International University - VNUHCM
School of Computer Science and Engineering

Midterm Examination

Date: 13/11/2023; Duration: 90 minutes

Printed slides and handwritten notes are allowed, laptops/PCs/PDAs **are not allowed.**

SUBJECT: Algorithms & Data Structures (IT013IU)	
Approval Signature Full name: Trần Thanh Tùng	Lecturer: Vi CHÍ THÀNH Signature Full name: VI CHÍ THÀNH
Proctor 1 Signature Full name:	Proctor 2 Signature Full name:

STUDENT INFO
Student name: **Student ID:**

INSTRUCTIONS: the total of point is 100 (equivalent to 30% of the course)

1. *Purpose:*
   - Test your knowledge of data structures and algorithms in the following topics: Array, Searching algorithms, Queue, Stacks (CLO1)
   - Examine your skill in analysis and design algorithms (CLO2)
2. *Requirement:*
   - Write the answers and draw models CLEAN and TIDY directly in the exam paper
   - Submit your exam including this paper inside

1. **Time Complexity (12 points)**

   a) What is the complexity of binary search on an array of size **n**?

   b) For a linked list, the search time can be improved by keeping the list sorted. What is the time complexity of insertion to a sorted linked list in terms of the number of nodes **n**?

   c) Order the following functions in increasing order of asymptotic complexity: **n!, 2n , nlog(n), n, n^2**. You do not have to justify your answer

   d) Given function `f(n)` and `g(n)`, we say that `f(n)` is `O(g(n))` if there are positive constants `c` and n_0 such that

   $f(n) \le c*g(n)$ for all $n \ge n_0$

   For example: We can say that `2n + 10` is `O(n)` because

   $2n + 10 \le cn$

   $\Leftrightarrow (c - 2)n \ge 10$

   $\Leftrightarrow n \ge 10(c - 2)$

   → `Pick c = 3 and n0 = 10` we will have `2n + 10 ≤ cn`

   Dr Thanh claims that **2^n** is **O(3^n)** and **3^n** is **O(2^n)**. Is he right? If so, give witness pairs to prove these assertions; otherwise, show that they cannot exist.

2. **Sorting (18 points)**

   An Integer array **b** holds values in the following order: **5, 19, 13, 3, 9, 11, 7, 17, 2**

   a) (9 points) Assuming that we need to sort **b** using **Selection sort**, complete the following table such that every row contains the state of **b** after each iteration of the outer loop.

(Initial)	5	19	13	3	9	11	7	17	2
(Step 1)									
(Step 2)									
(Step 3)									
(Step 4)									
(Step 5)									
(Step 6)									
(Step 7)									
(Step 8)									

b) (9 points) Assuming that we need to sort **b** using **Insertion sort**, draw another table similar to the one in (a) such that every row contains the state of **b** after each iteration of the outer loop.

3. **Linked list (30 points)**

Given the **sorted (ascending order)** `SinglyLinkList` class below

a) Complete the **`insertNode`** method that inserts a Node with a given integer into the sorted list at the right location.

b) Complete the **`deleteNode`** method that deletes the occurrence of a given integer **`value`** from the list.

c) Assuming that the `SinglyLinkList` is an unsorted linked list, complete the **`insertionSort`** method that sorts the list using the insertion sort algorithm.

```
public class SinglyLinkList {
  private Node first;
  public void insertNode(int value){
  // YOUR CODE
  }
  public void deleteNode(int value) {
  // YOUR CODE
  }
  public void insertionSort() {
  // YOUR CODE
  }
}
```

```
public class Node {
  private int key;
  private Node next;
}
```

4. **Queue and Stack (20 marks)**

a. (10 marks) Show the queue after each operation of the following sequence of **PRIORITY QUEUE** operations, if executed on an initially empty queue?
enqueue(1), enqueue(3), dequeue(), enqueue(0), enqueue(11), dequeue(), dequeue(), enqueue(2), enqueue(0), dequeue(), enqueue(2), enqueue(3), dequeue(), dequeue(), enqueue(1030), dequeue(), dequeue()

b. (10 marks) Show the stack after each operation of the following sequence of stack operations, if executed upon an initially empty stack?

push(1), push(3), pop(), push(0), push(11), pop(), pop(), push(2), push(0), pop(), push(2), push(3), pop(), pop(), push(1030), pop(), pop()

5. **Complexity (20 marks)**

Propose the worst-case complexity of the following operations in linked-lists

Notion: A linked list has many elements. Each element is composed of a data value and link(s).

Data structure	Unsorted, simple linked list	Sorted simple linked list	Unsorted, double linked list	Sorted double linked list
Search				
Insert a new value				
Delete a given **ELEMENT** (with data and link(s))				
Get minimum				
Get maximum				