Project Report: Timetable Optimization Using Genetic Algorithm

Introduction:

Creating an optimized classroom timetable is a complex problem that educational institutions face regularly. This project report details the development of a genetic algorithm designed to efficiently allocate rooms and schedule professors and courses while addressing various logistical and academic constraints. The project's primary objective is to maximize resource utilization, minimize scheduling conflicts, and improve the educational experience for both students and faculty.

Data Preparation:

The initial phase involves data collection and preparation. We gathered information from a CSV file containing details about professors, batches, courses, rooms, and sections. This data was processed to remove empty values and enrich it with simulated attributes, such as randomly assigned student strengths, to reflect varying room capacity needs. This preprocessing step is crucial as it ensures the integrity and usability of the data for the subsequent steps of the algorithm.

Genetic Algorithm Design:

The genetic algorithm (GA) used in this project follows the standard GA process, which includes initialization, selection, crossover, mutation, and fitness evaluation.

Gene Representation:

In our GA, each timetable slot is treated as a gene, and a complete timetable is an individual chromosome. The attributes of a gene include the professor, batch, course, section, room, timeslot, and day, encoded in a binary format to facilitate genetic operations.

Initial Population:

The algorithm begins by generating an initial population of random timetables. Each timetable consists of a predefined number of slots, adhering to the total number of available times, professors, courses, and rooms.

Fitness Evaluation:

The fitness function evaluates how well a timetable meets the defined constraints. It applies penalties for any violations, such as:

- **Constraint 1: Free Classroom Allocation**
    - Penalize if classes are scheduled in classrooms that are already occupied.
- **Constraint 2: Classroom Capacity**
    - Penalize if the number of students assigned to a classroom exceeds its capacity.
- **Constraint 3: Professor Overlap**

    - Penalize if a professor is scheduled for more than one lecture simultaneously.
- **Constraint 4: Room Collision**
    - Penalize if the same course and section are assigned to different rooms at the same time slot.
- **Constraint 5: Maximum Courses per Professor**
    - Penalize if a professor is assigned more than 3 courses across different sections.
- **Constraint 6: Section Course Limit**
    - Penalize if a section is assigned more than 5 courses in a semester.
- **Constraint 7: Course Lecture Distribution**
    - Penalize if a course has fewer or more than two lectures per week, not scheduled on the same or adjacent days.
- **Constraint 8: Lab Lecture Sequence**
    - Penalize if lab lectures are not scheduled in two consecutive time slots.

Selection Process:

After evaluating the fitness of all timetables in the population, the algorithm selects the best-performing ones. This selection process is critical as it ensures that only the timetables with the highest fitness scores contribute their genes to the next generation.

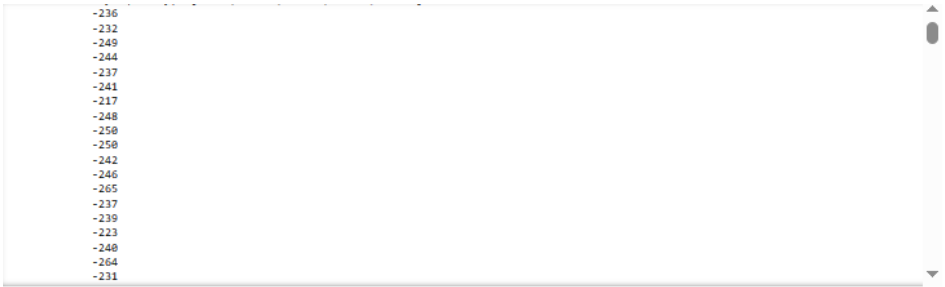
Crossover and Mutation:

Crossover involves combining genes from two parent timetables to create offspring. This process allows for the mixing of good genes from different parents to potentially produce superior offspring. Mutation, albeit rare, is introduced randomly to maintain genetic diversity within the population, helping prevent premature convergence to local optima.

