

Student ID: _____

Student Name: _____

CS203 Data Structure and Algorithm Analysis**Quiz 2**

Note 1: Write all your solutions in the question paper directly. You can ask additional answer paper if necessary

Note 2: If a question asks you to design an algorithm, you should **only need to** describe your ideas in general words (e.g., step 1, step 2, ..., step 3).

Problem 1 [20 points] Postfix Expression Evaluation

Given postfix expression 5 9 3 + 4 2 * * 7 + *, please fulfill the following table to show your steps.

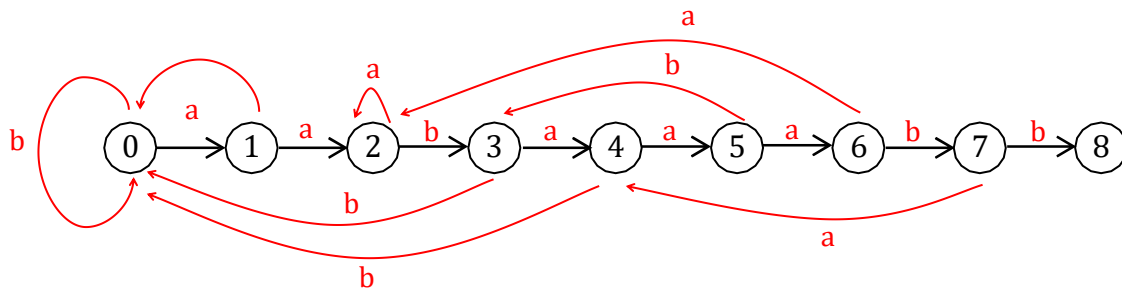
Stack operations:

Push(x): Add an item x to the top of stack.

Pop(): Remove and return the item at the top of stack.

Stack operations	Stack elements
Push(5)	5
Push(9)	5 9
Push(3)	5 9 3
Push(Pop()+Pop())	5 12
Push(4)	5 12 4
Push(2)	5 12 4 2
Push(Pop()*Pop())	5 12 8
Push(Pop()*Pop())	5 96
Push(7)	5 96 7
Push(Pop()+Pop())	5 103
Push(Pop()*Pop())	515

Problem 2 [10 points] Draw FSA graph for pattern string “aabaabb”.



Problem 3 [15 points] Heap sort: a sorted array is created by repeatedly removing the root of the min-heap until the min-heap becomes empty. Given an array-based min-heap (as follows), fulfill the following table to show the elements in the min-heap during heap sort progress.

1	6	15	17	8	54	23	93	39	52	26	79
6	8	15	17	26	54	23	93	39	52	79	
8	17	15	39	26	54	23	93	79	52		
15	17	23	39	26	54	52	93	79			
17	26	23	39	79	54	52	93				
23	26	52	39	79	54	93					
26	39	52	93	79	54						
39	54	52	93	79							
52	54	79	93								
54	93	79									
79	93										
93											

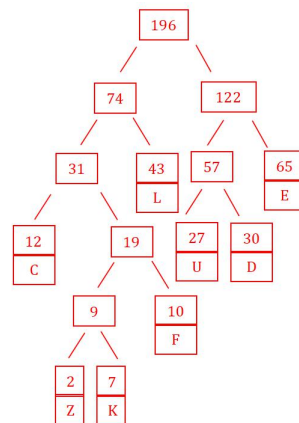
Problem 4 [25 points] Binary Tree

(a) [10 points] Build the Huffman tree for the following (character, frequency) pairs:

Z	K	F	C	U	D	L	E
2	7	10	12	27	30	43	65

One possible answer:

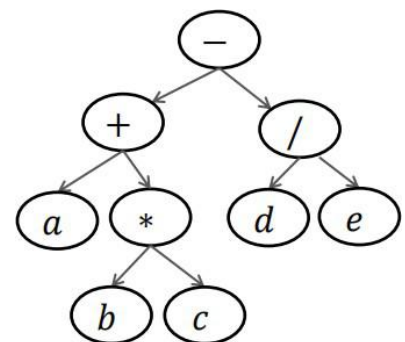
Others can be obtained by swapping left child and right child of the same node.



(b) [5 points] According to Huffman tree of (a), write down the Huffman code of string "LUCK": 011000000101 (One possible answer).

(c) [10 points] Given you the following binary tree of an algebraic expression. First, please write down the post-fix expression of it: abc*+de/-. Second, considering an algebraic expression in a binary tree form, please design an algorithm to find its equivalent postfix expression. The input of your algorithm is the root node **R** of the binary tree.

Post-order traversal starting from root node R.

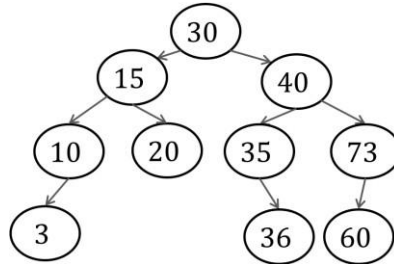


Problem 5 [40 points] Binary Search Tree

(a) [15 points] Fill in the blank of the node-deletion process

Given a binary search tree T with height h . Suppose that we want to delete a node with key e .

1. Find the node u whose key equals to e .
2. **Case 1:** if u is a leaf node:
3. remove u from tree T
4. **Case 2:** if u has a right subtree:
5. node $v \leftarrow$ predecessor¹/successor² (predecessor/successor) of e
6. $u.key \leftarrow v.key$
7. **Case 2.1:** if v is a leaf node:
8. remove v from tree T
9. **Case 2.2:** else:
10. replace v by subtree which rooted at v 's left child¹/ v 's right child²
11. **Case 3:** if u has no right subtree:
12. replace u by subtree which rooted at u 's left child

(b) [5 points] The time complexity of the above node-deletion operation is $O(h)$ (Big-O notation).(c) [20 points] Draw the tree after deletion in each **independent** case**Binary Search Tree T**

- ① Given binary search tree T , draw the tree after deleting node with key $e = 60$
- ② Given binary search tree T , draw the tree after deleting node with key $e = 40$
- ③ Given binary search tree T , draw the tree after deleting node with key $e = 30$
- ④ Given binary search tree T , draw the tree after deleting node with key $e = 73$

