Data Structures

Lab Exercise # 10 (Heap/Priority Queue)

Task 1: **(0.7 Points)**

Download the startercode from Brightspace and complete the following methods of min **Heap** class.

1. **int Heap::getMin()**
   Return (but don't remove) the minimum value from the Heap

2. **int Heap::removeMin()**
   Remove the minimum value from Heap keeping the Heap property intact

3. **int Heap::parent(int i)**
   Returns the index of the parent of the node i

4. **int Heap::left(int i)**
   Returns the index of the left child of the node i

5. **int Heap::right(int i)**
   Returns the index of the right child of the node i

6. **void Heap::siftup(int k)**
   Sift-up an element at index k

7. **void Heap::siftdown(int k)**
   Sift-down an element at index k
   Hint:
   1. if element has no children do nothing
   2. if element has only a left child which is smaller than element then swap the element with left child
   3. if element has both children then swap the element with smaller child
   4. keep repeating step 1-3 until the element/node reaches to its correct position

Task 2: **(0.3 Points**)

Add a method sort() which will sort the heap.

Note: Do not use typical sorting methods e.g. BubbleSort, SelectionSorty etc., instead keep extracting the min value from the heap and store it into an auxiliary array. Replace the original internal array of the Heap with the auxiliary array. Don't forget to delete the original/old array.

Code of Conduct

All assignments are graded, meaning we expect you to adhere to the academic integrity standards of NYU Abu Dhabi. To avoid any confusion regarding this, we will briefly state what is and isn't allowed when working on an assignment/lab-task.

Any documents and program code that you submit must be fully written by yourself. You can, of course, discuss your ideas with fellow students, as long as these discussions are restricted to general solution techniques. Put differently, these discussions should not be about concrete code you are writing, nor about specific results you wish to submit. When discussing an assignment with others, this should never lead to you possessing the

complete or partial solution of others, regardless of whether the solution is in paper or digital form, and independent of who made the solution, meaning you are also not allowed to possess solutions by someone from a different year or course, by someone from another university, or code from the Internet, etc. This also implies that there is never a valid reason to share your code with fellow students, and that there is no valid reason to publish your code online in any form.

Every student is responsible for the work they submit. If there is any doubt during the grading about whether a student created the assignment themselves (e.g. if the solution matches that of others), we reserve the option to let the student explain why this is the case. In case doubts remain, or we decide to directly escalate the issue, the suspected violations will be reported to the academic administration according to the policies of NYU Abu Dhabi.

(see https://students.nyuad.nyu.edu/campus-life/community-standards/policies/academic-integrity/)