Results:

Initial Fitness Scores:

1:



2:

```
Fitness Score :  -251
Fitness Score :  -252
Fitness Score :  -230
Fitness Score :  -250
Fitness Score :  -247
Fitness Score :  -238
Fitness Score :  -256
Fitness Score :  -246
Fitness Score :  -258
Fitness Score :  -240
Fitness Score :  -241
Fitness Score :  -247
Fitness Score :  -256
Fitness Score :  -248
Fitness Score :  -236
Fitness Score :  -250
Fitness Score :  -259
Fitness Score :  -229
Fitness Score :  -227
```

Final Fitness Score:

1:

```
5
Slot : 47, Professor: Prof2, Batch: Batch2, Course: Cor3, Section: B, Strength: 72, Room: Room13, Time Slot: Slot3, Day: Day
5
Slot : 48, Professor: Prof3, Batch: Batch1, Course: Cor4, Section: D, Strength: 61, Room: Room18, Time Slot: Slot4, Day: Day
5
Slot : 49, Professor: Prof4, Batch: Batch3, Course: Cor1, Section: D, Strength: 78, Room: Room12, Time Slot: Slot5, Day: Day
5
Slot : 50, Professor: Prof1, Batch: Batch4, Course: Cor4, Section: A, Strength: 51, Room: Room6, Time Slot: Slot5, Day: Day5
Slot : 51, Professor: Prof2, Batch: Batch2, Course: Cor3, Section: A, Strength: 72, Room: Room15, Time Slot: Slot2, Day: Day
5
Slot : 52, Professor: Prof12, Batch: Batch1, Course: Cor4, Section: D, Strength: 61, Room: Room16, Time Slot: Slot1, Day: Da
y5
Slot : 53, Professor: Prof20, Batch: Batch2, Course: Cor2, Section: A, Strength: 61, Room: Room18, Time Slot: Slot6, Day: Da
y5
Slot : 54, Professor: Prof15, Batch: Batch1, Course: Cor4, Section: D, Strength: 68, Room: Room12, Time Slot: Slot6, Day: Da
y5
Slot : 55, Professor: Prof20, Batch: Batch1, Course: Cor2, Section: C, Strength: 45, Room: Room3, Time Slot: Slot1, Day: Day
5
Fitness Score: -132
```

2:

```
5
Slot : 39, Professor: Prof6, Batch: Batch2, Course: Cor2, Section: A, Strength: 53, Room: Room2, Time Slot: Slot4, Day: Day5
Slot : 40, Professor: Prof3, Batch: Batch1, Course: Cor1, Section: B, Strength: 50, Room: Room3, Time Slot: Slot1, Day: Day5
Slot : 41, Professor: Prof19, Batch: Batch1, Course: Cor4, Section: E, Strength: 51, Room: Room4, Time Slot: Slot6, Day: Day
5
Slot : 42, Professor: Prof6, Batch: Batch3, Course: Cor1, Section: B, Strength: 54, Room: Room5, Time Slot: Slot6, Day: Day5
Slot : 43, Professor: Prof20, Batch: Batch1, Course: Cor3, Section: D, Strength: 58, Room: Room7, Time Slot: Slot6, Day: Day
5
Slot : 44, Professor: Prof13, Batch: Batch4, Course: Cor3, Section: C, Strength: 64, Room: Room20, Time Slot: Slot1, Day: Da
y5
Slot : 45, Professor: Prof1, Batch: Batch1, Course: Cor1, Section: E, Strength: 48, Room: Room9, Time Slot: Slot4, Day: Day5
Slot : 46, Professor: Prof14, Batch: Batch4, Course: Cor3, Section: A, Strength: 41, Room: Room4, Time Slot: Slot2, Day: Day
5
Slot : 47, Professor: Prof1, Batch: Batch2, Course: Cor3, Section: C, Strength: 54, Room: Room3, Time Slot: Slot3, Day: Day5
Slot : 48, Professor: Prof6, Batch: Batch2, Course: Cor1, Section: E, Strength: 53, Room: Room10, Time Slot: Slot5, Day: Day
5
Slot : 49, Professor: Prof17, Batch: Batch1, Course: Cor3, Section: C, Strength: 50, Room: Room10, Time Slot: Slot3, Day: Da
y5
Fitness Score: -126
```

The fitness scores from the genetic algorithm, ranging from -236, -251 to -132, -126 respectively show an improvement over time, indicating that the algorithm is progressively reducing the number of scheduling constraint violations. This progression suggests that the algorithm is on a path to achieving more optimized timetables as it continues to iterate.

Conclusion:

The project successfully demonstrates the application of a genetic algorithm to optimize classroom timetables, significantly improving resource management and scheduling efficiency. The potential for future applications of this algorithm in other scheduling tasks or its adaptation for larger educational institutions presents an exciting avenue for further research and development.