ShanghaiTech University

CS240 Algorithm Design and Analysis

Fall 2024

Problem Set 1

Due: 23:59, Oct. 13, 2024

1. Submit your solutions to the course Gradescope.
2. If you want to submit a handwritten version, scan it clearly.
3. Late homeworks submitted within 24 hours of the due date will be marked down 25%. Homeworks submitted more than 24 hours after the due date will not be accepted unless there is a valid reason, such as a medical or family emergency.
4. You are required to follow ShanghaiTech's academic honesty policies. You are allowed to discuss problems with other students, but you must write up your solutions by yourselves. You are not allowed to copy materials from other students or from online or published resources. Violating academic honesty can result in serious penalties.

Problem 1:

Sort the following functions in ascending order of growth.

$$f_1(n) = 2024^{2025^n} \tag{1}$$

$$f_2(n) = \log_2 4^n \tag{2}$$

$$f_3(n) = 2^{\frac{1}{2}\log_2 n} \tag{3}$$

$$f_4(n) = n^{\log_2 n} \tag{4}$$

$$f_5(n) = 2025^{2024^n} \tag{5}$$

$$f_6(n) = 2024^{2025} \tag{6}$$

$$f_7(n) = n^{2024} \tag{7}$$

$$f_8(n) = n^{\sqrt{n}} \tag{8}$$

$$f_9(n) = n \log_2 (\log_2 n) \tag{9}$$

Problem 2:

Analyze the time complexities of the following algorithms and explain your reasoning.

Algorithm 1

```
for i ← 1 to n by i ← i + 1 do
    for j ← i^2 downto 0 by j ← j − 1 do
        for k ← 1 to j by k ← k + 1 do
            res_1 ← res_1 + ij + jk
        end for
        for k ← 1 to j by k ← 2k do
            res_2 ← res_2 + ik + ij
        end for
    end for
end for
```

Problem 3:

ShanghaiTech University is planning to establish new research centers throughout its campus with the goal of maximizing the total research output. The campus is modeled as an undirected graph $G = (V, E)$, where each vertex u corresponds to a potential location for a research center. Every vertex u has a nonnegative integer value p_u, representing the potential research output for that location. Due to constraints on resources, no two research centers can be placed at neighboring vertices to prevent competition for resources. Your task is to develop an algorithm that selects a subset $U \subseteq V$ of locations that maximizes the total research output $\sum_{u \in U} p_u$. Assume that the campus network G is a tree (i.e., it is acyclic).

(a) Consider the following "greedy" algorithm for placing research centers: First, choose the vertex u_0 with the highest research output from the tree and add it to the subset U. Then, remove u_0 and all its neighboring vertices from the graph, making them ineligible for further selection. Continue this process until no vertices remain. Provide a counterexample to demonstrate that this algorithm does not always produce a placement that maximizes total research output.

(b) Now, suppose that ShanghaiTech University does not have specific research output values for different locations, and instead assumes that all potential research center sites are of equal importance. The new objective is to design an algorithm that selects the largest possible number of research center sites. Describe a simple greedy algorithm for this situation and provide a proof of its correctness.

Problem 4:

You are driving on a highway represented by an array of fuel stations. You start at the first station. Each station in the array has a number representing how far you can drive from that station before needing to stop at another one for fuel. Assume you always have enough fuel to reach the last station. Your goal is to reach the last station using the smallest possible amount of refuel. Please provide a greedy solution for this scenario.

Example:

- **Input:** `stations = [2, 3, 0, 1, 4]`
- **Output:** `2`
- **Explanation:** The minimum number of refuels to reach the last station is 2. Refuel at station 0 to drive to station 1, then refuel at station 1 to drive directly to the last station.

Problem 5:

The High-Speed Railway Network Company has decided to improve its railway connections across a vast country with n cities spread from West to East. As part of the infrastructure planning team, your job is to design an efficient network of train routes. However, the company has placed specific constraints: all trains must only travel eastward, and every passenger must be able to travel from any city to any other city to the east with at most one stop. Currently, setting up direct routes for every possible journey would require a prohibitively large number of routes, approximately $\Omega(n^2)$. Your task is to develop a more efficient plan, designing no more than $O(nlogn)$ routes while ensuring that any passenger traveling between any two cities to the east can do so with at most a single stop. Prove that your set of routes satisfies these requirements.

Divide Step. Describe the divide step of your algorithm here, making sure to mention what subproblems are produced. Include the total number of routes created during the divide step.

Combine Step. Describe the combine step of your algorithm here. Include the total number of routes created during this combine step.

Algorithm. Describe your whole algorithm here. You do not need to repeat the procedures for divide and combine, you're welcome to simply reference them here.

Correctness. Prove the correctness of your algorithm. Namely, shows that the selected routes have the property that one can travel from any city to an eastward city with at most a single connection.

Number of Routes. Express the number of routes your algorithm selects as a recurrence relation. Describe, using the recursion tree method or induction proof, how you know that this recurrence is $O(nlogn)$.

Problem 6:

In a city rich with history, there is an old town that was once the heart of the city. However, as time passed, the streets and buildings of the old town fell into disrepair, and the layout became chaotic. The streets in this old town are connected like a linked list, one after another, and each street has a unique number representing its construction period and architectural style. Over the years, these numbers have fallen out of order, leading to inconvenient traffic, dilapidated buildings, and a decline in the quality of life for the residents. To revitalize the old town, the city government has launched the "Old Town Street Reconstruction Plan". The planners aim to restore the streets to their former glory by arranging them in an orderly sequence according to their numbers, making the layout more logical and facilitating better traffic flow and daily life. Your task is to act as the engineer for this reconstruction project, responsible for reordering these streets.

Given a linked list representing the structure of the streets, where the head of the list is *head*, you are required to use an efficient divide-and-conquer algorithm to sort all the street numbers in ascending order. The reconstructed streets must be sequentially connected, restoring the old town's orderly beauty. Please return the sorted linked list of streets.

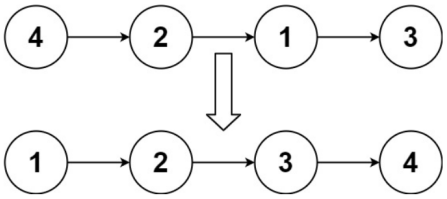

Figure 1: For example

For example, the street numbers of the old town are currently arranged as $[4, 2, 1, 3]$, with the head node $head = 4$. You need to sort them in ascending order as $[1, 2, 3, 4]$.