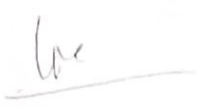



## Midterm Examination

Date: 03/04/2023; Duration: 90 minutes

Handwritten notes are allowed, laptops/PCs/PDAs are not allowed.

<b>SUBJECT: Algorithms &amp; Data Structures (IT013IU)</b>	
Approval Signature 	Lecturer: Signature 
Full name: <i>V. Thi Thu Phuong</i>	Full name: Trần Thanh Tùng
Proctor 1 Signature	Proctor 2 Signature
Full name:	Full name:
<b>STUDENT INFO</b>	
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

Note: For all calculations in this subject, the rounding up convention is used. For example, 7/2 is round up to 4

### 1. Sort (20 marks)

Given array A

Index	0	1	2	3	4	5	6	7	8	9	10	11	12
Data	72	4	12	53	15	2	8	16	40	21	7	87	35

Sort the array B with selection sort by filling the table below

Action	0	1	2	3	4	5	6	7	8	9	10	11	12

### 2. Queue and Stack (20 marks)

- (10 marks) What values are returned after each dequeue() during the following sequence of queue operations, if executed on an initially empty queue?  
enqueue(0), enqueue(3), dequeue(), enqueue(0), enqueue(4), dequeue(), dequeue(), enqueue(2), enqueue(0), dequeue(), enqueue(2), enqueue(3), dequeue(), dequeue(), enqueue(1330), dequeue(), dequeue()
- (10 marks) What values are returned after each pop() during the following series of stack operations, if executed upon an initially empty stack?  
push(0), push(3), pop(), push(0), push(4), pop(), pop(), push(2), push(0), pop(), push(2), push(3), pop(), pop(), push(1330), pop(), pop()

### 3. Linked list (30 marks)

Given 2 classes Node and DoublyLinkedList

```
public class DoublyLinkedList {
    public Node first;
    public Node last;
}
```

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

- (5 marks) Implement function delete\_first for the doubly linked list
- (5 marks) Implement function insert\_last for the doubly linked list
- (20 marks) Implement an algorithm to merge two sorted doubly linked lists to make a new sorted linked list.

#### 4. Complexity (20 marks)

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

Data structure	Unsorted, simple linked list	Sorted simple linked list	Unsorted, double linked list	Sorted double linked list
Search	$O(n)$	$O(n)$	$O(n)$	$O(n)$
Insert a new value	$O(1)$	$O(1)$	$O(1)$	$O(1)$
Delete a given value	$O(1)$	$O(1)$	$O(1)$	$O(1)$
Get minimum	$O(n)$	$O(1)$	$O(n)$	$O(1)$
Get maximum	$O(n)$	$O(1)$	$O(n)$	$O(1)$

#### 5. Queue (10 marks)

Whereas a stack allows insertion and deletion of elements at only one end, and a queue allows insertion at one end and deletion at the other end, a deque (double ended queue, pronounced like "deck") allows insertion and deletion at both ends.

Write 04 (**four**)  $O(1)$ -time procedures to insert elements into and delete elements from both ends of a deque implemented by an array.