German University in Cairo
Computer Science Department
Prof. Dr. Slim Abdennadher

Data Structures and Algorithms, Winter Term 2007-2008
Practice Assignment 4

Discussion: 17.11.2007 - 22.11.2007

Exercise 4-1 To be discussed in the Tutorial
Postfix Evaluation

Write a class `Postfix.java` that allows for the evaluation of postfix expressions using a static method `int evaluate(String s)`. Your `evaluate` method will be passed a string representing a postfix expression containing space-seperated elements (“2 3 4 + *”,etc.) and is required to evaluate the expression and return the resulting value. The valid set of characters is {1, 2, 3, 4, 5, 6, 7, 8, 9, +, -, *, / }. Make use of a stack to solve this problem.

The postfix evaluation algorithm can be described as follows:

a) Initialize an empty stack.

b) Read the postfix expression from left to right.

c) If the character is an operand, push it onto the stack.

d) If the character is an operator, pop two operands, perform the appropriate operation, and then push the result onto the stack.

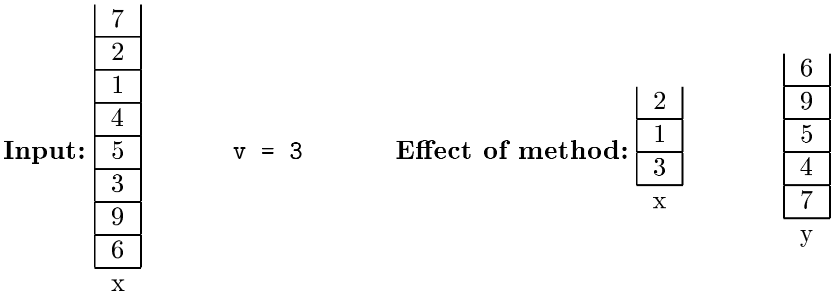
e) At the end of the postfix expression, pop the result from the stack.

Exercise 4-2 To be discussed in the Tutorial
Split a Stack

You are required to implement a method `static Stack split(Stack x, int v)` where `x` is a stack of `int`s. The method is required to split the values stored in the stack `x` into two groups:

- values less than or equal to `v` and store them in the stack `x`
- values greater than `v` and store them in a new stack, say `y`

The method should finally return the newly created stack `y` containing the values greater than `v`. The following is a sample method run:



Only stacks can be used!.

Exercise 4-3 To be discussed in the Tutorial
Set Datatype using Arrays

Implement a class `Set` as described below. You are required to use arrays for storage and manipulation of data.

- The class `Set` is used for organizing objects of type `Integer`. You should provide implementations for all the methods described below and specify the complexity of each of your implementations in Big O notation. **A `Set` is not allowed to contain duplicates**. You can assume that a `Set` contains a maximum of 100 elements.
- `Set` operations:
  - Construct an empty `Set`

    `public Set()`
  - Return the number of elements in the `Set`

    `public int cardinality()`
  - Insert an `Integer` element into the `Set`

    `public void insert(Integer element)`
  - Remove a given `Integer` value from the `Set`

    `public void remove(Integer element)`
  - Check if the `Set` contains a given `Integer` value: return `true` if it is, return `false` otherwise.

    `public boolean contains(Integer element)`
  - Replace a given `Integer x` with another given `Integer y`. If `x` is not in the `Set` then you should just insert `y`.

    `void replace(Integer x, Integer y)`
  - Check if the `Set` is empty: return `true` if it is, return `false` otherwise.

    `public boolean isEmpty()`

Exercise 4-4 To be discussed in the Lab
Palindrome using Stacks

You are required to create a class that can detect palindromes. A palindrome is a word or sentence that spells the same forwards as backwards. For example,

- Mom
- Radar
- Race car
- Dennis sinned
- Murder for a jar of red rum
- Are we not drawn onward we few drawn onward to new era

There are many ways to detect if a phrase is a palindrome. The method that you will use in this assignment is by using two stacks. This works in the following way: Push half of the characters into one of the stacks and then compare the characters as you pop them from the stacks.

Note: You must ignore case when determining if the word is a palindrome or not. Thus, it does not matter at all if some of the characters are capitalized and others not. Furthermore, whitespace (spaces) also does not matter, so you must ignore any spaces.

Exercise 4-5 To be discussed in the Lab

Duplicating Elements of a Stack

Write a Java method `createDuplicates` that takes 2 stacks of equal size as input. The first one is a stack of strings and the second one is a stack of integers. The method should return a new stack of strings where each element of the original string stack is repeated the same number of times specified in the corresponding cell in the integer array.

For example:

`"Hend"`
`"Abdellatif"`
`"Slim"`

Stack of Strings

1
3
2

Stack of integers

`"Slim"`
`"Slim"`
`"Abdellatif"`
`"Abdellatif"`
`"Abdellatif"`
`"Hend"`

Result

Exercise 4-6 To be discussed in the Lab

Average GPA

Given a stack of `Student` objects you are required to implement a method that calculates the average `GPA` of some students given in a stack and returns this average value.