Important

There are general submission guidelines you must always follow. If you fail to follow any of the following guidelines you risk receiving a **0** for the entire assignment.

1. All submitted code must compile under **JDK 8**. This includes unused code, so don't submit extra files that don't compile.

2. Do not include any package declarations in your classes.

3. Do not change any existing class headers, constructors, or method signatures.

4. Do not add additional public methods when implementing an interface.

5. Do not use anything that would trivialize the assignment. (e.g. don't import/use `java.util.LinkedList` for a Linked List assignment. Ask if you are unsure.)

6. You must submit your source code, the `.java` files, not the compiled `.class` files.

7. All methods must be efficient, even if an expected runtime is not specified.

8. . Do not add any new instance variables.

9. After you submit your files redownload them and run them to make sure they are what you intended to submit. You are responsible if you submit the wrong files.

10. The submission on T-Square **MUST** contain only files you want us to grade. **Remove any old versions of files, and submit only the latest files.**

Heaps

You are to code a heap that is backed by an array. This array will start at the size that is in the HeapInterface. Your heap will be a max heap which means that each element in the heap will be larger than both of its children. The first element in your heap should be at index 1. There should be nothing in index 0. See the interface for more information regarding implementation.

You will need to implement private helper methods to help maintain the properties of the heap. You may add as many private helper methods as you want, but do not add any public methods.

Priority Queue

You are to code a Priority Queue that is backed by your heap. See the interface for more information regarding implementation.

Style and Formatting

It is important that your code is not only functional but is also written clearly and with good style. We will be checking your code against a style checker that we are providing. It is located in resources along with instructions on how to use it. We will take off a point for every style error that occurs. If you feel like what you wrote is in accordance with good style but still sets off the style checker please email Jonathan Jemson (jonathanjemson@gatech.edu) with the subject header of "CheckStyle XML".

Javadocs

Javadoc any helper methods you create in a style similar to the Javadocs for the methods in the interface.

Forbidden Statements

You may not use these in your code at any time in CS 1332.

- `break` may only be used in switch-case statements
- `continue`
- `package`
- `System.arrayCopy()`
- `clone()`
- `assert()`
- `Arrays` class
- `Array` class
- `Collections` class
- Reflection APIs

If you use these, we will take off points.

Debug print statements are fine, but should not print anything when we run them. We expect clean runs - printing to the console when we're grading will result in a penalty.

Provided

The following file(s) have been provided to you. There are several, but you will only edit two of them.

1. `HeapInterface.java` This is the interface you will implement when designing your heap. All instructions for what the methods should do are in the javadocs. **Do not alter this file.**

2. `MaxHeap.java` This class is where you will be implementing your heap. **Do not add any public methods to this file.**

3. `PriorityQueueInterface.java` This is the interface you will implement when designing your priority queue. All instructions for what the methods should do are in the javadocs. **Do not alter this file.**

4. `MaxPriorityQueue.java` This class is where you will be implementing your priority queue. **Do not add any public methods to this file.**

5. `HeapPQStudentTest.java` These are some sample JUnits, similar to those that will be used to grade your assignment. **Passing this does not guarantee any sort of grade.**

Deliverables

You must submit all of the following file(s). Please make sure the filename matches the filename(s) below. Be sure you receive the confirmation email from T-Square, and then download your uploaded files to a new folder, copy over the interfaces, recompile, and run. It is your responsibility to re-test your submission and discover editing oddities, upload issues, etc. **Submit only the files you want us to grade, and remove any old versions of files. There should be exactly ONE file for each of the files listed below.**

1. `MaxHeap.java`

2. `MaxPriorityQueue.java`

You may attach each file individually or submit them in a zip archive.