

Homework 3

CSE321 - Introduction to Algorithm Design

Deadline: 19/12/2021 - 23:59

1. Derive recurrence relations of the following algorithms and solve them to decide the complexity of the algorithms. Which algorithm would you prefer for the same problem, elaborate your answer.

(a) `Algorithm alg1(L[0..n-1])`
 `if (n==1) return L[0]`
 `else`
 `tmp = alg1(L[0..n-2])`
 `if (tmp<=L[n-1]) return tmp`
 `else return L[n-1]`

(b) `Algorithm alg2(X[l..r])`
 `if (l==r) return X[l]`
 `else`
 `flr = floor((l+r)/2)`
 `tmp1 = alg2(X[l..flr])`
 `tmp2 = alg2(X[flr+1..r])`
 `if (tmp1<=tmp2) return tmp1`
 `else return tmp2`

2. You are given a polynomial $p(x)$ like

$$p(x) = a_n x^n + a_{n-1} x^{n-1} + \dots + a_1 x + a_0,$$

and you are supposed to write a brute-force algorithm for computing the value of the polynomial at a given point x_0 . Analyze the complexity of your brute-force algorithm, and discuss whether it is possible to design an algorithm that has better complexity. Don't forget to implement your algorithm in Python.

3. You are asked to design a brute-force algorithm that counts the number of substrings that start with a specific letter and end with another letter in a given text. For example, let the first letter be 'X' and the last letter be 'Z', there are four such substrings in TXZXXJZWX. Write your brute-force algorithm, analyze its complexity, and implement in Python.

4. A metric space consists of a set and a distance function. We are given a metric space that is made up of a set of n points in k -dimensional space and Euclidean distance function. Design a brute-force algorithm that gives the distance between the closest pair of points, and find the complexity of the algorithm.
5. Profit rates of many companies have changed due to the epidemic. XYZ is a retail company with many branches on the Marmaray line. The management of the company is trying to identify the regions where they make the most profit. For this purpose, they gather the profit-loss rates of their retail shops to the table. Entries on the table are sorted in Marmaray station order.
 - (a) Write a brute-force algorithm that can find **the most profitable cluster**, provided that the cluster must contain a consecutive region. Explain the time complexity of the algorithm. For example, the most profitable cluster is (C-D-E-F) on table below.
 - (b) Write a divide and conquer algorithm that **finds the maximum profit** that belongs to the most profitable cluster (the cluster must contain a consecutive region), analyze its complexity, and write the Python code of the algorithm. For example, the maximum profit is 14 (C-D-E-F) on table below.

A	B	C	D	E	F	G
3	-5	2	11	-8	9	-5

Notes

1. Late submissions will not be accepted.
2. No collaboration is allowed, the homework must be done individually.
3. The homework will be submitted in ZIP format. The file hierarchy will be like:


```
CSE321_HW3_[StudentID].zip
| → CSE321_HW3_ANS_[StudentID].pdf
| → polynomial_[StudentID].py
| → count_str_[StudentID].py
| → cluster_[StudentID].py
```