Take Test: Quiz #1

Test Information

Description

King Saud University

College of Computer and Information Sciences

Data Structures CSC 212

Quiz 1 - Fall 2020

Date: 30 / 9 / 2020

Duration: 60 minutes

Instructions	
Timed Test	This test has a time limit of 1 hour and 10 minutes.This test will save and submit automatically when the time expires. Warnings appear when **half the time**, **5 minutes**, **1 minute**, and **30 seconds** remain.
Multiple Attempts	Not allowed. This test can only be taken once.
Force Completion	This test can be saved and resumed at any point until time has expired. The timer will continue to run if you leave the test.

Remaining Time: 07 minutes, 13 seconds.

Question Completion Status:

1 2 3 4 5 6 7 8 9 10 11 12 13 14 15 16 1
25 26 27 28 29 30 31 32 33

- ○ 2
- ◉ 1
- ○ None

QUESTION 2

1.66667 points Saved

```
1 int sum = 0;
2 for (int i = n * n; i >= n + 1; i--)
3   sum++;
4 for (int i = 1; i < n * n; i++) {
5   sum -= i; int j = 1;
6   while (j < i)
7     sum += j++; }
8 System.out.println(sum);
```

Line 2

- ○ n^2
- ○ $n^2 + 1$
- ◉ $n^2 - n + 1$
- ○ $n^2 - n$
- ○ None

QUESTION 3

1.66667 points Saved

```
1 int sum = 0;
2 for (int i = n * n; i >= n + 1; i--)
3   sum++;
4 for (int i = 1; i < n * n; i++) {
5   sum -= i; int j = 1;
6   while (j < i)
7     sum += j++; }
8 System.out.println(sum);
```

Line 3

- ○ $n^2 - 1$
- ◉ $n^2 - n$

Remaining Time: 07 minutes, 13 seconds.

Question Completion Status:

1 2 3 4 5 6 7 8 9 10 11 12 13 14 15 16 1
25 26 27 28 29 30 31 32 33

```
1  int sum = 0;
2  for (int i = n * n; i >= n + 1; i--)
3    sum++;
4  for (int i = 1; i < n * n; i++) {
5    sum -= i; int j = 1;
6    while (j < i)
7      sum += j++; }
8  System.out.println(sum);
```

Line 4

- ○ $n^2 + 1$
- ○ $n^2 - 1$
- ○ 2
- ○ n + 1
- ◉ None

QUESTION 5

1.66667 points Saved

```
1  int sum = 0;
2  for (int i = n * n; i >= n + 1; i--)
3    sum++;
4  for (int i = 1; i < n * n; i++) {
5    sum -= i; int j = 1;
6    while (j < i)
7      sum += j++; }
8  System.out.println(sum);
```

Line 5

- ○ $n^2 + 1$
- ○ $n^2 - 1$
- ○ $n^2 + 2$
- ○ n + 1
- ◉ None

QUESTION 6

1.66666 points Saved

```
1  int sum = 0;
```

Remaining Time: **07 minutes, 13 seconds.**

Question Completion Status:

1 2 3 4 5 6 7 8 9 10 11 12 13 14 15 16 1
25 26 27 28 29 30 31 32 33

QUESTION 7

1.66667 points Saved

```
1 int sum = 0;
2 for (int i = n * n; i >= n + 1; i--)
3   sum++;
4 for (int i = 1; i < n * n; i++) {
5   sum -= i; int j = 1;
6   while (j < i)
7     sum += j++; }
8 System.out.println(sum);
```

Line 7

- ○ $n^2 (n^2 - 2) / 2$
- ◉ $n^2 (n^2 - 1) / 2$
- ○ n (n - 1) / 2
- ○ i - 1
- ○ None

QUESTION 8

1.66666 points Saved

```
1 int sum = 0;
2 for (int i = n * n; i >= n + 1; i--)
3   sum++;
4 for (int i = 1; i < n * n; i++) {
5   sum -= i; int j = 1;
6   while (j < i)
7     sum += j++; }
8 System.out.println(sum);
```

Line 8

- ○ n
- ○ 0
- ○ 2
- ◉ 1
- ○ None

Remaining Time: 07 minutes, 13 seconds.

Question Completion Status:

1 2 3 4 5 6 7 8 9 10 11 12 13 14 15 16 1
25 26 27 28 29 30 31 32 33

- n^2
- n
- None

QUESTION 10 — 2 points — Saved

Choose the most appropriate answer
5n + 2n log n - 11 is

- *O* (2n log (n))
- *O* (5n)
- *O* (n log n)
- *O* (n)
- None

QUESTION 11 — 2 points — Saved

Choose the most appropriate answer
3n + 4 log n^n is

- *O* (n^3)
- *O* (n log n)
- *O* (log n)
- *O* (n)
- None

Remaining Time: 07 minutes, 13 seconds.

