TTIT33 Lab1: Stacks, Queues and Deques

In this lab we exercise the implementation and use of the abstract data types (ADT) Stack and Queue. As an application we write a program to figure out whether a string is a palindrome. A palindrome is a string that reads exactly the same backwards as forwards. For instance, “a”, “abba” and “abcbca” are palindromes, while “abc” and “abab” are not. We write a program that checks whether a string is a palindrome and uses stacks and queues.

We also learn how to define a new ADT and provide an implementation. In this lab we define the ADT Deque, provide an array-based implementation for ADT Deque and use the ADT in a program for recognizing palindroms.

1 Palindrom Recognition using Stacks and Queues

Copy the code found in /home/TTIT33/kursdir/Lab to your own directory.

Write a program that recognizes whether a string is a palindrome. Use the template in Palindrom.java. Use stacks and queues to find a solution to the problem. The ADTs Stack and Queue are defined in the directory DataStructures. You may use the array or the list implementations of ADTs Stack and Queue.

2 Deques

A deque is a sequence where elements can be added and removed at both ends (front and rear) of the sequence. The operations that are defined on deques are

- addFront( element ), inserts element at the front
- addRear( element ), inserts element at the rear
- getFront, returns the item at the front
- getRear, returns the item at the rear
- removeFront, returns and removes the item at the front
- removeRear, returns and removes the item at the rear
- isEmpty, returns true if the deque is empty, false otherwise
- makeEmpty, makes an empty deque

In DataStructures/Deque.java an interface class is defined that defines the ADT Deque.
1. Implement the ADT Deque using an array-based implementation. Use the template given in DataStructures/DequeAr.java.

2. What is the time complexity for the operations?

3 **Palindrom Recognition using Deques**

   1. Write a program that recognizes whether a string is a palindrom. Use the template in Palindrom2.java. Use a deque to find a solution to the problem.

   2. What is the time complexity of your palindrom recognition algorithm?

4 **Examination**

   Hand in the code together with a test run for task 1, task 2.1 and task 3.1. Hand in a calculation of the time complexities for tasks 2.2 and 3.2.