CSE 321 – Fall 2023

Homework 3

Due Date: (10/12/2023) 23:55

1. Let us consider a scenario where you are a member of a loyalty program offered by a shopping mal. The mall center has launched a campaign offering significant discounts to its customers. Customers can benefit from this campaign by receiving various levels of discounts based on the list of shops they have made purchases from. The discounts vary depending on the sequence of stores visited, providing customers with different potential savings opportunities according to their budget.
   To determine the highest achievable discount, please provide pseudo code for a recursive exhaustive search algorithm. This algorithm will calculate the individual discount for every possible combination of stores visited and will return the set of stores resulting in the highest discount. For the purpose of this algorithm, we can assume the existence of a *calc_discount(set_of_stores)* function. This function calculates the discount for a given set of stores in constant time complexity $O(1)$. Write the recurrence relation for your algorithm and calculate the average-case $\Theta()$ complexity of the derived recurrence relation.

2. In a distributed computer system, there are n users, n processes, and n processors. Each processor has a unique cost associated with executing each process for every user. Develop an exhaustive search algorithm aimed at identifying the optimal job scheduling strategy. This strategy should aim to minimize the overall cost incurred in executing all processes within the system. The exhaustive search algorithm should be designed to explore all possible combinations of job assignments, considering user-process pairs allocated to processors. Provide the pseudo code and a comprehensive analysis of the algorithm encompassing the worst-case, best-case, and average-case time complexities of your proposed algorithm.

3. Within a manufacturing facility, there exists a robotic arm responsible for assembling several distinct parts to complete a product. Upon assembling a part, the arm remains in a specific position, and there is an associated energy cost to assemble the next part from the current arm position. Provide pseudo code of an exhaustive search algorithm aimed at determining the most energy-efficient sequence for assembling the product. This algorithm should thoroughly explore all possible assembly sequences and identify the sequence that minimizes

the total energy consumption. Provide a comprehensive analysis encompassing the worst-case, best-case, and average-case time complexities of your proposed algorithm.

4. In the coin change problem, a set of coins with different denominations is provided along with a target amount of change. The objective is to determine the minimum number of coins needed to achieve the exact amount of change. Each coin denomination represents a value usable to make the change. Imagine a scenario where you are given a set of coins with varying denominations and a specific target amount. Please propose an exhaustive search algorithm to solve this problem by determining the minimum number of coins required to reach the exact given amount of change. Provide the pseudocode for the algorithm and explain each step in detail. Additionally, analyze the time complexity (Big-Oh notation) of your algorithm.

5. Write a recurrence relation for the following function when called with an array of length n. Provide the average-case complexity by solving the recurrence relation.

```
def find_min_max(arr, low, high):
    if low == high:
        return arr[low], arr[low]

    if high - low == 1:
        if arr[low] < arr[high]:
            return arr[low], arr[high]
        else:
            return arr[high], arr[low]

    mid = (low + high) // 2

    left_min, left_max = find_min_max(arr, low, mid)
    right_min, right_max = find_min_max(arr, mid + 1, high)

    min_val = min(left_min, right_min)
    max_val = max(left_max, right_max)

    return min_val, max_val
```

Notes:

- Your answer must be handwritten and submitted via the Course MS Teams page.
- Pseudocodes should be submitted as actual Python code and submitted as separate files together with your handwritten solutions.

- If you have any questions, you can send an email to b.koca@gtu.edu.tr
- Please complete your homework individually; group studies will be regarded as cheating.