

IF YOU ANSWER LIKE THIS YOU WILL GET 17/20.

QUIZ 01

الموقع الإلكتروني moodle.iugaza.edu.eg

تراكييب بيانات وخوارزميات

Question 1
Not yet answered
Marked out of 3
Flag question

This code is dedicated for a special purpose of doubly-linked-list. Which functionality is done by this code

```
public boolean Play_With_Doubly() {  
    bool result = false;  
    Node node_temp = null;  
    Node node_current = head;  
    if (node_current == null) System.out.println("Cannot do anything with an empty list!");  
    while (node_current != null) {  
        node_temp = node_current.previous;  
        node_current.previous = node_current.next;  
        node_current.next = node_temp;  
        node_current = node_current.previous;  
    }  
    if (node_temp == null) {  
        head = node_temp.previous;  
        result = true;  
    }  
    return result;  
}
```

Select one:

- ☐ a. Reverse the doubly-linked-list
- ☐ b. Reverse the first half of the doubly-linked-list
- ☐ c. Check whether the doubly-linked list is symmetric or not.
- ☐ d. Reverse the Second half of the doubly-linked-list

The Answer is A

Question 2
Not yet answered
Marked out of 1
Flag question

The big O for the following function: $6*n + 12*n*\log n - 2$ is

Select one:

- ☐ a. $O(n)$ for $c_0 = 10$ and $n_0 = 1$
- ☐ b. $O(n^2)$ for $c_0 = 10$ and $n_0 = 1$
- ☐ c. $O(n^2)$ for $c_0 = 10$ and $n_0 = 1$
- ☐ d. $O(n.\log n)$ for $c_0 = 12$ and $n_0 = 2$

The answer is D

Question 3

Not yet answered

Marked out of 2

Flag question

The running time for many algorithms for n elements; e.g. compute the average, sum, product is following this formula: $O(1 + 2 + \dots + n)$. This Big O is:

Select one:

- ☐ a. $O(n^3)$
- ☐ b. $O(n)$
- ☐ c. $O(n^2)$
- ☒ d. $O(\frac{n^2+n}{2})$

The answer is D

Question 4

Not yet answered

Marked out of 3

Flag question

What is the functionality of the following code for circularly-linked list?

```
public float CLL_function()
{
    if(head == null)
        return -1.0;
    float var;
    Node node_temp = head;
    while(node_temp.getNext() != head)
        node_temp = node_temp.getNext();
    if(node_temp == head)
    {
        var = head.getElement();
        head = null;
        return var;
    }
    node_temp.setNext(head.getNext());
    var = head.getElement();
    head = head.getNext();
    return var;
}
```

Select one:

- ☐ a. It returns element data from the end of the circularly-linked list
- ☐ b. It returns the element data from the start of the circularly-linked list
- ☐ c. It returns the data and deletes the node at the end of the circularly-linked list
- ☐ d. It returns the element data and deletes the node from the start of the circularly-linked list

The answer is D

Question 5

Not yet
answered

Marked out of 3

🚩 Flag question

What is the functionality of the following piece of code?

```
public int SLL_function(int i)
{
    Node temp = head;
    int var = 0;
    while(temp != null)
    {
        if(temp.getElement() == i)
        {
            return var;
        }
        var = var+1;
        temp = temp.getNext();
    }
    return Integer.MIN_VALUE;
}
```

- ☐ a. Find and return the position of the given element in the singly-linked list
- ☐ b. Find and insert a new element in the singly-linked list
- ☐ c. Find and delete a given element in the singly-linked list
- ☐ d. Find and return a given element in the singly-linked list

The answer is A

Question 6

Not yet
answered

Marked out of 3

Flag question

In the lecture, we learned how to insert an element at beginning of doubly-linked-list. Which code will perform this task?

