# Data Structures Homework 1

**Submission instructions**

You don't need to submit this homework. TAs will check and grade it during the lab hours.

**Deadline**: You must finish the homework by Monday, April 12. TAs will evaluate it in the lab.

## Programming Exercises

1. **(Weiss Q. 3.28)** A deque is a data structure consisting of a list of items, on which the following operations are possible:

   - `push(x)`: Insert item `x` on the front end of the deque.
   - `pop()`: Remove the front item from the deque and return it.
   - `inject(x)`: Insert item `x` on the rear end of the deque.
   - `eject()`: Remove the rear item from the deque and return

   it. Each of these operations should execute in O(1) time.

   Name your deque class `MyDeque` and use the exact method names described above. Your class should be generic and may use either the built in Java `ArrayList/LinkedList` classes or the textbook's `MyArrayList/MyLinkedList` implementations. ($H_{INT}$ : you should only need one of these.) Alternatively, you can also build this data structure from scratch, without piggybacking on these other classes. Test your class using the class `MyDequeTester`, which is in the assignment folder.

2. Palindrome Detector. A palindrome is a phrase that reads the same forwards as it does backwards. For example, "a man, a plan, a canal, Panama," is a palindrome. Write a program that uses a stack to check for palindromes in each line of a text file. Try your program on the example text file,

   `palindromes.txt`

   Your program should output the palindromes that it finds in the document. For example:

   ```
   java PalindromeFinder palindromes.txt
   "a man, a plan, a canal, Panama" is a
   palindrome. "Don't nod" is a palindrome.
   "Taco Cat!" is a palindrome.
   ...
   ```

   You may write a MyStack class for this problem, or you may use your MyDeque class since it supports the two stack methods. Don't use the built in Stack. Feel free to use either a LinkedList, ArrayList or an array to implement MyStack.