Program 2 Report – Genetic algorithm

There are challenges in writing the program and it did not go smoothly from the beginning. At first the creation of the initial population was easy as it was all random but to evaluate every single condition was a bit challenging and needed thorough planning on how the population must be made to make sure every single condition can be checked. Therefore, I used parallel arrays to create a single random class schedule and then repeated it to make an array of the population.

About checking instructor load, I had to check how many classes each instructor is teaching but it would have been very difficult and computationally expensive to check what class is being taught by whom by having a special variable for each instructor and this would have made the code very redundant, so I decided to keep using parallel arrays this time an array for faculty as given and one for each's number of classes taught, I created an integer array to keep track of the number of classes being taught by each instructor and the index of an instructor's number of classes is the same as the instructor's name in the instructor's array. This way I was able to simultaneously access instructor's name and number of courses taught at the same index of two different arrays. Originally number of courses taught was all zeroes and for any instructor is teaching a class it will get incremented by one, thus giving us an array of instructor's load that we can use to compare with the given limits and change the fitness score accordingly.