Select one:

- ☐ a.

```
public class insertFront(int data)
{
    Node node = new Node(data, head, head.getNext());
    node.getNext().setPrev(node);
    head.setNext(node.getNext());
    size++;
}
```
- ☐ b.

```
public class insertFront(int data)
{
    Node node = new Node(data, head, head);
    node.getNext().setPrev(node);
    head.setNext(node);
    size++;
}
```
- ☐ c.

```
public class insertFront(int data)
{
    Node node = new Node(data, head, head.getNext());
    node.getNext().setPrev(head);
    head.setNext(node);
    size++;
}
```
- ☐ d.

```
public class insertFront(int data)
{
    Node node = new Node(data, head, head.getNext());
    node.getNext().setPrev(node);
    head.setNext(node);
    size++;
}
```

The answer is D

Question 7

Not yet answered

Marked out of 1

Flag question

According to our lectures, which task is the more appropriate for circularly-linked list?

Select one:

- ☐ a. Redo or Undo operations in any editor since you can traverse the operations easily.
- ☐ b. Implementing hash-tables while having several collisions
- ☐ c. Assigning the processes in CPU to resources
- ☐ d. In program counter, while using the recursive calls

The answer is C

Question 8

Not yet answered

Marked out of 4

Flag question

Match each data structure operation and its complexity

Insert an element at the start of the array	O left parenthesis n right parenthesis
Insert an element at the end of the singly-linked-list	O left parenthesis n right parenthesis
Sort the elements of an array by bubble sort	O left parenthesis n squared right parenthesis
The height of binary search tree	O left parenthesis log space n right parenthesis

But Second one its O left parenthesis 1 right parenthesis

Question 9

Not yet answered

Marked out of 1

Flag question

Given that the size of an array is "n" and we want to insert an element at Index "k" where $0 \leq k < n$. What is the complexity of this insertion?

Select one:

- ☐ a. $O(\log n)$
- ☐ b. $O(1)$
- ☐ c. It depends on the value of k and n . It could be $O(1)$ or $O(n)$
- ☐ d. $O(n)$

The answer is D

Question 10
Not yet
answered
Marked out of 4
Flag
question

A singly-linked-list contains number of elements where each element is a number. The following code is designed to count the number of elements in the list. Which one is doing this task?

Select one:

- ☐ a.

```
public int length(Node node_head)
{
    int size = 0;
    Node cur = node_head;
    while(cur!=null)
    {
        size++;
        cur = cur.getNext();
    }
}
```
- ☐ b.

```
public int length(Node node_head)
{
    int size = 0;
    Node cur = node_head;
    while(cur!=null)
    {
        cur = cur.getNext();
        size++;
    }
    return size;
}
```
- ☐ c.

```
public int length(Node node_head)
{
    int size = 0;
    Node cur = node_head;
    while(cur!=null)
    {
        size++;
        cur = cur.getNext().getNext();
    }
    return size;
}
```
- ☐ d.

```
public int length(Node node_head)
{
    int size = 0;
    Node cur = node_head;
    while(cur!=null)
    {
        size++;
        cur = cur.getNext();
    }
    return size;
}
```

The answer is D

QUIZ 02

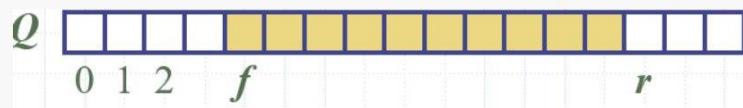
Question 1

Not yet answered

Marked out of 3

Flag question

After some operations which run over the a circular queue implemented by an array, we get the following snap-shot:



Can you identify the set of operations which lead to this situation?

Select one:

- ☐ a. 10 enqueue operations
- ☐ b. 4 Enqueue operations, then 2 dequeue operation, then 4 enqueue operations, then 3 dequeue operations, then 7 enqueue operations
- ☐ c. 13 enqueue operations, then 4 dequeue operations
- ☐ d. 2 Enqueue operations, then 1 dequeue operation, then 5 enqueue operations, then 3 dequeue operations, then 7 enqueue operations

Question 2

Not yet answered

Marked out of 2

Flag question

Which of these could happen if a recursive method does not have a base-case at the start of method?

