King Saud University	College of Computer and Information Sciences Department of Computer Science
Data Structures CSC 212	**Midterm Exam - Spring 2018**
Date: 22/03/2018	Duration: 90 minutes

Guidelines

•No calculators or any other electronic devices are allowed in this exam.

Student ID:	Name:
Section:	Instructor:

1	2.1	2.2	3.1	3.2	Total

Question 1 ..... 30 points

1. Trace the execution of the following expression: **2 9 3 1 + * 5 4 3 % 1 - - + > 35 14 8 + = ||**. Show the content of the data structure(s) **after** parsing each operation.

+	*	%	-	-
+	>	+	=	||

2. Trace the execution of the following expression: **4 + (9 -(3 * 2)) % 3 + 5 * (2 +(6 / 3)) -1**. Draw the content of the data structure(s) after parsing each operation.

+	-	*	%	+
*	+	/	-	$

3. Indicate the **preorder**, **inorder** and **postorder** traversals of the tree shown in **Figure 1** (choose answer from the table below).

   Preorder: ..... Inorder: ..... Postorder: .....

Figure 1: Binary Tree.

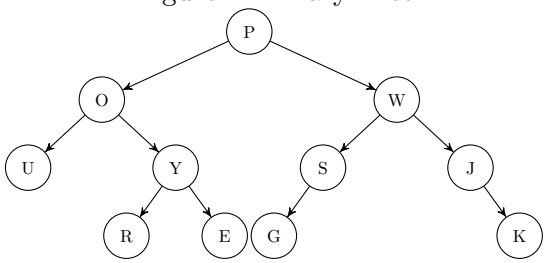
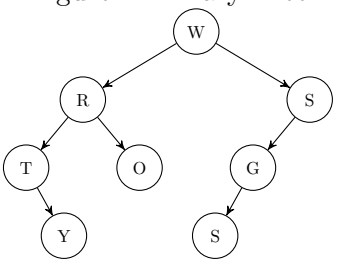


Figure 2: Binary Tree.



1.	P O W U Y S J R E G K	**5.**	U O Y R E P G S W K J	**9.**	U Y R E O G S W K J P
2.	P O U Y R E W S G J K	**6.**	O U Y R E P S G W K J	**10.**	R E G K U Y S J O W P
3.	P O U R E Y W S G J K	**7.**	U O R Y E P G S W K J	**11.**	U R E Y O G S K J W P
4.	P O U Y R E W S J G K	**8.**	U O R Y E P G S W J K	**12.**	U Y R E O G S K J W P

Question 2 . . . . . 35 points

1. Write the method **public static** `<T>` **void** `swapAdj(Queue<T> q)` which swaps adjacent elements in the queue starting from the first element. Do not use any extra data structures.

   Example 1. *If* $q : A, B, C, D, E$*, then after calling* `swapAdj(q)`*,* q *becomes* B, A, D, C, E*.*

2. Write the method **public static** `Stack<Character> replace(Stack<Character> st, char a, char b)`, which replaces all occurrences of the char `a` in the stack `st` by the char `b` and returns the result as a new stack. The stack `st` must not change after the call.

   Example 2. *If* `st` *before the call contains:* `'A', 'B','C', 'A', 'E'` *(from top to bottom), and we called:* `replace(st, 'A', 'B')`*, then the returned stack contains:* `'B', 'B', 'C', 'B', 'E'`*, and* `st` *remains unchanged.*

Question 3 . . . . . 35 points

1. As a member of the class `LinkedList`, write the method **`public void insertBefore(T e, int i)`** that inserts the element e before the `ith` element. The numbering starts from 0. Assume that `i` is a valid index. Do not call any methods of the class `LinkedList`. Do not use any auxiliary data structures.

   Example 3. *If $l : A \to B \to C \to D \to E$, then calling `l.insertBefore('N', 4)` changes the list to $l : A \to B \to C \to D \to N \to E$. Calling `l.sublist('N', 0)`, changes the list to $l : N \to A \to B \to C \to D \to E$.*

2. Write the **recursive** method **boolean** `notRightChild(T e)`, member of the class `BT` (binary tree) which returns true if and only if `e` does not appear as a right child of another node in the tree.

   Example 4. *In the tree shown in Figure 2,* `notRightChild('T')` *returns true,* `notRightChild('S')` *returns false,* `notRightChild('Y')` *returns false,* `notRightChild('W')` *returns true and* `notRightChild('X')` *returns true.*