

k -ary Max-Heap

In the last classes we learnt about Max/Min - Heaps, its properties and the following operations on them.

- `build_MAX_heap(A)`
- `insert(m)`
- `Extract_max(A)`

The max-heap implementation we dealt with, in the class, was a binary heap (each node had at most 2 children). Similarly, a MAX-Heap can be implemented using a k -ary heap (each node can have at most k children). Remember that a k -ary heap is a complete k -ary tree.

In this assignment, you take an array (of length n) as input and you are required to

1. Check that the array is already a k -ary MAX-heap.
2. If not, implement a k -ary max-Heap. You should allow the above mentioned operations on the k -ary max-Heap (with array representation). Please implement all the operations WITHOUT RECURSION/RECURSIVE METHODS.
3. write an algorithm (`change(A, i, m)`) for changing an element in the middle of max-heap ($A[i] \leftarrow m$) and restore the max-heap property.

Notes:

- Before starting coding revise the algorithms taught in the class.
- Figure out the corresponding algorithms for k -ary heaps.
- To implement `change(A, i, m)`, remember that in class we discussed two algorithms for pushing an element up or down. You can use those routines depending on whether the changed number is bigger or smaller than the previous number.
- Use the template given to you for implementing all the operations.