



NATIONAL OPEN UNIVERSITY OF NIGERIA

University Village, Plot 91, Cadastral Zone,
Nnamdi Azikiwe Expressway, Jabi, Abuja

FACULTY OF SCIENCES
DEPARTMENT OF COMPUTER SCIENCE
2021_1 EXAMINATIONS

Course Code: CIT 341

Course Title: Data Structures

Credit Units: 3

Time Allowed: 2½ Hours

Instruction: Answer Question one (1) and any other four (4)

1.

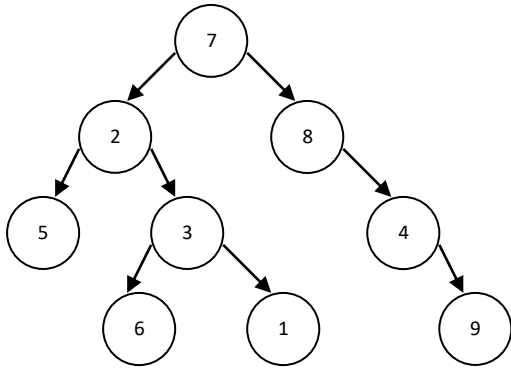
- a. Explain what you understand by Data structure **(3 marks)**
- b. Explain the essence of analyzing algorithms **(3 marks)**
- c. Differentiate between private and protected member of a class **(3 marks)**
- d. Given a empty queue Q , what will be the content of Q after each of the following operations is executed?
Q.enqueue(5); Q.enqueue(3); Q.enqueue(Q.dequeue()); Q.enqueue(Q.getHead()); enqueue(8); dequeue();
(3 marks)
- e. Given two empty stacks A and B when this sequence of operations is performed,
A.push(1) , B.push(A.top()), B.push(2) , A.push(3) , B.push(A.pop())
 - i. What is the element on top of stack A **(2 marks)**
 - ii. What is the element on top of stack B **(2 marks)**
- f. Given a tree of height h=3,
 - i. what is the maximum number of nodes that the tree can have if it is an AVL tree? **(2 marks)**
 - ii. what is the maximum number of nodes that the tree can have if it is a binary search tree? **(2 marks)**
 - iii. what is the maximum number of leave nodes that the tree can have if it is an AVL tree? **(2 marks)**

2.

- a. Discuss the running time performance of the array-based implementations of a list with regards to the following.

- i. Time taken to add an element to the list (2 marks)
- ii. Time taken to delete an element from the list (2 marks)
- iii. Time taken to access an element in list (2 marks)

b. Use the tree below to answer the following questions



- i. What is the root node of the tree? (2 marks)
- ii. Explain if the tree an AVL tree. (2 marks)
- iii. Assume we are to search for key 9 on the tree. Show the order and the nodes that will be visited if the search is performed using depth-first search (2 marks)

3.

- a. Explain what you understand by the following concepts
 - i. **O-Notation** (2 marks)
 - ii. **Ω -Notation** (2 marks)
- b. Given two functions $f(n) = n^2$ and $g(n) = n$, show that $g(n) = O(f(n))$ (4 marks)
- c. What do you understand by brute force analysis of algorithm? State its strengths and weakness (4 marks)

4.

- a. Study the following code fragment program carefully and answer all questions associated with the programs

```

1 for (i=0; i<4; i++) {
2   for (j=0; j<=i; j++){
3     System.out.print(i-j);
4     System.out.println();
5   }
6 }
  
```

- i. How many times does **System.out.print(i-j)** at line 3 execute? (2 marks)
- ii. How many times does **System.out.println()** at line 4 execute? (2 marks)
- iii. What is the output of this code fragment? (2 marks)

- b. Assume a new member function for the array implementation of the unsorted list called `SwitchFront`. `SwitchFront` swap the first elements of two lists. For example, if `list1` contains the elements `a, c, d, r, t` and `list2` contains: `*, &, 6, a, b`, then the call `SwitchFront(list1, list2);` would leave `list1` with the elements `*, c, d, r, t` and `list2` with the elements `a, &, 6, a, b` Write a java methods that implements the function `SwitchFront`. **(6 marks)**
5. Consider sorting n numbers stored in array A by first finding the least element of A and exchanging it with the element in A [1]. Then find the second least element of A , and exchange it with A [2]. Continue in this manner for the first $n - 1$ elements of A .
- Write pseudo-code for this algorithm **(5 marks)**
 - What is the worst-case efficiency class of this algorithm? **(2 marks)**
 - Use the algorithm to sort the following list by completing the passes below. If you need additional passes draw them: `1, 8, 6, 5, 3, 7` **(5 marks)**

Index \ Passes	0	1	2	3	4	5
Initial list:	1	8	6	5	3	7
Pass 1:						
Pass 2:						
Pass 3:						
Pass 4:						
Pass 5:						
Pass 6:						

- 6.
- Create a class `Person`. A person has a first name and last name. The class should have a method `getName` that returns the full name of the person. Create a class, `Student` that inherits from `Person`. A student has a matriculation number and a test score. The class `student` should have a `getScore` which returns the test score of the student. **(8 marks)**

- b. Write a code fragment that will create a student object named student1 using the class you defined in part (a) above. The program should then display the name and the test score of the student. (**4 marks**)