- ☐ a. The Java runtime will control the problem
- ☐ b. After 1000,0000,000 calls it will be automatically stopped
- ☐ c. An infinite loop occurs
- ☐ d. System stops the program after some time

Question 3

Not yet answered

Marked out of 1

Flag question

A data structure i.e. queue, if implemented using an array of size "MAX_SIZE", gets full when?

Select one:

- ☐ a. $\text{Rear} = \text{front}$
- ☐ b. $\text{Front} = \text{rear} + 1$
- ☐ c. $\text{Rear} = \text{MAX\_SIZE} - 1$
- ☐ d. $\text{Front} = (\text{rear} + 1) \bmod \text{MAX\_SIZE}$

Question 2) The answer is C

Question 3) The answer is D

Question 4

Not yet answered

Marked out of 1

Flag question

The following algorithm is used to compute the summation of array elements:

Algorithm **linearSum(A, n)**:

Input: Array A, of integers, Integer n such that $0 \leq n \leq |A|$

Output: Sum of the first n integers in A

if $n = 0$ **then**

return 0

else

return linearSum(A, n - 1) + A[n - 1]

Which statements are correct?

Select one or more:

- ☐ a. This algorithm will be called 4 times to find the sum of data array = [2, 6, 5].
- ☐ b. The result of the algorithms come at the end of the call back and not directly at the base step.
- ☐ c. This algorithm uses the tail recursion.
- ☐ d. The complexity of this algorithm is $O(n)$.

The answer is a, b and d

Question 5

Not yet answered

Marked out of 3

Flag question

We use two stacks to store the numbers and the operations respectively in order to solve mathematical inequalities. The stack of numbers has 2, 3, 24, 14 in this order. The stack of operations has *, -, $\leq$ in this order. How does the original inequality look like?

Select one:

- ☐ a. $14 - 24 \leq 3 \times 2$
- ☐ b. $14 \leq 24 - 3 \times 2$
- ☐ c. $14 - 24 \leq - 3 \times 2$
- ☐ d. $2 \times 3 - 4 \leq 14$

The answer is B

Question 6

Not yet
answered

Marked out of 1

Flag question

What will be the output of this code?

```
import java.util.*;
class Collection_iterators
{
    public static void main(String args[])
    {
        LinkedList list = new LinkedList();
        list.add(new Integer(2));
        list.add(new Integer(8));
        list.add(new Integer(5));
        list.add(new Integer(1));
        Iterator i = list.iterator();
        Collections.reverse(list);
        Collections.shuffle(list);
        i.next();
        i.remove();
        while(i.hasNext())
            System.out.print(i.next() + " ");
    }
}
```

Handwritten notes:
A red 'X' is next to the 'main' method signature.
A red bracket on the right side of the list operations is labeled 'List'.
A red arrow points from the 'i.next()' call to the word 'Iterator'.
A red arrow points from the 'while' loop to the word 'Printing'.

- ☐ 2 8 5
- ☐ 2 1 8
- ☐ 2 5 8
- ☐ 8 5 1

The answer is D

Question 7

Not yet
answered

Marked out of 2

Flag question

Given that a Stack where its size is 5 which is implemented in Java. It has executed the following operations:

Push(1); Pop(); Push(2); Push(3); Pop(); Push(4); Pop(); Pop(); Push(5);

After the completion of all operation, the number of elements in the Stack?

