TEL AVIV UNIVERSITY
Department of Computer Science
0368.4281 – Advanced topics in data structures
Spring Semester, 2003/2004

Homework 0, March 15, 2003

Due on March 29. Please keep a copy of your homework ! Hand in your homework via email or to me personally, make sure I mark up that you handed it in. You may be able to find some of the solutions (or ideas leading to them) in the literature, but I think it is more fun to first try by yourself. The questions are not very hard and sometimes there may be more than one interesting solution.

1. A double ended queue (deque) is a data structure which supports insertions and deletions of items from both sides. The operations push and pop, insert and delete an item, respectively, from one side of the deque. The operations inject and eject, insert and delete an item, respectively, from the other side of the deque. Describe an implementation of a deque via stacks (where a stack supports only push and pop). Simulate each operation on the deque by a sequence of operations on the two stacks. Your implementation should require $O(1)$ amortized time per operation and $O(n)$ space. Prove that indeed your data structure achieves this amortized time bound.

2. We maintain items in an array subject to the following operations.

   1. insert(i): insert item i in the next free slot of the array.
   2. delete: delete an item from the last occupied slot of the array.

Describe an implementation for these operations such that each operation takes $O(1)$ amortized time and the length of the array at any time is proportional to the number of occupied slots. (I.e. there should be constants c_1 and c_2 such that $c_1 n \leq L \leq c_2 n$ at any time where L is the length of the array and n is the number of occupied slots.) Prove that your implementation indeed achieves these time bounds.