**LAB7**

**Jimmy Palma**

**610756**

1. **[Interview Question] Devise an O(n) algorithm to accomplish this task:**  
    **Given a none-empty string S of length n, S consists some words separated by spaces. We want to reverse every word in S.**  
    **For example, given S = “we test coders”, your algorithm is going to return a string with every word in S reversed and separated by spaces. So the result for the above example would be “ew tset sredoc”.**

Algorithm reverse

Input: string S

Output: string reversed

Stack <- new stack

S1 <- new string

for words in S do

Stack <- insertWord(word)

While(¬ Stack.isEmpty()) do

S1 <- Stack.pop

S1 <- insertSpace

1. **Create a sorting routine based on a BST and place it in the sorting environment, distributed earlier. For this, your new class, BSTSort, should be a subclass of Sorter. Your BSTSort class can be essentially the same as the BST class given in the slides (see the folder in your labs directory for this lab), except that you will need to modify the printTree method so that it outputs values to an array (rather than printing to console).**

**After you have implemented, discuss the asymptotic running time of your new sorting algorithm. Run an empirical test in the sorting environment and explain where BSTSort fits in with the other sorting routines (which algorithms is it faster than? which is it slower than?).**

70 ms -> MergeSortPlus

79 ms -> MergeSort

157 ms -> MyBST

46198 ms -> BubbleSort

1. **For each integer n = 1, 2, 3,..., 7, determine whether there exists a red-black tree having exactly n nodes, with all of them black. Fill out the chart below to tabulate the results:**

|  |  |
| --- | --- |
| **Nun Nodes n** | **Does there exist a red-black tree with n nodes, all of which are black?** |
| 1 | **Yes** |
| **2** | **no** |
| **3** | **yes** |
| **4** | **no** |
| **5** | **no** |
| **6** | **no** |
| **7** | **yes** |

1. **For each integer n = 1,2,3,..., 7, determine whether there exists a red-black tree having exactly n nodes, where exactly one of the nodes is red. Fill out the chart below to tabulate the results:**

|  |  |
| --- | --- |
| **Nun Nodes n** | **Does there exist a red-black tree with n nodes, where exactly one of the nodes is red?** |
| 1 | **no** |
| **2** | **yes** |
| **3** | **yes** |
| **4** | **no** |
| **5** | **yes** |
| **6** | **no** |
| **7** | **yes** |