Select one:

- ☐ a. 2
- ☐ b. 4
- ☐ c. 3
- ☐ d. 1

The answer is D

Question 8

Not yet answered

Marked out of 3

Flag question

What does this code exactly do?

```
public void Stack_Function(queue tmp_q) {
    stack tmp_s;

    while (!tmp_q.empty()) {
        tmp_s.push(tmp_q.front());
        tmp_q.dequeue();
    }

    while (!tmp_s.empty()) {
        tmp_q.enqueue(tmp_s.top());
        tmp_s.pop();
    }
}
```

Select one:

- ☐ a. This code will copy the elements in descending order
- ☐ b. This code will implement the enqueue operation of queue by using stack operations
- ☐ c. This code will reverse the order of the queue elements
- ☐ d. This code will copy the elements in ascending order

The answer is C

Question 9

Not yet answered

Marked out of 2

Flag question

The following operations are applied in a **List data structure**. What will be the list contents at the end?

addLast(8); first(); addAfter(p,5); before(q); addBefore(q,3); addFirst(9);

Select one:

- ☐ a. (9z, 8p, 5q, 3r)
- ☐ b. (9s, 3r, 5q, 8p)
- ☐ c. (9s, 8p, 3r, 5q)
- ☐ d. (3r, 9z, 8p, 5q)

Question 10

Not yet answered

Marked out of 1

Flag question

The java.util.Iterator interface has several methods available in order to perform certain operations while using it in programming. Which statement is not correct?

Select one:

- ☐ a. The iterable interface in Java is the same one as iterator; however it is the new implementation.
- ☐ b. This interface supports remove() operation.
- ☐ c. You cannot reset the iterator to go back to the start of the sequence.
- ☐ d. This interface supports hasNext() and next() operations.

9) The answer is C

10) The answer is a

Write a code **ONLY** three Java instructions: which recursively reduces **x** to 0, while accumulating the result in **y**? Each code instruction shall end with **;** and write them in one line.

```
int fun (int x, int y) {  
}
```

Answer:

```
int fun(int x, int y) { return x == 0 ? y : fun(x - 1, y + x);  
}
```

Question 12

Not yet answered

Marked out of 2

Flag question

Fibonacci sequence can be computed by using recursion since it contains modularity in its form. Which statements are correct?

Select one or more:

- ☐ a. The Fibonacci can be computed by using linear calling which is $O(n)$
- ☐ b. In order to optimize the algorithm to compute Fibonacci number, one shall return two values in each return of the algorithm; one for $\text{Fib}(k-1)$ and one for $\text{Fib}(k-2)$.
- ☐ c. The Fibonacci can be computed by using Fibonacci_Tree calling which is $O(\log n)$
- ☐ d. The best recursion algorithm to compute it will be exponential, since it depends on binary calling of the function; one $\text{Fib}(n-1)$ and one for $\text{Fib}(n-2)$.

The answer is a b d

الموقع الإلكتروني

Quiz (page 3 of 3) - Google Chrome

moodle.fugaza.edu.ps/mod/quiz/attempt.php?attempt=2465752&cmid=343633&page=2#question-2717936-3

Question 13

Not yet answered

Marked out of 1

Flag question

The following code is to evaluate the spans which is useful for many application:

```
Algorithm spans1(X, n)  
Input array X of n integers  
Output array S of spans of X  
S ← new array of n integers  
for i ← 0 to n - 1 do  
    s ← 1  
    while s ≤ i ∧ X[i - s] ≤ X[i]  
        s ← s + 1  
    S[i] ← s  
return S
```

One of the following statements is correct?

Select one:

- ☐ a. This code complexity is $O(n^2)$, however it can be implemented by queue in complexity of $O(n)$
- ☐ b. This code complexity is $O(n^2)$, however it can be implemented by stack in complexity of $O(n)$
- ☐ c. This code complexity is $O(n^2)$ even if we implement this with queue
- ☐ d. This code complexity is $O(n^2)$ even if we implement this with stack

The answer is D

QUIZ 03

Quiz#3 (page 1 of 2) - Google Chrome

moodle.iugaza.edu.ps/mod/quiz/attempt.php?attempt=2501216&cmid=343635

الرئيسية

تراكيب بيانات وخوارزميات

Question 1
Not yet answered
Marked out of 2
Flag question

Why is it very important to have a total order relation over the keys in priority queue?

