8/28 CSP(L) notes:

CSP(L) Class Notes - 8/28

Lecture Overview

- **Reading:** Read Chapter 2 before Friday.
- **Class Schedule:**
    - No class on Monday!
    - Instructor will be gone Wednesday and Friday; recorded lectures will be provided.

Key Concepts

- **Computing Machine, Algorithm, and Computer Program:**
    - **Process:** Takes some input, processes it in a stateful (usually deterministic) manner, and produces output.
    - **Memory vs. Storage:**
        1. Memory (RAM) is volatile.
        2. Storage is non-volatile.
    - **Program Correctness:**
        1. A program is correct if it meets the problem specification, typically mapping inputs to outputs.
        2. The program should run, produce the correct outcome, and terminate in finite time.
        3. It is impossible to determine if a program will always terminate in the general case, due to the infeasibility of checking all possible input combinations.
    - **Infinite Inputs:** If a program has infinite inputs, it is impossible to guarantee termination.
- **Problem Solving in Programming:**
    - **Well-Formed Problem:** Derive a strategy or algorithm.
    - **Algorithm Validation:** Ensure the proposed algorithm always produces the correct output for all reasonable inputs in a finite amount of time.
    - **Implementation Process:**
        1. Design the problem specification.
        2. Develop an algorithm.
        3. Implement the algorithm as a program in a chosen language.
        4. Compile the program.

5. Test with a subset of inputs to ensure consistency with the algorithmic solution.

- **Programming Flow:**
  - Problem Specification (Design) → Algorithm (Implementation) → Program (Source Code) → Compilation/Interpretation → Machine Code
- **Programming in the Large:**
  - Complex and challenging task.

Case Study: Stable Marriage Problem

- **Problem Statement:**
  - Matching problem involving a certain number of men and women to create 5 couples.
  - Design a solution to match N men and N women such that no man and woman who are matched to other partners would both prefer to be with each other (avoiding a destabilizing pair).
- **Inputs & Outputs:**
  - **Input:** N men with preference-ordered lists of N women, and N women with preference-ordered lists of N men.
  - **Output:** N male/female pairings with perfect matching (no one left out).
- **Solution Algorithm:**
  - **Proposal Process:**
    - A man proposes to his first-choice partner.
    - If she is free, they become engaged.
    - If she is already engaged, she accepts the new suitor only if he outranks her current partner; otherwise, she rejects him.
    - Rejected suitors propose to the next woman on their list.
    - Repeat until all men are matched.
  - **Key Points:**
    - The algorithm is symmetrical, meaning roles can be interchanged.
    - Over the process, each man can propose to at most N women, creating N^2 possible engagements.
    - The algorithm will terminate once all men are matched, ensuring perfect matching.
    - No destabilized pair will exist, as this would contradict the preference order.
    - **Runtime:**
      - Best case: Each man matches with his first choice.
      - Worst case: Each man requires up to N^2 proposals.
- **Implementation:**
  - Ensure the implementation faithfully follows the algorithm.
  - **Choice of Programming Language:**
    - Choice is not as important in Computer Science; it's essential to be versatile.

    - Python is chosen for this course because it is interactive, type-flexible, and has useful built-in types (e.g., strings, lists, dictionaries).
    - Python runtime errors are easier to debug than compile-time errors.
    - Python is portable, easy to learn, widely used, and has plenty of libraries and extensions.

Lab on Tuesday

- **Task:** Installing Anaconda.