Code output is on the last page and my program is at

1) There were a lot of challenges while writing this program. Just in the order they came up: Firstly getting started was daunting and I wasn't exactly sure where to start. I ended up first just copying the provided data into data structures that I think would be easiest to work with. This helped me get started, however it may have ended up hurting me because I created the data structures before I knew exactly how I would be using them. Next creating the fitness function was also challenging, just with so many conditions it was difficult to sufficiently test that it was working correctly. It was also hard to check for some of the conditions, namely the consecutive classes conditions, I ended up figuring it out but my solution feels sloppy and bloated. To help test my fitness function easier I ended up just re-using a premade schedule that I would tweak to fit my cases. Lastly, actually integrating the selection, mutation, and crossover was challenging to get started and completely figure out how to implement it. But once I wrote my crossover function it started to all come together and was one of the more satisfying parts of this program.

2) I think the schedule my program produced looks appropriate. It's kind of hard to judge the schedules, just because they're fake and it's sorta difficult to actually rate them based on my intuition. However that being said, there's nothing about the randomly created schedules that seem off to me.

3) To improve my program I would definitely work on and improve my fitness function more. It ended up being kind of convoluted with a lot of different data structures being used to calculate the score. If I were to redo it I would try to design my data structures around the problem, rather than designing the problem around the data structures. I would also try to improve the fitness function more, perhaps by building a way to better analyze / view the resulting schedules. To figure out what common patterns are occurring and try to change the fitness function to accommodate that. To further improve my program I think I would tweak the mutation algorithm as well. I think there's potential to create a more unique / intricate mutation versus what I implemented.

4) Overall the best score I found was a 33. I may have implemented something incorrectly, I didn't completely understand making a pdf function of the individual activity scores, so I settled for a sum of the results. I also ran into some challenges by scoring individual activities versus scoring the entire schedule at once. I'm sure this leads to some variety in results. But overall this was a fun assignment to complete!

```
C:\Users\micha\Documents\Classwork\cs462\genetic-algo>python program2.py
Genetic algorithm ran 5 times, here are the results ordered
32.7
32.7
32.4
32.1
31.599999999999998

Best schedule has been written to results.txt
```

```
C:\Users\micha\Documents\Classwork\cs462\genetic-algo>python program2.py
Genetic algorithm ran 5 times, here are the results ordered
33.0
32.7
32.7
32.1
30.9

Best schedule has been written to results.txt
```

```
C:\Users\micha\Documents\Classwork\cs462\genetic-algo>python program2.py
Genetic algorithm ran 5 times, here are the results ordered
33.0
32.7
32.7
32.7
32.4
```

Here's one of my best schedules that gets written to the file:

{'SLA100A': {'room': 'Loft 310', 'time_slot': 11, 'facilitator': 'Banks'},
'SLA100B': {'room': 'Loft 310', 'time_slot': 11, 'facilitator': 'Glen'},
'SLA191A': {'room': 'Loft 310', 'time_slot': 12, 'facilitator': 'Glen'},
'SLA191B': {'room': 'Frank 119', 'time_slot': 12, 'facilitator': 'Banks'},
'SLA201': {'room': 'Roman 201', 'time_slot': 15, 'facilitator': 'Banks'},
'SLA291': {'room': 'Loft 310', 'time_slot': 11, 'facilitator': 'Lock'},
'SLA303': {'room': 'Loft 206', 'time_slot': 10, 'facilitator': 'Banks'}
, 'SLA304': {'room': 'Beach 301', 'time_slot': 14, 'facilitator': 'Tyler'},
'SLA394': {'room': 'Beach 201', 'time_slot': 10, 'facilitator': 'Singer'}
, 'SLA449': {'room': 'Beach 201', 'time_slot': 12, 'facilitator': 'Shaw'}
, 'SLA451': {'room': 'Loft 310', 'time_slot': 14, 'facilitator': 'Shaw'}}