Select one or more:

- ☐ a. Because priority queue ADT implements Comparator ADT
- ☐ b. Because total order relation means the keys must be integers
- ☐ c. Because total order relations means one can perform comparison over the keys.
- ☐ d. Because this makes the implementation easier and understandable

Question 2
Not yet answered
Marked out of 2
Flag question

If we perform the following operations, for priority queue; what will be the last status of it?
insert(7, Y), insert(12, X), insert(23, B), min(), insert(5, M),
removeMin(), removeMin()

Select one:

- ☐ a. {(12, X), (23, B)}
- ☐ b. {(5, M), (23, B)}
- ☐ c. {(7, Y), (12, X)}
- ☐ d. {(23, B), (12, X)}

Question 3
Not yet answered
Marked out of 2
Flag question

In many applications, **Tree** data structure is used to represent the internal representation of the data stored. However, in order to traverse the data, one

1) C, D

2) A

Quiz#3 (page 1 of 2) - Google Chrome

moodle.iugaza.edu.ps/mod/quiz/attempt.php?attempt=2501216&cmid=343635

الرئيسية

Question 3
Not yet answered
Marked out of 2
Flag question

In many applications, **Tree** data structure is used to represent the internal representation of the data stored. However, in order to traverse the data, one has to think carefully about the used mechanism. Which of the following statements is correct?

Select one:

- ☐ a. Breadth first traversal can be used in a tree, however without recursion
- ☐ b. Depth first traversal can be used in a tree, however with recursion
- ☐ c. Euler traversal represents all kind of traversal possible over the tree structure
- ☐ d. None of the statements is correct

Question 4
Not yet answered
Marked out of 1
Flag question

Given a binary search tree which is complete tree. If its depth is 5, then the number of nodes will be _____

Select one:

- ☐ a. 32
- ☐ b. 64
- ☐ c. 31
- ☐ d. 63

Question 5
Not yet answered

In the following picture, you see in the first line a sequence which is sorted in decreasing order. A clever student has applied certain algorithms without

3) B

4) C

Question 5
Not yet answered
Marked out of 3
Flag question

In the following picture, you see in the first line a sequence which is sorted in decreasing order. A clever student has applied certain algorithms without using external data structure to sort the elements of this sequence which is seen from Line 2 to Line 7.

Select one or more:

- ☐ a. This is done by using in-place mechanism
- ☐ b. This sort use swaps instead of modifying the sequence.
- ☐ c. This sort could be a selection-sort
- ☐ d. This sort could be an insertion-sort

5) A, B

Question 6
Not yet answered
Marked out of 2
Flag question

What is the postorder traversal of the following tree?

```

graph TD
    7((7)) --> 1((1))
    7 --> 3((3))
    1 --> 6((6))
    6 --> 5((5))
    3 --> 2((2))
    2 --> 4((4))
  
```

Select one:

- ☐ a. 7165423
- ☐ b. 5614237
- ☐ c. 5624137
- ☐ d. 6517423

6) B

Question 7

Not yet answered

Marked out of 1

Flag question

In order to implement the tree by using singly-linked list, one has to adjust the node definition which is used commonly in singly linked list. The modification is: _____

Select one:

- ☐ a. The node shall have four references; one for the parent, one for the child, one for sibling and one for the element.
- ☐ b. The node shall have two references, one for the child and one for the element.
- ☐ c. The node shall have two references, one for the parent and one for the element.
- ☐ d. The node shall have three references; one for the parent, one for the child and one for the element.

7) A

Question 8

Not yet answered

Marked out of 4

Flag question

In order to compute the height of a tree, a lecturer has obtained many results from the students. Which of the following choices are correct?