Question Completion Status:

1 2 3 4 5 6 7 8 9 10 11 12 13 14 15 16 1
25 26 27 28 29 30 31 32 33

- None

QUESTION 13 — 2 points — Saved

Choose the most appropriate answer

$12\sqrt{n} + 12n \log(2^{4n})$ is

- $O(n^{1/2})$
- $O(n^2)$
- $O(2^n)$
- $O(n\sqrt{n})$

QUESTION 14 — 2 points — Saved

Choose the most appropriate answer

$(n+4) \log 4^{1/n} - (14n - n \log n^2)$ is

- $O(4^{1/n})$
- $O((n+4) \log 4^{1/n})$
- $O(n^2 \log n)$
- $O(n \log n)$

QUESTION 15 — 1.66667 points — Saved

Write the method `public static <T> boolean check (List <T> l1, List`

Remaining Time: 07 minutes, 13 seconds.

Question Completion Status:

- `l1.findFirst();`
- `l2.findFirst();`
- None

QUESTION 16 1.66667 points Saved

Line 2

- `while(!l2.last()) {`
- `while(l1.findNext()) {`
- `while(!l1.last()) {`
- `while(l1.retrieve().equals.(l2.retrieve())) {`
- None

QUESTION 17 1.66667 points Saved

Line 3

- `l2.findFirst();`
- `l1.findNext();`
- `l1.findFirst();`
- `l2.findNext();`
- None

QUESTION 18 1.66666 points Saved

Remaining Time: **07 minutes, 13 seconds.**

Question Completion Status:

1 2 3 4 5 6 7 8 9 10 11 12 13 14 15 16 1
25 26 27 28 29 30 31 32 33

QUESTION 19 **1.66667 points** Saved

Line 5

- (●) `l1.findNext();`
- ( ) `return true; }`
- ( ) `return false; }`
- ( ) `l1.findNext(); l2.findFirst(); }`
- ( ) None

QUESTION 20 **1.66666 points** Saved

Line 6

- ( ) `if(!l1.last() && l1.retrieve().equals(l2.retrieve()))`
- ( ) `if(l1.last() && l1.retrieve().equals(l2.retrieve()))`
- (●) `if(!l1.retrieve().equals(l2.retrieve()))`
- ( ) `if(l2.data.equals(l1.data))`
- ( ) None

QUESTION 21 **1.66667 points** Saved

Line 7

- ( ) `l1.findNext(); l2.findNext(); }`
- (●) `return false;`
- ( ) `return true;`
- ( ) `l1.findNext(); l2.findFirst();`
- ( ) None

Remaining Time: 07 minutes, 13 seconds.

Question Completion Status:

1 2 3 4 5 6 7 8 9 10 11 12 13 14 15 16 1
25 26 27 28 29 30 31 32 33

```
l1.findFirst(); l2.findNext(); }
```

None

QUESTION 23 1.66667 points Saved

Line 9

```
return true;
```

```
l1.findFirst();
```

```
return false;
```

```
l2.findNext();
```

None

QUESTION 24 1 points Saved

Write the method `public static <T> void insertUnique(List <T> l, T e)`, user of the ADT List. The method takes a list `l` and an element `e` and inserts the element at the end of the list only if it is not already there.

Example 0.2. ***If `l` : A → B → C, then after calling `insertUnique(l, "C")`, the list does not change. Calling `insertUnique(l, "D")` will make `l` be : A → B → C → D.***

Line 1

```
l.findFirst();
```

```
while(!l.last()) {
```

```
if(l.head != null) {
```

Remaining Time: 07 minutes, 13 seconds.

Question Completion Status:

1 2 3 4 5 6 7 8 9 10 11 12 13 14 15 16 1
25 26 27 28 29 30 31 32 33

- `while(!l.last()) {`
- `if(l.retrieve().equals(e))`
- None

QUESTION 26 — 1 points — Saved

Line 3

- `l.findNext();`
- `return; }` (selected)
- `while(l.current != null) {`
- `if(l.retrieve().equals(e))`
- None

QUESTION 27 — 1 points — Saved

Line 4

- `l.insert(e);`
- `l.findFirst();` (selected)
- `if(l.current.data.equals(e))`
- `return;`
- None

QUESTION 28 — 1 points — Saved

Remaining Time: 07 minutes, 13 seconds.

Question Completion Status:

1 2 3 4 5 6 7 8 9 10 11 12 13 14 15 16 1
25 26 27 28 29 30 31 32 33

QUESTION 29 — 1 points — Saved

Line 6

- ○ `l.current = l.current.next; }`
- ○ `while(l.findNext())`
- ○ `if(!l.retrieve().equals(e))`
- ◉ `if(l.retrieve().equals(e))`
- ○ None

QUESTION 30 — 1 points — Saved

Line 7

- ◉ `return;`
- ○ `l.current.next = new Node <T>(e); }`
- ○ `l.insert(e);`
- ○ `if(!l.retrieve().equals(e))`
- ○ None

QUESTION 31 — 1 points — Saved

Line 8

- ○ `l.insert(e);`
- ○ `l.findFirst();`
- ○ `else {`
- ◉ `l.findNext(); }`

Remaining Time: **07 minutes, 13 seconds.**

Question Completion Status:

1 2 3 4 5 6 7 8 9 10 11 12 13 14 15 16 1
25 26 27 28 29 30 31 32 33

- `Node<T> n = new Node <T>(e);`
- None

QUESTION 33 — **1 points** — Saved

Line 10

- `l.insert(e);`
- `l.update(e)`
- `return 1;`
- `l.current = l.head = n; }`
- None