courses/817632/modules/items/18752955)

Checklist:
Here is a checklist that includes frequently asked questions, suggestions on testing, and possible progress steps.
http://www.cs.princeton.edu/courses/archive/fall14/cos226/checklist/queues.html (http://www.cs.princeton.edu/courses/archive/fall14/cos226/checklist/queues.html)

Submission

One team member embeds the screenshot from CodePost that shows your bruinmail, the assignment, and the number of passing tests. How to embed images in Canvas (https://community.canvaslms.com/t5/Student-Guide/How-do-I-embed-images-from-Canvas-into-the-Rich-Content-Editor/ta-p/356)

Both team members submit the name of the partner and the discussed pebble distribution. If the pebble distribution is not 50/50, include a description that explains the difference.

View Rubric

A02 - Part 1 Rubric

Criteria	Ratings								Pts
JUnit tests for class RandomizedQueue view longer description	**30 pts Full Marks** No more than 1 test fails	**26 pts 2 Tests fail**	**22 pts 3 tests fail**	**18 pts 4 tests fail**	**14 pts 5 tests fail**	**10 pts 6 tests fail**	**6 pts 7 or more tests fail**	**0 pts Insufficient** No submission or no evidence of passed tests	/ 30 pts
									Total Points: 0

Keep in mind, this submission will count for everyone in your Project Groups group.

Choose a submission type

Text | Media | Upload | Office 365 | More

Previous (https://slcc.instructure.com/courses/817632/modules/items/18752951)

Submit Assignment

Next (https://slcc.instructure.com/courses/817632/modules/items/18752955)