Select one or more:

- ☐ a. The best solution to compute in by $O(n)$ as in the following algorithm:


```

Public int Height()
{
    int h = 0;
    for(Position <E> p: positions())
        if(!isExternal(p))
            h = Math.max(h, depth(p))
    return h;
}
```
- ☐ b. The following algorithm is inefficient, since it uses *height* function


```

Public int Height(Position <E> p)
{
    int h = 0;
    for(Position <E> c: children(p))
        h = Math.max(h, 1 + Height(c))
    return h;
}
```
- ☐ c. The best solution to compute in by $O(n)$ as in the following algorithm:


```

Public int Height(Position <E> p)
{
    int h = 0;
    for(Position <E> c: children(p))
        h = Math.max(h, 1 + Height(c))
    return h;
}
```
- ☐ d. The following algorithm is inefficient, since it uses *depth* function


```

Public int Height()
{
    int h = 0;
    for(Position <E> p: positions())
        if(!isExternal(p))
            h = Math.max(h, depth(p))
    return h;
}
```

8) C, D

Question 9

Not yet answered

Marked out of 2

Flag question

If you know that the height of a tree is the longest path from root to a leaf. What is the minimum number of nodes in binary tree which its height 7 is _____

Select one:

- ☐ a. 6
- ☐ b. 7
- ☐ c. 5
- ☐ d. 8

Question 10

Not yet answered

Marked out of 1

Flag question

Given a binary tree which its preorder is: 25, 10, 12, 13, 20, 18, 16, 19. The postorder traversal will be _____

Select one:

- ☐ a. 13, 12, 10, 16, 19, 18, 20, 25
- ☐ b. 20, 19, 18, 16, 25, 12, 13, 10
- ☐ c. 19, 16, 18, 20, 13, 12, 10, 25
- ☐ d. 10, 13, 12, 25, 16, 18, 19, 20, 13

9) D

10) C

QUIZ 04

Quiz#4 (page 1 of 2) - Google Chrome

moodle.iugaza.edu.ps/mod/quiz/attempt.php?attempt=2501307&cmid=354201

الرقمية

تراكيب بيانات وخوارزميات

Question 1
Not yet answered
Marked out of 4
Flag question

In the following given hash table, we use linear probing to resolve any conflict in case of the same hash value occurring?

0	1	2	3	4	5	6	7	8	9
								38	69

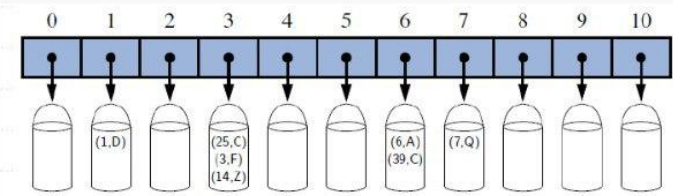
In case of inserting 28, what will be the right location?

Select one:

- ☐ a. 0
- ☐ b. 6
- ☐ c. 7
- ☐ d. 1

Question 2
Not yet answered
Marked out of 3
Flag question

During our course, we have seen the following picture:



Select one:

- ☐ a. This presents a Map which has 11 elements
- ☐ b. Hash function is used to allow more usage of this map by using double hashing
- ☐ c. In the worst case, searching for key complexity is $O(1)$
- ☐ d. A possible hash function for this Map is: $h(k) = k \bmod 11$

Question 3
Not yet answered
Marked out of 2

Given a binary tree which contains words in alphabetical order. If we insert the following words BXXXXX, PEAXXX, AXXXXX, PEAXXX, CXXXXX, MXXXXX, PAXXXX. What will be height of the tree?

Select one:

1) A

2) D

Quiz#4 (page 1 of 2) - Google Chrome

moodle.iugaza.edu.ps/mod/quiz/attempt.php?attempt=2501307&cmid=354201

الرئيسية

☐ d. A possible hash function for this Map is: $h(k) = k \bmod 11$

Question 3
Not yet answered
Marked out of 2
Flag question

Given a binary tree which contains words in alphabetical order. If we insert the following words BXXXXX, PEAXXX, AXXXXX, PEAXXX, CXXXXX, MXXXXX, PAXXXX. What will be height of the tree?

