Result & Analysis

Student: MODI BHUMIL DHA... Test: PREP_DSA_AS_Stacks Course: 2023 Data Structure...

Attempt 1

IP Address: 103.53.73.17 **Tab switches:** 37 **OS used:** Windows **Browser used:** Chrome

Test Duration: 02:25:42 **Test Start Time:** Mar 29, 2022 | 10:18 AM

Test Submit Time: Mar 29, 2022 | 06:47 PM **Resume Count:** 2

Overall score

Rank: NA

Topper score: 79.00 / 79

Average score: 18.95 / 79

Least score: 0.00 / 79

MCQ

Rank: NA

Topper score: 9.00 / 9

Average score: 6.17 / 9

Least score: 0.00 / 9

Coding

Rank: NA

Topper score: 70.00 / 70

Average score: 16.96 / 70

Least score: 0.00 / 70

Overall Question Status

Total Questions: 16

Questions Attempted: 16

Questions Correct: 14

Question Wrong: 0

Partially Correct: 2

Question Not Viewed: 0

MCQ - Question Status

Total Questions: 9

Questions Attempted: 9

Questions Correct: 9

Question Wrong: 0

Partially Correct: 0

Question Not Viewed: 0

Coding - Question Status

Total Questions: 7

Questions Attempted: 7

Questions Correct: 5

Question Wrong: 0
Partially Correct: 2
Question Not Viewed: 0

Topic wise Analysis MCQ Coding

Question No: 1 **Multi Choice Type Question**

'Array implementation of Stack is not dynamic', which of the following statements supports this argument?

- space allocation for array is fixed and cannot be changed during run-time CORRECT
- user unable to give the input for stack operations
- a runtime exception halts execution
- All of these

Recommended Learning Content: Introduction to Stacks Stacks Implementation Stacks

Status: Correct **Mark obtained:** 1/1 **Hints used:** 0 **Level:** Medium
Question type: MCQ Single Correct **Subject:** Programming **Subject:** Data Structures
Subject: Stacks

Question No: 2 **Multi Choice Type Question**

A linear list of elements in which insertion and deletion is done at one end is known as

- Queue
- Stack CORRECT
- Tree

- Linked list

Recommended Learning Content: Deletion in Red-Black Tree
Red-Black Tree Introduction AVL Tree Insertion in Red-Black Tree

Status: Correct **Mark obtained:** 1/1 **Hints used:** 0 **Level:** Easy
Question type: MCQ Single Correct **Subject:** Programming **Subject:** Data Structures
Subject: Tree traversal

Question No: 3 **Multi Choice Type Question**

A single array A[1..MAXSIZE] is used to implement two stacks. The two stacks grow from opposite ends of the array. Variables top1 and top2 (topl< top 2) point to the location of the topmost element in each of the stacks. If the space is to be used efficiently, the condition for “stack full” is

- (top1 = MAXSIZE/2) and (top2 = MAXSIZE/2+1)
- top1 + top2 = MAXSIZE
- (top1= MAXSIZE/2) or (top2 = MAXSIZE)
- top1= top2 -1 CORRECT

Recommended Learning Content: Introduction to Stacks Stacks Implementation
Stacks

Status: Correct **Mark obtained:** 1/1 **Hints used:** 0 **Level:** Hard
Question type: MCQ Single Correct **Subject:** Programming **Subject:** Data Structures
Subject: Stacks

Question No: 4 **Multi Choice Type Question**

Assume that the operators +, -, × are left associative and ^ is right associative. The order of precedence (from highest to lowest) is ^, x , +, -. The postfix expression corresponding to the infix expression a + b × c - d ^ e ^ f is

- abc × + def ^ ^ - CORRECT
- abc × + de ^ f ^ -
- ab + c × d - e ^ f ^
- - + a × bc ^ ^ def

Recommended Learning Content: Introduction to Stacks Stacks Implementation Stacks

Status: Correct **Mark obtained:** 1/1 **Hints used:** 0 **Level:** Medium
Question type: MCQ Single Correct **Subject:** Programming **Subject:** Data Structures
Subject: Stacks

Question No: 5 **Multi Choice Type Question**

Balanced parenthesis is an application of

- Queue
- Array
- Stack CORRECT
- Tree

Recommended Learning Content: Introduction to Stacks Stacks Implementation Stacks

Status: Correct **Mark obtained:** 1/1 **Hints used:** 0 **Level:** Easy
Question type: MCQ Single Correct **Subject:** Programming **Subject:** Data Structures
Subject: Stacks

Question No: 6 **Multi Choice Type Question**

Convert the following infix expression into its equivalent post fix expression (A + B^ D) / (E – F) + G

- ABD^ + EF – / G+ CORRECT
- ABD + ^EF – / G+
- ABD + ^EF / – G+
- ABD^ + EF / – G+

Recommended Learning Content: Introduction to Stacks Stacks Implementation Stacks

Status: Correct **Mark obtained:** 1/1 **Hints used:** 0 **Level:** Medium
Question type: MCQ Single Correct **Subject:** Programming **Subject:** Data Structures
Subject: Stacks

Question No: 7 **Multi Choice Type Question**

Following is an incorrect pseudocode for the algorithm which is supposed to determine whether a sequence of parentheses is balanced:

```
declare a character stack
while ( more input is available)
{
  read a character
  if ( the character is a '(' )
   push it on the stack
  else if ( the character is a ')' and the stack is not empty )
   pop a character off the stack
  else
   print "unbalanced" and exit
 }
 print "balanced"
```

Which of these unbalanced sequences does the above code think is balanced?

- ((()) CORRECT
- ())(()

- ((()()))
- (()))()

Recommended Learning Content: Introduction to Stacks Stacks Implementation Stacks

Status: Correct **Mark obtained:** 1/1 **Hints used:** 0 **Level:** Medium
Question type: MCQ Single Correct **Subject:** Programming **Subject:** Data Structures
Subject: Stacks

Question No: 8 **Multi Choice Type Question**

Stack can be implemented using

- Array and Binary Tree
- Linked List and Graph
- Array and Linked List CORRECT
- Queue and Graph

Recommended Learning Content: Introduction to Stacks Stacks Implementation Stacks

Status: Correct **Mark obtained:** 1/1 **Hints used:** 0 **Level:** Easy
Question type: MCQ Single Correct **Subject:** Programming **Subject:** Data Structures
Subject: Stacks

Question No: 9 **Multi Choice Type Question**

Suppose a stack is to be implemented with a linked list instead of an array. What would be the effect on the time complexity of the push and pop operations of the stack implemented using linked list (Assuming stack is implemented efficiently)?

- O(1) for insertion and O(n) for deletion
- O(1) for insertion and O(1) for deletion CORRECT
- O(n) for insertion and O(1) for deletion
- O(n) for insertion and O(1) for deletion

Recommended Learning Content: Introduction to Stacks Stacks Implementation Stacks

Status: Correct **Mark obtained:** 1/1 **Hints used:** 0 **Level:** Medium
Question type: MCQ Single Correct **Subject:** Programming **Subject:** Data Structures
Subject: Stacks