5 Queue with Maximum Element — 60 points

Consider the abstract data type MAXQUEUE that maintains a sequence S of integers and supports the following three operations:

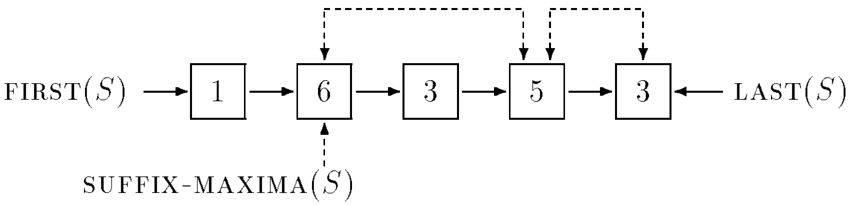
DEQUEUE(S) removes the first element of S and returns its value.

ENQUEUE(S, x) appends the integer x to the end of S.

MAXIMUM(S) returns the largest integer in S, but does not delete it.

An element x is a *suffix maximum* in a sequence if all elements that occur after x in the sequence are smaller in value. In other words, when all elements closer to the beginning of the sequence are dequeued, x is the unique maximum in the resulting sequence. For example, in the sequence 1,6,3,5,3, the suffix maxima are the second, fourth, and fifth elements.

One way to implement the MAXQUEUE abstract data type is to use a singly linked list to represent the sequence, with additional pointers to the first and last elements of that list, and to have a doubly linked list of the suffix maxima (arranged in the same order as they are in the sequence). For example, the sequence $S = 1, 6, 3, 5, 3$ would be represented as follows:



Note that the last element of the list is always the last element of the suffix maxima list. Note that the suffix maxima list is sorted in decreasing order.

a) [4 points] What does the data structure look like after performing ENQUEUE(S,4) in the sequence $S = 1, 6, 3, 5, 3$.

b) [8 points] Explain how to perform the three operations:

DEQUEUE

ENQUEUE

MAXIMUM

c) [8 points] What is the worst case asymptotic time complexity of each of your algorithms?

DEQUEUE

ENQUEUE

MAXIMUM

... continued

d) [20 points] Prove that the amortised time complexity of DEQUEUE, ENQUEUE, and MAXIMUM are all $O(1)$. Hint: Define a potential function based on the number of suffix maxima in the sequence.

... continued

e) [20 points] Describe a data structure for the MAXQUEUE abstract data type that performs all three operations DEQUEUE, ENQUEUE, and MAXIMUM in worst case $O(\log n)$ time. Explain briefly why your algorithms are correct and why they achieve the desired time bound.

To the extent that your solution is based upon data structures described in the course, you can simply explain how to use or modify them in your implementation without repeating details of such data structures.

... continued