Select one:

☐ a. 3
☐ b. 3
☐ c. 5
☐ d. 4

Question 4
Not yet answered
Marked out of 1
Flag question

In order to handle the collision while using hash tables, we use linear probing. What does it mean a linear probing?

Select one:

☐ a. All previous options are correct
☐ b. Hash function includes no polynomial functions but rather linear functions
☐ c. We go by linear manner to try the next free slot
☐ d. Another linear hash function will be used in case of collision

Question 5
Not yet answered
Marked out of 2
Flag question

After performing some operations in **Map data structure**, we get the following view:

$(7,B), (2,E), (8,D)$

Which is the right order of operations to get the previous structure?

Select one:

☐ a. put(7, B), put(7, F), put(2, E), put(8, D), size(), get(7)
☐ b. put(7, B), put(2, E), put(8, D), size(), get(7), put(8, F)
☐ c. put(5, A), put(7, B), put(2, C), put(8, D), size(), put(2, E), remove(5)
☐ d. put(7, B), put(2, C), put(8, D), size(), get(7)

المصفحة التالية

Quiz navigation

3) D

4) C

5) C

Question 6

Not yet answered

Marked out of 1

Flag question

A hash function is composed into two parts; one for hash code and one for compression function. Considering the hash code mechanisms, what is the correct statement?

Select one:

- ☐ a. All options are valid hash coding
- ☐ b. One can use the memory address to encode it as integer
- ☐ c. One can partition the bits of key into several components, where each one can be computed separately and then summed at the end.
- ☐ d. One can interpret the bit-representation of the key to be an integer

Question 7

Not yet answered

Marked out of 3

Flag question

The following fragment of Java code is performing _____ operation over splay tree:

```
Public node Splay_Speical_Operation(node tmp)
{
    node y = tmp.left;
    tmp.left = y.right;
    y.right = tmp;
    return y;
}
```

Select one:

- ☐ a. zig-zig operation
- ☐ b. left rotation of subtree
- ☐ c. zig-zag operation
- ☐ d. right rotation of subtree

6) A

7) D

Question 8

Not yet answered

Marked out of 2

Flag question

In the standard implementation of map data structure ADT by using unsorted table map, the following method of *put* is used.

```
26 public V put(K key, V value) {
27     int j = findIndex(key);
28     if (j == -1) {
29         table.add(new MapEntry<>(key, value));
30         return null;
31     } else
32         return table.get(j).setValue(value);
33 }
```

Why Line 32 is very important in this method?

Select one:

- ☐ a. All the previous answers are correct
- ☐ b. To assure that the return value is not null always
- ☐ c. To overwrite the existing Value if the key already exists
- ☐ d. To find the proper place of inserting the value

8) C

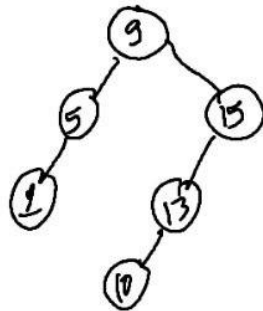
Question 9

Not yet answered

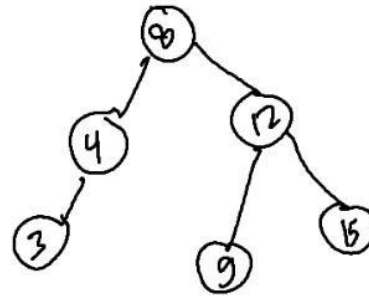
Marked out of 2

Flag question

Given that the following figures, which statements are correct?



fig(1)



fig(2)

Select one or more:

- ☐ a. Figure 1 and Figure 2 represent AVL Tree property
- ☐ b. Figure 1 represents AVL Tree property
- ☐ c. Figure 2 represents AVL Tree property
- ☐ d. Figure 1 is not a binary search tree

9) C