CSE 531: Algorithm Analysis and Design	Fall 2024
Homework 2	
Instructor: Kelin Luo	**Deadline: Oct/07/2024**

Your Name: ______________________ Your Student ID: ________________

Problems	1	2	3	4	5	Total
Max. Score	5	10	15	10	15	50
Your Score						

Requirement:

- Save and submit your HW 2 submission to Brightspace as a single typed PDF file. Name your file: Your Last Name Your First Name YourStudents ID Number Assignment Number. Example: **Doe_John_55552222_HW2**
- You should view your submission after you upload it to make sure that it is not corrupted or malformed. Submissions that are rotated, upside down, or that do not load will not receive credit. Illegible submissions may also lose credit depending on what can be read. You are responsible for making sure your submission went through successfully.
- To create a figure, you have the option to use software such as "Ipe Drawing Editor" or other similar tools for digital illustrations. Alternatively, you may choose to draw the figure by hand and then insert it into your document.
- The HW 2 deadline is 07 Oct, 11:59PM EST. Late submissions (within 24 hours with 25 % penalty or 48 hours with a 50% penalty) should be submitted via Email to Instructor and Head TAs. Submissions will close 48 hours after the deadline.
- Only the most recent submission is kept on Brightspace.
- The solutions will be released when the Homework 2 grades are published.
- Note that your total points for HW2 will not exceed 50. Suppose you receive $P1$, $P2$, $P3$, $P4$, and $P5$ points for each of the questions. Your HW2 grade, as counted towards your final grade, will be calculated as $5*\min\{P1+P2+P3+P4+P5, 50\}/50$.

Problem 1 (5 points). Consider an offline caching instance described as follows: we have a cache that can store $k = 4$ pages and the sequence of page requests are $\{1, 3, 2, 6, 4, 1, 6, 2, 5, 1, 4, 3, 5, 2, 6\}$. Please follow the Furthest in Future Algorithm Pseudocode using the Priority Queue in Lecture Notes and present the status of the priority queue after time $t = 6$ and $t = 12$. Refer to the provided example priority queue status after $t = 4$ for guidance.

Pages	**Priority Values**
1	6
3	12
2	8
6	7

Table 1: Priority Queue Status after time 4

Problem 2 (10 points).

(a) **(5 points)** Construct the Huffman code (i.e, please draw the encoding tree and write down the optimum prefix code) for the alphabet {a, b, c, d, e, f, g, h} with the following frequencies:

a	b	c	d	e	f	g	h
10	2	11	3	7	2	5	6

(b) **(5 points)** Give the weighted length of the code (i.e, please write down the computational formula and calculate the sum over all symbols the frequency of the symbol times its encoding length).

Problem 3 (15 points). We are given a set of boxes B, where each box i is represented by its dimensions (a_i, b_i), denoting its width and height, respectively. Given are a set of items T, where each item j is represented by its dimensions (a'_j, b'_j), denoting its width and height, respectively. Our objective is to maximize the number of items placed in the boxes. Note that for any two boxes i and j with $i < j$, we know $a_i \leq a_j$ and $b_i \leq b_j$. We can put at most 1 item in a box. Item j can be put into box i if one of the following is true:

- $a'_j \leq a_i$ and $b'_j \leq b_i$
- $a'_j \leq b_i$ and $b'_j \leq a_i$

(a) **(5 points)** Given the boxes and items provided below, please provide an optimal packing solution, e.g., put item 1 to box 1, ..., and state how many items can be placed in the boxes.

- 7 boxes of size $\{(2,1),(2,3),(5,3),(5,4),(8,5),(8,7),(8,10)\}$.
- 7 items of size $\{(1,6),(2,2),(3,1),(4,3),(8,6),(5,2),(6,10)\}$.

