York University CSE 4101/5101 April 21, 2009

Homework Assignment #6
Due: April 30, 4:00 p.m.

1. In each of the following parts, you are required to design an algorithm that uses stacks to implement other ADTs. Your implementations should not use any other auxiliary data structures besides the stacks. Assume each operation on an ordinary stack takes $O(1)$ time.

For each part, you should briefly explain why your algorithm is correct (state loop invariants and data structure invariants as needed). You should also explain carefully why it achieves the required running times.

(a) A superstack is like an ordinary stack, but provides one additional operation: MAX(), which returns the maximum element that is currently stored in the stack without changing the state of the stack.

Describe how to implement a superstack using two ordinary stacks so that the worst-case time for all three operations (PUSH, POP and MAX) is $O(1)$.

Hint: use one stack to store all the data items and use the other to help you answer the MAX queries.

(b) Describe how to implement a queue from two (ordinary) stacks so that the amortized time for each queue operation (ENQUEUE and DEQUEUE) is $O(1)$.

(c) A superqueue is like an ordinary queue, but provides one additional operation: MAX(), which returns the maximum element that is currently stored in the queue without changing the state of the queue.

Describe how to implement a superqueue using ordinary stacks so that the amortized time for all three operations (ENQUEUE, DEQUEUE and MAX) is $O(1)$.