

CSE 321 - Introduction to Algorithm Design

Homework 05

Deadline: 23:55 December 26th, 2016

PS: Upload your homework to Moodle website. Do **not** bring papers to the room of the TA.

1. Answer questions below.

a) Why do we use greedy algorithms?

b) Do greedy algorithms always fail to find globally optimal solution? Why or why not?

Provide concrete examples.

2. Prove or disprove the following statements.

a) If e is a minimum-weight edge in a connected weighted graph, it must be among edges of at least one minimum spanning tree of the graph.

b) If e is a minimum-weight edge in a connected weighted graph, it must be among edges of each minimum spanning tree of the graph.

c) If edge weights of a connected weighted graph are all distinct, the graph must have exactly one minimum spanning tree.

d) If edge weights of a connected weighted graph are not all distinct, the graph must have more than one minimum spanning tree.

3. <http://www.geeksforgeeks.org/greedy-algorithms-set-1-activity-selection-problem/>
Design a greedy algorithm to solve 0/1 Knapsack Problem¹. Describe your greedy approach in detail. Write a program to show your algorithm including test on a sample set. Write code in Python. Analyze its best case, worst case, and average case complexities.

4. <http://stackoverflow.com/questions/30552656/python-traveling-salesman-greedy-algorithm>
Design a greedy algorithm to solve Travelling Salesman Problem (TSP)² with greedy approach. Describe your greedy approach in detail. Write a program to show your algorithm including test on a sample set. Write code in Python. Analyze its best case, worst case, and average case complexities.

5. <http://www.geeksforgeeks.org/graph-coloring-set-2-greedy-algorithm/>
Design a greedy algorithm to solve Map Coloring Problem³. Describe your greedy approach in detail. Write a program to show your algorithm including test on a sample set. Write code in Python. Analyze its best case, worst case, and average case complexities.

6. Consider the problem of scheduling n jobs of known durations t_1, \dots, t_n for execution by a single processor. The jobs can be executed in any order, one job at a time. You want to find a schedule that minimizes the total time spent by all the jobs in the system. (The time spent by one job in the system is the sum of the time spent by this job in waiting plus the time spent on its execution.)

Design a greedy algorithm for this problem. Does the greedy algorithm always yield an optimal solution? Prove or disprove.

7. Design a greedy algorithm for the assignment problem (see Section 3.4). Does your greedy algorithm always yield an optimal solution? Prove or disprove.

1 https://en.wikipedia.org/wiki/Knapsack_problem

2 https://en.wikipedia.org/wiki/Travelling_salesman_problem

3 https://en.wikipedia.org/wiki/Map_coloring

8. Write a pseudocode of the greedy algorithm for the change-making problem, with an amount n and coin denominations $d_1 > d_2 > \dots > d_m$ as its input. What is the time efficiency class of your algorithm?

9. Answer questions below.

a) How can we use Prim's algorithm to find a spanning tree of a connected graph with no weights on its edges?

b) Is it a good algorithm for this problem? Discuss.

10. Watch Travelling Salesman (2012), directed by Timothy Lanzone, movie and write an essay. (It should be minimum half page A-4 paper size or maximum one page A-4 paper size. Also you are expected to use 1,5 text-space and Times New Roman font family in your essay.)