CSC207 Lab 4 – Java generics

To earn lab marks, you must arrive on time, actively participate for the entire session, and make good progress.

1 Overview

This week, you are going to implement a generic interface and two generic classes.

2 Log in and get things set up

`s1` drives and `s2` navigates.

1. `pull` into your git repository to get the newly created `lab4` directory. This directory contains directories `Week4Lab`, `src`, and `w4lab`, and file `Main.java` and `WaitList.java`.
2. Start Eclipse and select `lab4` as the workspace to work in (File → switch workspace)
3. Create a new Java Project called `Week4Lab`. You should now be able to view the starter files in Eclipse, in package `w4lab`.

3 The Various Waiting Lists

As starter code, we have provided the generic class `WaitList` that you may have seen in lecture. Look it over carefully.
We now decide that we need two more kinds of wait lists:

- `BoundedWaitList`: This wait list has a limited capacity, specified at the time of creation. It does not accept more than capacity elements.
- `UnfairWaitList`: In this wait list, it is possible to remove an element that is not the first in line — and not return it!. (In our implementation, we will remove the first occurrence of the given element.) It is also possible for an element to be sent back to the end of the line.

After giving the task some thought, we decide that we want an interface `IWaitList`, and three classes for the three wait lists. We also decide that `WaitList` should be a super class of the other two wait lists.

4 Your task

Here's what to do:

1. Read the UML class diagram in figure 1 and observe how it expresses what we said above in words. Make sure that you understand all aspects of the diagram. Get any clarifications needed from your TA.
2. Extend and modify the starter code as needed to fully implement the UML diagram. Include Javadoc comments. **Switch roles after implementing each interface/class!**
3. Study and run `Main`'s `main` method, which uses your new classes and interface.

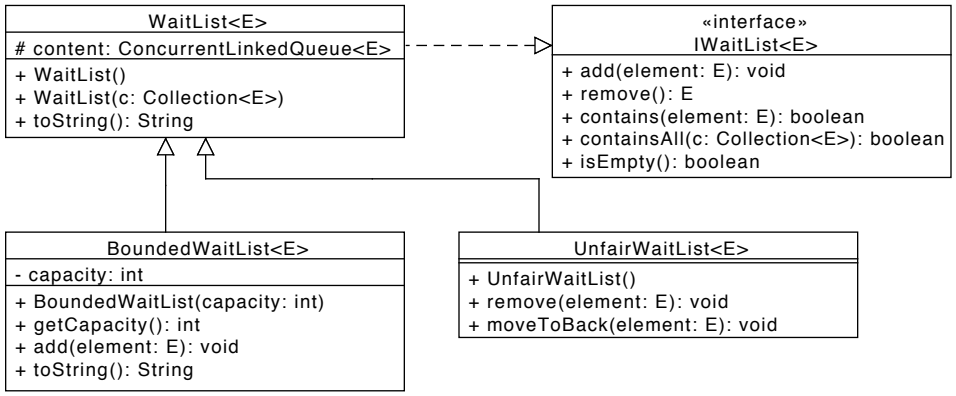

Figure 1: UML diagram for the code you will write.

Show your work (including your Javadocs!) to your TA.

Using a text editor (e.g., `nedit` for the cdf machines), in your `lab4` directory create a file named `partner.txt`. In that file there should be exactly one line and that line should contain only the CDF username of `s2` (assuming you are using `s1`'s repository during this lab). Add and commit that file to the repository.

Add all new Java files to the repository. Commit and push all changes. **Do not include the `bin` folder, the `doc` folder, or Eclipse's "hidden" files!**

5 Find a Team

You are going to be working in teams of two to four people for the project – we've dedicated the last ten minutes of this lab to continue meeting and talking to other students in your lab. Even if you have already found a team, please spend the last fifteen minutes of this lab talking to your neighbors and discussing the project.

It is okay if you don't complete this lab in tutorial in the forty minutes given– you should, however, finish it on your own time if you don't finish it in tutorial. This lab covers material crucial for creating your project, and you should know this lab's material in and out before you start the final project!