This homework is due on **10/16/24, 12:30pm ET**.

Academic Integrity. You can work in groups of two or three people, however you must explicitly list all collaborators and materials that you used. You must write up your own solution and your own code to every problem. See Georgetown University Honor System. When in doubt, ask the instructor what is allowed.

The use of AI text generating tools (such as ChatGPT) at any point when working on the assignments will be treated as academic dishonesty.

You may only discuss problems with your group members and the course staff. The only materials you can reference when working on these problems are the textbooks, course notes, and lectures from the current semester of this course. You may not reference any other online materials or solutions from other semesters.

Coding Questions. Submit solutions to coding questions (Extra Credit Problem 4) in Cogniterra, submit solutions to all other questions in Canvas. You can typeset your solutions in LaTeX (for example, using this template: https://www.overleaf.com/read/gjdgtxxpwkkj).

Please sign up on Cogniterra using the following link:
https://cogniterra.org/invitation/c15f17682a6170840ecde408154e8655b771c981/

You can resubmit your code until it passes all tests, there is no limit on the number of attempts. You can submit your solutions in any of the following programming languages `C, C#, C++, Go, Haskell, Java, Javascript, Kotlin, Python 3, Ruby, Rust, Scala, TypeScript`, however starter files will be provided only for `Python, Java`, and `C++`.

Problem 1 (Covering Intervals, **10pt**). In this exercise, you are asked to design an efficient greedy algorithm. A complete solution will include an explanation of the main idea, pseudocode of the algorithm, proof of correctness, and analysis of the running time.

We say that an interval $[\ell, r]$ is covered by a set of intervals $[\ell_1, r_1], \ldots, [\ell_k, r_k]$ if every point of $[\ell, r]$ belongs to at least one interval $[\ell_i, r_i]$. For example, the interval $[1, 3]$ is covered by intervals $[0, 2]$ and $[1, 4]$, and is *not* covered by intervals $[0, 1]$ and $[2, 3]$.

The input consists of n intervals on the real line, given by their left and right endpoints in $L[1..n]$ and $R[1..n]$, respectively. The task is to design an efficient algorithm that finds the smallest subset of intervals that covers all the remaining intervals.

a (4 pt) Suggest a greedy way of picking one interval that should be included in a covering set. (Recall the "Activity Selection" and "Hitting Intervals" problems from class.) Note that some pairs of intervals may have same left or right endpoints. If it is crucial for your algorithm, then explain how your algorithm breaks such ties.

b (3 pt) Prove that the greedy choice you suggested is a safe move, i.e., there is an optimal solution consistent with the suggested greedy choice.

b (3 pt) Provide pseudocode of your algorithm, and analyze its running time.

Example 1:

Input:

```
L = [2,  5,  4,  1]
R = [6,  6,  7,  5]
```

The last two intervals $[4, 7]$ and $[1, 5]$ cover all the input intervals.

Output:

```
2
```

Problem 2 (Greedy Skiers, **10pt**). In this exercise, you are asked to design an efficient greedy algorithm. A complete solution will include an explanation of the main idea, pseudocode of the algorithm, proof of correctness, and analysis of the running time.

There are n skiers whose heights are given in an array $H[1..n]$, and n skis whose lengths are given in an array $L[1..n]$.

a Assign a ski to each skier so that the *total* difference between the height of a skier and her assigned skis is as small as possible. That is, find the minimum of the expression

$$\sum_{i=1}^{n} |H[i] - P[\pi(i)]|$$

over all permutations π.

a.1 (1 pt) Suggest a greedy way of pairing one skier and one ski. (Recall the "Maximum Inner Product" problem from class.)

a.2 (2 pt) Prove that this greedy choice is a safe move, i.e., there is an optimal solution consistent with your greedy choice.

a.3 (2 pt) Provide pseudocode of your algorithm, and analyze its running time.

b Assign a ski to each skier so that the *maximum* difference between the height of a skier and her assigned skis is as small as possible. That is, find the minimum of the expression

$$\max_{i=1}^{n} |H[i] - P[\pi(i)]|$$

over all permutations π.

b.1 (1 pt) Suggest a greedy idea of pairing one skier and one ski. (Recall the "Maximum Inner Product" problem from class.)

b.2 (2 pt) Prove that this greedy choice is a safe move, i.e., there is an optimal solution consistent with your greedy choice.

b.3 (2 pt) Provide pseudocode of your algorithm, and analyze its running time.

Example 1:

Input:

```
H = [8, 2, 7]
L = [8, 10, 3]
```

If the first skier gets the second ski, second skier gets the third ski, and the third skier gets the first ski, then the total difference is 4, and the maximum difference is 2.

Output:

```
total difference is 4
maximum difference is 2
```

Problem 3 (Huffman Codes, **10pt).**

a (4pt) Design an optimal Huffman code for the following frequencies forming an arithmetic progression: $f_i = 100 + i$ for $i \in \{1, \ldots, 8\}$.

b (4pt) Design an optimal Huffman code for the following frequencies forming a geometric progression: $f_i = 2^i$ for $i \in \{1, \ldots, 8\}$.

c (2pt) Consider a Huffman code of n frequencies. What is the longest possible length of a codeword? Provide *concrete* values of $f_1 \ldots, f_n$ that lead to such result, and justify your answer.

Problem 4 (Extra Credit 0+8pt). In each part of this exercise, the goal is to implement an efficient divide-and-conquer algorithm, and submit it on Cogniterra.

a (Extra credit, 1pt) Money Change.
b (Extra credit, 1pt) Car Charging.
c (Extra credit, 1pt) Fractional Knapsack.
d (Extra credit, 1pt) Largest Number.
e (Extra credit, 1pt) Maximum Inner Product.
f (Extra credit, 1pt) Hitting Intervals.
g (Extra credit, 1pt) Greedy Skiers.
h (Extra credit, 1pt) Huffman Codes.