(b) **(10 points)** Please design an efficient greedy algorithm for solving this problem. For this problem, write your algorithm as pseudo code (in fewer than 15 lines), prove the correctness of the algorithm (with safety strategy and self-reduce argument), and describe the runtime in no more than 3 sentences.

Problem 4 (10 points). Given are an array A of length n and an integer h. For every integer i in $[h^2, n]$, let b_i be the h^2-th largest number in $A[1...i]$. The goal of the problem is to output b_{h^2}, b_{h^2+1}, b_{h^2+2},..., b_n.

For example, if $h = 2$; $n = 10$ and $A = (50, 80, 10, 30, 90, 20, 100, 40, 65, 70)$. Then $b_4 = 10$, $b_5 = 30$, $b_6 = 30$, $b_7 = 50$, $b_8 = 50$, $b_9 = 65$, $b_{10} = 70$.

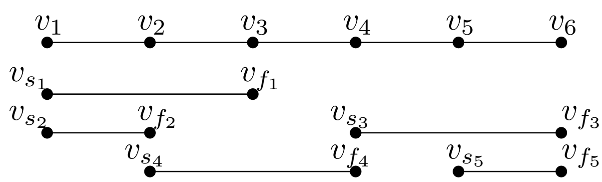
Please design an $O(n \log n)$ time algorithm for solving this problem. Please write your algorithm as pseudo code (in fewer than 15 lines), explain its correctness in no more than 3 sentences, and describe the runtime in no more than 3 sentences. (Hint: use the heap data structure.)

Problem 5: Bonus Question (15 points). Background: In this collaborative delivery problem, we address the challenge of delivering a package from a source node to a destination node within a road network, represented as a graph, using a group of drivers. Each driver is restricted to a specific area within the overall region or the road network. The handover may occur at a node within the road network when two drivers meet at that particular node. Our objective is to minimize the total number of drivers required to complete this task.

(a) **(10 points)** A formal description of the collaborative delivery problem when the road network is a **path graph**: The input is a path graph $G = (V, E)$ where $V = \{v_1, v_2, v_3, ..., v_m\}$ and $E = \bigcup_{i \in [m-1]} \{v_i, v_{i+1}\}$, n paths $I = [n]$ where each path $i \in I$ represents a driver's covered area with start-point $v_{s_i} \in V$ and end-point $v_{f_i} \in V$, i.e., the path i contains vertices $V_i = \{v_{s_i}, v_{s_i+1}, ..., v_{f_i}\}$ and edges $E_i = \bigcup_{s_i \le j < f_i} \{v_j, v_{j+1}\}$. We need to select a subset of paths $S \subseteq I$ such that the drivers could deliver the package from node v_1 to node v_m, i.e., $\bigcup_{i \in S} V_i = V$ and $\bigcup_{i \in S} E_i = E$. The objective is to minimize the total number of paths $|S|$.

Please design an efficient greedy algorithm for solving this problem. Please write your algorithm as pseudo code (in fewer than 15 lines), explain its correctness in no more than 5 sentences, and describe the runtime in no more than 3 sentences.

For example, suppose we have $m = 6$, $n = 5$ and the 5 paths are given in the following Figure. Then we can use paths 1, 4 and 3 to cover $[v_1, v_6]$ and thus only 3 paths is needed. You can not cover $[v_1, v_6]$ use less than three paths.



(b) Bonus Question **(5 points)**. The input is a **tree** graph $G = (V, E)$, and n trees $\mathcal{T} = \{T_1, T_2, ..., T_n\}$ where each tree $T_i \in \mathcal{T}$ represents a driver i's covered area. Note that for every i, $V_i \subseteq V$ and edges $E_i \subseteq E$. Still, we need to select a subset $S \subseteq \mathcal{T}$ such that the drivers could deliver the package from node v_1 to node v_m. The objective is to minimize the driver number $|S|$. Please state your safety strategy (for adding a driver to the solution in the first step) in no more than 3 sentences and explain its correctness in no more than 3 sentences.