

Code Overview

This code implements a genetic algorithm to generate a schedule for a set of courses in which each course is assigned a time slot and an exam hall. The goal is to minimize the number of student conflicts and the penalty for exceeding exam hall hours or having conflicting exams in the same time slot.

The main steps of the algorithm are:

1. Generate an initial population of random schedules.
2. Evaluate the fitness of each schedule based on the number of student conflicts and penalty.
3. Select the best schedules using tournament selection.
4. Create new offspring schedules by performing single-point crossover and mutation.
5. Replace the old population with the new population and repeat steps 2-4 until a satisfactory solution is found or the maximum number of generations is reached.

The fitness function calculates the total number of student conflicts and penalty for each schedule. It iterates over all pairs of conflicting courses and checks if they are scheduled in the same time slot and exam hall. It also checks if a course exceeds the allowed exam hall hours and if there are any conflicting exams in the same time slot but different exam halls.

The code prints the best schedule found and its fitness score at each generation. If a schedule with zero conflicts is found, it is immediately returned as the best solution. Otherwise, the algorithm terminates after the specified number of generations.

Solution Photo

```
File Edit Selection View Go Run Terminal Help
i200441_A2.py - Visual Studio Code

Welcome
i200441_A2.py X
C:\Users\shehe>Downloads\AI_A2>i200441_A2.py > ...
11 ('C2', 'C3'): 7,
12 ('C3', 'C4'): 12,
13 ('C4', 'C5'): 0

PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL

scode\extensions\ms-python.python-2023.4.1\pythonFiles\lib\python\debugpy\launcher '65068' '-...' 'c:\Users\shehe\Downloads\AI_A2\i200441_A2.py'

Generation Number: 0
Schedule: [('C1', 'T1', 'H2'), ('C2', 'T1', 'H2'), ('C3', 'T3', 'H2'), ('C4', 'T3', 'H1'), ('C5', 'T3', 'H2')] Fitness score: -260
Schedule: [('C1', 'T1', 'H2'), ('C2', 'T3', 'H1'), ('C3', 'T2', 'H1'), ('C4', 'T3', 'H1'), ('C5', 'T3', 'H2')] Fitness score: -230
Schedule: [('C1', 'T1', 'H1'), ('C2', 'T3', 'H1'), ('C3', 'T3', 'H1'), ('C4', 'T1', 'H2'), ('C5', 'T1', 'H1')] Fitness score: -226
Schedule: [('C1', 'T2', 'H2'), ('C2', 'T3', 'H1'), ('C3', 'T3', 'H2'), ('C4', 'T2', 'H1'), ('C5', 'T3', 'H2')] Fitness score: -224
Schedule: [('C1', 'T2', 'H1'), ('C2', 'T2', 'H2'), ('C3', 'T2', 'H1'), ('C4', 'T2', 'H1'), ('C5', 'T1', 'H2')] Fitness score: -154
Schedule: [('C1', 'T2', 'H1'), ('C2', 'T2', 'H2'), ('C3', 'T1', 'H2'), ('C4', 'T2', 'H1'), ('C5', 'T3', 'H1')] Fitness score: -130
Schedule: [('C1', 'T3', 'H2'), ('C2', 'T2', 'H1'), ('C3', 'T3', 'H2'), ('C4', 'T1', 'H2'), ('C5', 'T2', 'H2')] Fitness score: -114
Schedule: [('C1', 'T2', 'H1'), ('C2', 'T3', 'H2'), ('C3', 'T2', 'H1'), ('C4', 'T1', 'H1'), ('C5', 'T3', 'H1')] Fitness score: -114
Schedule: [('C1', 'T2', 'H2'), ('C2', 'T3', 'H2'), ('C3', 'T2', 'H1'), ('C4', 'T3', 'H2'), ('C5', 'T3', 'H2')] Fitness score: -30
Schedule: [('C1', 'T3', 'H2'), ('C2', 'T1', 'H2'), ('C3', 'T2', 'H1'), ('C4', 'T2', 'H1'), ('C5', 'T3', 'H2')] Fitness score: -24
Generation Number: 1
Schedule: [('C1', 'T3', 'H2'), ('C2', 'T1', 'H2'), ('C3', 'T2', 'H2'), ('C4', 'T2', 'H1'), ('C5', 'T3', 'H2')] Fitness score: -124
Schedule: [('C1', 'T2', 'H1'), ('C2', 'T3', 'H2'), ('C3', 'T2', 'H1'), ('C4', 'T1', 'H2'), ('C5', 'T3', 'H1')] Fitness score: -114
Schedule: [('C1', 'T2', 'H2'), ('C2', 'T3', 'H2'), ('C3', 'T2', 'H1'), ('C4', 'T3', 'H2'), ('C5', 'T3', 'H2')] Fitness score: -30
Schedule: [('C1', 'T3', 'H2'), ('C2', 'T1', 'H2'), ('C3', 'T2', 'H1'), ('C4', 'T2', 'H1'), ('C5', 'T3', 'H2')] Fitness score: -24
Schedule: [('C1', 'T3', 'H2'), ('C2', 'T1', 'H2'), ('C3', 'T2', 'H1'), ('C4', 'T2', 'H1'), ('C5', 'T3', 'H2')] Fitness score: -24
Schedule: [('C1', 'T2', 'H1'), ('C2', 'T2', 'H1'), ('C3', 'T2', 'H1'), ('C4', 'T1', 'H1'), ('C5', 'T3', 'H1')] Fitness score: -20
Schedule: [('C1', 'T2', 'H1'), ('C2', 'T3', 'H1'), ('C3', 'T2', 'H1'), ('C4', 'T1', 'H1'), ('C5', 'T3', 'H1')] Fitness score: -14
Schedule: [('C1', 'T3', 'H1'), ('C2', 'T1', 'H2'), ('C3', 'T3', 'H1'), ('C4', 'T2', 'H1'), ('C5', 'T3', 'H2')] Fitness score: 0
Schedule: [('C1', 'T3', 'H2'), ('C2', 'T2', 'H1'), ('C3', 'T3', 'H2'), ('C4', 'T1', 'H2'), ('C5', 'T3', 'H2')] Fitness score: 0
Best schedule with minimum conflicts:
C1: T3, H1
C2: T1, H2
C3: T3, H1
C4: T2, H1
C5: T3, H2
Minimum conflicts: 0

Ln 14, Col 2 (45 selected) Spaces: 4 UTF-8 CRLF Python 3.10.8 64-bit 4:49 AM 3/23/2023
```