Harrisburg University of Science & Technology

CISC 610 Data Structures & Algorithms

Instructor: Caleb Druckemiller, M.S.

Assignment 4: Heaps

You have been provided a Python file, `heap.py`, which constructs a heap structure with a list. Using that code as a guide:

- Develop a heap data structure using a binary tree structure
- The heap must support add and remove from the heap
- All operations most abide by the rules that govern a heap (see lecture slides for reference)

Once you have your heap structure created, next you must use it as a backing structure to a priority queue.

- Develop a priority queue data structure that is backed by the heap you just created
- Implement the normal methods that accompany a priority queue structure
  - Enqueue, dequeue, and peek by priority not position
  - Also length and whether or not the structure is empty (is_empty)
- Perform the following operations to showcase your working structure
  - Enqueue the following items: 4, 7, 5, 11, 8, 6, 9
  - Dequeue 3 items by priority, they should be 4, 5, & 6.
- Provide a basic interface to allow users to interact with the Priority Queue. This can be a simple CLI that provides the user with the basic options of "Enqueue", "Dequeue", etc.

Your submission should be accompanied by a 8 minute walk-through of your code. This analysis should include your decision making process, the logic behind you code, an your original thoughts that went into the decision making on why your code is written and performs in the manner in which you have written it. If you can not adequately explain how your code functions, it is difficult to believe that you created it yourself as it is inherently difficult to make that which you don't understand.
All video submissions must:

- Be narrated by your own voice - Silent submissions will not be considered
  - If you need accommodations regarding your voice recording, reach out to me **BEFORE** the due date of the assignment
- Capture your screen to include the source code and other assets required by the assignment if necessary for the comprehension of your explanation.

Your submission should include:

- A link to your YouTube/Loom Video upload submitted as a `.txt` file or as a submission comment **OR** a video file (`.mp4` preferred).
  AND
- Your code project (source code, resource files, etc.) unzipped.
- If necessary, provide a README file if an explanation is required to execute your code.