



UNIVERSITY COLLEGE TATI (UCTATI)

FINAL EXAMINATION QUESTION BOOKLET

COURSE CODE	: BCS1223
COURSE	: DATA STRUCTURE AND ALGORITHM
SEMESTER/SESSION	: 1-2022/2023
DURATION	: 3 HOURS

**Instructions:**

1. This booklet contains 5 questions. Answer ALL questions.
2. All answers should be written in answer booklet.
3. Write legibly and draw sketches wherever required.
4. If in doubt, raise your hands and ask the invigilator.

**DO NOT OPEN THIS BOOKLET UNTIL YOU ARE TOLD TO DO SO**

**THIS BOOKLET CONTAINS 7 PRINTED PAGES INCLUDING COVER PAGE**

**QUESTION 1**

- a) Write the output for the C++ program with a recursive function in Figure 1. (5 marks)

```
#include <iostream>
using namespace std;

void fun_recursive1(int n){
    cout<<n;
    if(n!=1)
        fun_recursive1(n-1);
    cout<<n;
}

int main()
{
    fun_recursive1(5);
    return 0;
}
```

Figure 1

- b) Write a recursive function that takes as a parameter a nonnegative integer and generates the following pattern of stars (Figure 2). If the nonnegative integer is 4, then the pattern generated is: (10 marks)

```
****
***
**
*
*
**
***
****
```

Figure 2

**QUESTION 2**

- a) Explain the major advantages and disadvantages of a bubble sort algorithm. (4 marks)
  
- b) List any **TWO** (2) sorting algorithm other than bubble sort. (2 marks)
  
- c) Give a reason why maintaining a sorted list in an array for a very active process of inserting and deleting data is not preferable compared to a linked list. (4 marks)

**QUESTION 3**

You are given an array A of non-negative integers of size **20**. Your task (21 marks) is to write a C++ program to sort the array using *Bubble Sort* in an increasing order and print out the new-sorted array A.

```
const int LENGTH= 20;
int
A[LENGTH]={6, 1, 20, 2, 11, 3, 4, 5, 13, 7, 16, 10, 8, 9, 12, 15, 19
, 17, 18, 14};
```

Bubble sort algorithm is given as follows

```
for (int i = 1; i < length; i++) {
// Perform the ith pass
    for (int j = 0; j < length -i; j++) {
        if (list[j] > list[j + 1]) then
            swap list[j] with list[j + 1];
        }
    }
}
```

**QUESTION 4**

- a) Consider the list of an array in figure 3.

(8 marks)

[0]	[1]	[2]	[3]	[4]	[5]	[6]	[7]	[8]	[9]	[10]	[11]
4	8	19	25	34	39	45	48	66	75	89	95

Figure 3

Using binary search, how many comparisons are required to determine whether the following items are in the list or not? Show the values of first, last, and middle and the number of comparisons after each iteration of the loop.

- i) 89  
ii) 34
- b) You are given an array A of non-negative integers of size twenty (20). Your task is to write a C++ program that will prompt user to enter the search item and use **sequential search** algorithm to search the array list. (15 Marks)

```
const int LENGTH= 20;  
int A[LENGTH]={6,1,20,2,11,3,4,5,13,7,16,10,8,9,12,15,19,17,  
18,14};
```

## QUESTION 5

Assume that you have a linked list as shown below (Figure 4).

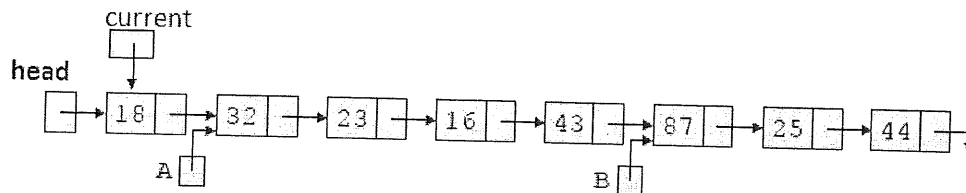


Figure 4

The definition of the node is as follows.

```

struct Node
{
    int info;
    Node *link;
};
  
```

The variable declaration is as follows:

```
Node *head, *current, *A, *B;
```

- Write the output for the following C++ statement.  
`cout<<current->link->info;` (2 marks)
- Write the output for the following C++ statement.  
`cout<<B->link->link->link;` (2 marks)
- Write a C++ statement to remove the first node from the linked list. (2 marks)
- Write a C++ statement to remove node 23 from the list. (2 marks)
- Write a C++ statement to store integer 99 to the node pointed by B. (2 marks)
- Write a C++ statements to print all values in the linked list. (5 marks)

(16 marks)

g) Write a C++ statement to remove the last node from the linked list in Figure 4. Use the following pseudocode.

1. Use a loop statement to count the number of nodes in the linked list (*nodeCount*).

2. if *nodeCount* is more than 1 then

    Traversed the linked list in (*nodeCount* -1) times (current will be at the second last node).

    Set the *current*->*link* to NULL

-----End of question-----

