VNUHCM-UNIVESRITY OF SCICENCE

# FINAL EXAMNINATION

## Semester III – Academic year 2023-2024

ARCHIVE CODE
(written by ET&QA Office)
ĐA_CK2324.3
CSC14111

| | | | |
|---|---|---|---|
| **Course name:** | **Introduction to Design and Analysis of Algorithms** | **Course code:** | **CSC14111** |
| Time: | **120 minutes** | Date: | **19/08/2024** |

Note: *Students are allowed to use paper-based materials during the examination.*

**Full name of Student:** ..Diệp Hữu Phúc.....**Student ID:** 21127125 **No:** 11...

**Question 1** (*30 points*). Given an array of $n$ integers, $\{a_1, a_2, .., a_n\}$. The task is to find the subsequence $\{a_i, a_{i+1}, .., a_{j-1}, a_j\}$ such that $\sum_{k=i}^{j} a_k$ is largest, where $1 \le i \le k \le j \le n$. If all integers in the sequence are negative, the subsequence is empty and the result is 0.

You are asked to

a. Describe **at least two** approaches for solving the task, aside from divide-and-conquer. For each approach, explain its Big-O complexity. Note that only a brief explanation for each approach is sufficient, neither an algorithm nor a full complexity analysis is required.
b. State the divide-and-conquer recurrence for this task and then use the *Master Theorem* to obtain an asymptotic Θ-bound for the recurrence.
c. Without the Master Theorem, find the closed formula for the recurrence. For simplicity, assume that $n$ is a power of 2.

**Question 2** (*25 points*). Given an unsorted array $arr[\,]$ of size $n$,where $1 \le n \le 10^5$, in which all the elements belong to the set of $\{1,2,\ldots,n\}$ (the elements are not necessary distinct). The task is to find the smallest positive number that missing from this array.

Here are some examples.

- If $arr[4] = \{2,2,1,3\}$ then answer is 4.
- if $arr[4] = \{3,3,3,4\}$ then answer is 1.
- If $arr[4] = \{1,2,4,3\}$ then answer is 5.

You are asked to describe **three** algorithms to solve this problem (not required to implement) and for each of them, briefly analyze the trade-off between time-space complexity.

**Question 3** (*20 points*). Consider the following code block.

```
1  int process(int n){
2      int i = n, m = 1, res = 0;
3      while (i > 0){
4          int j = 1;
5          while (j < m){
6              res = i * j;
7              j = j + ?;
8          }
9          m = m + 1;
10         i = i - @;
11     }
12     return res;
13 }
```

You are asked to replace the character ? and @ at the 7th line and 10th line by any two of four ways: $(1,1), (1,j), (j,1), (j,j)$, then for each of these ways, find the number of the assignment operations for estimating the complexity of corresponding code blocks.

ARCHIVE CODE
(written by ET&QA Office)
ĐA_CK2324.3
CSC14111

**Question 4** (*25 points*). A string s is called **palindrome** if

$$s[i] = s[n-1-i], \qquad \forall i = 0,1,\dots,n-1.$$

Note that the string of one letter is also palindrome.

Given a string $s$ consists of $n$ lowercase alphabet letters, where $2 \le n \le 10^5$. We need to divide it into one or more substrings such that every letter appear in some *continuous* substring and the number of **non-palindrome** substrings is maximum.

Here are some examples.

- If $s = abcdee$ then we can divide it into $ab, cde, e$ where 2 of them non-palindrome, and the answer is 2 (there is no way to divide it into 3 non-palindrome substrings).
- If $s = abc$ then the answer is 1.
- If $s = aa$ then the answer is 0.

You are asked to solve this problem using **greedy method** or **dynamic programming** (not required to implement) and give a short analysis of the time-space complexity.

Note: if you solve by exactly one way then you will get at most 15pt, if you can solve by two ways and compare the effectiveness of them then you will able to get the maximum score.

--- THE END ---