

College of Computing and Information Technology



Lecturer: Dr. Nahla Belal
Course: Computing Algorithms (CS312)
TA: Eng. Mohammad Badawy



Sheet 4

1. Design an efficient algorithm to solve the following scheduling problem. Provide a pseudo-code and a worst case complexity analysis for your algorithm.

You are given a set of n jobs with a processing time t_i and a weight w_i for each job. You want to order the jobs so as to minimize the weighted sum of the completion times,

$$\sum_{i=1}^n w_i C_i .$$

Example: Suppose there are two jobs, the first takes time $t_1 = 1$ and has weight $w_1 = 10$, while the second takes time $t_2 = 3$ and has weight $w_2 = 2$. Then doing job 1 first would yield a weighted completion time of $10 \times 1 + 2 \times 4 = 18$, while doing job 2 first yields the larger weighted completion time of $10 \times 4 + 2 \times 3 = 46$.

2. Write the pseudocode of the greedy algorithm for the change-making problem, with an amount n and coin denominations d_1, d_2, \dots, d_n as its input. What is the time efficiency class of your algorithm?
3. Job Scheduling: Consider the problem of scheduling n jobs of known durations t_1, t_2, \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?
4. Write a greedy algorithm to solve the fractional knapsack problem.
5. Huffman Codes:
 - (a) Construct a Huffman code for the following data:

character	A	B	C	D	E
probability	0.4	0.1	0.2	0.15	0.15

- (b) Encode the text ABACABAD using the code of question a.
- (c) Decode the text whose encoding is 100010111001010 in the code of question a.

6. Consider a file with the following character frequencies:

A:5 B:7 C:10 D:3 E:5.

- (a) Find the Huffman tree needed to construct the Huffman code for this set.
- (b) What is the Huffman code for each of the characters in the example above?
- (c) Design an efficient algorithm that constructs the Huffman tree in $O(N)$, given a sorted set of frequencies for the N characters.