

5/5/2020

Review Test Submission: Quiz 4 – 35191.1202

Syracuse University

Blackboard @ SU

Courses

Organiz

Doung Lan Cheung

Home

Assignments

Quiz

Review Test Submission: Quiz 4

Review Test Submission: Quiz 4

User	Doung Lan Cheung
Course	CIS.351.M001.SPRING20.Data Structures
Test	Quiz 4
Started	3/23/20 3:50 PM
Submitted	3/23/20 3:51 PM
Due Date	3/28/20 11:59 PM
Status	Completed
Attempt Score	100 out of 100 points
Time Elapsed	0 minute out of 25 minutes
Results Displayed	All Answers, Submitted Answers, Correct Answers, Feedback, Incorrectly Answered Questions

Question 1

10 out of 10 points

Given a 5 element queue Q (from front to back: 1, 3, 5, 7, 9), and an empty stack S, remove the elements one-by-one from Q and insert them into S, then remove them one-by-one from S and re-insert them into Q.

Select, what are the contents of the queue now look like (from front to back)?

Selected Answer:

9, 7, 5, 3, 1

Answers: 9, 7, 5, 1, 3

9, 7, 5, 3, 1

1, 7, 5, 3, 9

9, 5, 3, 7, 1

Question 2

10 out of 10 points

Which of the following statements about stacks is incorrect?

Selected Answer:

Stacks are first-in, first-out (FIFO) data structures.

Answers:

https://blackboard.syracuse.edu/webapps/assessment/review/review.jsp?attempt\_id=\_22632626\_1&course\_id=\_426824\_1&content\_id=\_6061298\_1&... 1/5

5/5/2020

Review Test Submission: Quiz 4 – 35191.1202

Stacks can be implemented using linked lists.

Stacks are first-in, first-out (FIFO) data structures.

New nodes can only be added to the top of the stack.

The last node (at the bottom) of a stack has a null (0) link.

Question 3

10 out of 10 points

Every node (except of the last node) in a singly linked list contains

Selected Answer:

the address of the next node

Answers: the next node

no address information

the address of the next node

the address of the previous node

Question 4

10 out of 10 points

What is the correct postfix version of the infix expression  $A * (B + C * D) + E$

Selected Answer:

ABCD\*\*+E+

Answers: ABCD\*\*+E

ABCD\*\*+E+

ABCD\*\*+E+

ABC\*D\*\*+E+

Question 5

10 out of 10 points

The postfix expression  $14 \ 2 \ 5 \ + \ =$  will generate an error, because



Selected Answer:



there will be too many elements in the stack when the equal sign is encountered

Answers:

it contains an illegal operator

it does not have enough operands

it has too many operators



there will be too many elements in the stack when the equal sign is encountered

### Question 6

5 out of 5 points



How would you access elements of an aggregated object (think of a collection) sequentially without exposing the underlying structure of the object?

Selected Answer:



using an iterator

Answers:

using indexes



using an iterator

using a stack.

using a queue

### Question 7

10 out of 10 points



Which of the following is true about linked list implementation of queue?

Selected Answer:



Both of the above

Answers:

In push operation, if new nodes are inserted at the beginning of linked list, then in pop operation, nodes must be removed from end.

In push operation, if new nodes are inserted at the end, then in pop operation, nodes must be removed from the beginning.



Both of the above

None of the above

Response Feedback:

To keep the First In First Out order, a queue can be implemented using linked list in any of the given two ways.

### Question 8

10 out of 10 points



You have a singly linked list constructed out of nodes defined as follows:

```
public class Node
{
    public int datum;
    public Node next;
}
```

In all of functions shown below, the parameter first refers to the first node in the linked list, if there is one, and has the value null otherwise. Which of the following functions correctly inserts a value x at the front of the linked list and returns a reference to the new front of the linked list?

Selected Answer:

```
public Node insertFront(Node first, int x)
{
    Node n = new Node();
    n.datum = x;
    n.next = first; return n;
}
```



Answers:

```
public Node insertFront(Node first, int x)
{
    first = new Node();
    first.datum = x;
    first.next = first;
    return first;
}
```

```
public Node insertFront(Node first, int x)
{
    Node n = new Node();
    n.datum = x;
    n.next = first; return n;
}
```



Both of the above

None of the above

Response Feedback: Only function II works correctly. Function I loses the reference to the front of the list when it executes the statement: first = new Node(); Because there is no other reference to the front of the list, assigning this new value to first causes the pre-existing nodes in the list to be lost. Function II avoids this problem by using a separate variable n for the new node, and it sets n's next field to the old front of the list. It then returns the value of n, which is a reference to the new front of the list.

### Question 9

10 out of 10 points



What is the postfix version of the infix expression  $12 / 3 * (50 / 3 * (3 / 4)) / 2 * 10$

Selected Answer:

Answers:

☒ 12 3 / 50 3 / 3 4 / \* \* 2 / 10 \*☐ 12 3 / 50 3 / 3 4 / \* 2 \* / 10 \*☐ 12 3 / 50 3 / 3 4 / 2 \* \* / 10 \*☐ 12 3 / 50 3 / 3 4 / \* 2 \* / 10 \*☒ 12 3 / 50 3 / 3 4 / \* \* 2 / 10 \***Question 10**

5 out of 5 points



The stack data type is restrictive in a sense that you cannot

Selected Answer: ☒ remove the bottom elementAnswers: ☐ remove the top element☐ insert at the top☐ look at the top elemen☒ remove the bottom element**Question 11**

10 out of 10 points



What is the reason for using a "circular queue" instead of a regular one?

Selected Answer: ☒ reuse empty spacesAnswers: ☐ running time of enqueue() is improved☒ reuse empty spaces☐ you can traverse all the elements more efficiently☐ none of the above

Tuesday, May 5, 2020 1:03:45 AM EDT

← OK