BIRZEIT UNIVERSITY
Computer Science Department

COMP338 Artificial Intelligence

Spring 2023/2024

Assignment II

Seating Arrangement Optimization

Objective

Design and implement a program that optimizes the seating arrangement of a group of people based on their mutual dislikes. The goal is to minimize the total cost of seating people together by using three optimization techniques: **Genetic Algorithm, Simulated Annealing, and Hill Climbing**.

Problem Description

You are given a group of 10 people and a 10x10 matrix representing the dislike costs between each pair of people. The cost of seating two people together is given by the matrix, where the value at the intersection of row i and column j represents the cost of seating person i next to person j. The seating arrangement is circular, meaning the first and the last person are also considered to be seated next to each other.

Dislike Matrix

As given in the previous assignment I

Constraints

1. The seating arrangement is circular.
2. The total cost calculation should consider both directions between adjacent pairs to avoid asymmetry.
3. Implement three optimization algorithms: Genetic Algorithm, Simulated Annealing, and Hill Climbing.

BIRZEIT UNIVERSITY
Computer Science Department

COMP338 Artificial Intelligence

Spring 2023/2024

Requirements

1. Function/Method to Calculate Cost

- Implement a method/function calculate_cost(arrangement) that calculates the total cost of a given seating arrangement.

2. Genetic Algorithm

- Implement a genetic algorithm method/function genetic_algorithm(population_size=100, num_generations=1000, mutation_rate=0.1) to find an optimal seating arrangement.
- Population initialization: Randomly generate the initial population.
- Selection: Use elitism to select the top 60% of the population based on their cost.
- Crossover: Combine pairs of parents to create offspring.
- Mutation: Randomly swap positions of two people in an arrangement with a given mutation rate.

3. Simulated Annealing

- Implement a simulated annealing method/function simulated_annealing(initial_temperature=1000,cooling_rate=0.99, num_iterations=10000) to find an optimal seating arrangement.
- Initial solution: Start with a random arrangement.
- Temperature: Decrease the temperature gradually according to the cooling rate.
- Acceptance probability: Accept worse solutions with a probability depending on the current temperature to escape local minima.

4. Hill Climbing

- Implement a hill climbing method/function hill_climbing(num_restarts=100) to find an optimal seating arrangement.
- Initial solution: Start with a random arrangement.
- Neighbors: Generate neighbors by swapping pairs of people.
- Restarts: Perform multiple random restarts to explore different parts of the solution space.

5. Main Program

- Implement the main part of the program to test the three algorithms.
- Display the best seating arrangement and the corresponding total cost for each algorithm.

Expected Output

Genetic Algorithm:

Best seating arrangement: [List of names in optimal order]

Total cost: X

Simulated Annealing:

Best seating arrangement: [List of names in optimal order]

Total cost: Y

Hill Climbing:

Best seating arrangement: [List of names in optimal order]

Total cost: Z