CS 323 (Spring 2018)
Assignment 1: Priority Queues
Due: 2/19/17 (11:59pm)

Written Problems (50pts):

From the textbook do problems R-9.5, R-9.6, R-9.18, R-9.19, R-9.24
Of these problems, R-9.18 is probably going to be the toughest technically. As a hint, try to relate the sum to an integral that is easier to solve, rather than trying to simplify the sum.

You will submit your writeup of the problems as a PDF on Canvas.

Programming Problem:

In lecture we will see an implementation of a heap based priority queue using an array to store the heap. In this homework project, you will build a binary tree structure to implement the heap in a file named `HeapTreePQ.java`.

In `HeapTreePQ.java` you will write a full implementation of a heap based priority queue, where the heap has a binary tree structure with nodes and reference links between parent and children. Your class must extend the `AbstractPriorityQueue` class from the textbook (§9.2.3) as well as from lecture. You may add any `private` helper methods as you would like. Program the logic of the tree and heap from scratch; do not rely on structures provided by Java.

While this project is not terribly complex, there are a few things that must be considered, which will be slightly more difficult with this implementation than they are with the array based heap implementation. Plan your structure and your strategies before you start coding in order to save yourself time.

You are responsible for testing to make sure that both your heap and your priority queue implementations can handle any input given to it.

Honor Code The assignment is governed by the College Honor Code and Departmental Policy. Please remember to have the following comment included at the top of the files.

```
/*
THIS CODE WAS MY OWN WORK, IT WAS WRITTEN WITHOUT CONSULTING
CODE WRITTEN BY OTHER STUDENTS OR SOURCES OUTSIDE OF THOSE
PROVIDED BY THE INSTRUCTOR.  _Your Name_Here_
*/
```

Submission:
Place your completed `HeapTreePQ.java` file directly under your `~/cs323/hw1` directory. Then, with the terminal in the same directory, use the turnin commands:

```
~cs323002/turnin HeapTreePQ.java hw1
```

Grading for the program:

- If your program does not compile, you will get 0 points for the programming portion.
- Your heap uses a tree structure and stores elements appropirately, maintaining the heap-order property and completeness property at all times. (20pts)
- The Priority Queue uses your heap implementation and correctly adds and removes elements (20pts)
- Code clarity and style (10pts)