

CS 60 254 Data Structures and Algorithms

Assignment 3 Due dates (Nov 6 - 8)

For this assignment, you will write a computer program to implement an Abstract Data Type called Priority Queue. A priority queue is like a regular queue, but each element has a "priority" associated with it. In a priority queue, an element with high priority is served before an element with low priority. If two elements have the same priority, they are served according to their enqueue order in the queue.

Implement the Priority Queue as an ADT using two different underlying representations:

1. Static array that stores the elements in priority order
2. Linked list that stores the elements in priority order

Regardless the underlying representations (array, a linked list) your Priority Queue Class should allow any programmer to store any element (data) without understand or knowing the details of the underlying representation.

Your Priority Queue ADT should provide the following interfaces (functions)

1. ***enqueue(element, priority, queue)***: add an element into the queue at the appropriate position in the queue based on the associated priority. If the priority parameter is missing, assign to the element 5 as default priority, which is the lowest priority.
2. ***deque(queue)***: remove the element with the highest priority from the queue. If two elements have the same priority, they are served according to their order in the queue.
3. ***peek(queue)***: return the element to be served from the queue without removing it
4. ***IsRegular(queue)***: return true if all the current elements in the queue have the same priority.
5. ***size(queue)***: returns the total number of elements in the queue.
6. ***isEmpty(queue)***: returns true if the queue is empty.
7. ***isFull(queue)***: returns true if the queue is full.
8. ***display(queue)***: prints all the elements in the queue with their priority.

Grading:

- Correctness 60%
- Algorithms Design (30%)
- Pair work (10%)