Assignment 2

Job Partitioning on two machines using Greedy Algorithm

Team members: NAME - EMAIL - UID

- Aditya Mallakula – mallakula.2@wright.edu – U01093160
- Chris Davis Jaldi – jaldi.2@wright.edu – U01099335
- Vanaja Uppala – uppala.19@wright.edu – U01080568

Greedy Algorithm Pseudocode:

1. Initialization:
    - We begin by creating a list called "*free_men"*, which includes all men who are not currently engaged to a woman.
    - We also define a variable "*proposals_count"* to keep track of the number of proposals made during the algorithm.
2. Core loop:
    - The algorithm continues until there are unmarried men available.
    - For each available man, the algorithm chooses the next woman on his preference list to propose to.
    - When a man proposes to a woman, the "*proposal_count"* increases.
    - Proposal handling:
        - If the chosen woman is not already engaged, the man will become engaged to her.
        - If the woman is already engaged to another man, the algorithm determines whether she prefers the new man (the proposer) to her current partner.
            - If she prefers the new man, she breaks off her current engagement and becomes engaged to the new man.
            - If she prefers her current partner, the new man remains free and is added back to the list of "*free_men"* to propose again later.
3. Output:
    - The loop continues until all men are engaged or there are no more women to propose to.
    - When the loop is completed, the result is a stable matching of women and their partners, as well as the total number of proposals made by the algorithm.

Complexity Analysis:

- **Time Complexity:**
  - The Gale-Shapley algorithm uses a loop in which every free man proposes to women on his preference list until all men are engaged.
  - Each man can make at most n proposals (where n is the number of women), resulting in a worst-case scenario of $O(n^2)$ for n men and n women.
  - In each iteration, engagements are updated based on whether a woman prefers the current proposal to her previous engagement.
  - Reading the input file to parse preferences for men and women involves iterating through each person's list, which takes $O(n^2)$.
  - Writing the output involves iterating through the final matching results, which takes $O(n)$.
  - The Gale-Shapley algorithm is the primary contributor to the time complexity, with a total time complexity of $O(n^2)$.
- **Space Complexity:**
  - The preference lists for men and women are stored in dictionaries, resulting in a space complexity of $O(n)$ for both men and women's preferences, for a total of $O(n)$.
  - Additional data structures, such as the list of free men and the dictionary of engagements (matching), require $O(n)$ space.
  - The overall space complexity is $O(n)$.