# **Assignment 3**

**Report: Exam Schedule Generation by Searching (Local and Evolutionary)**

**Section: Artificial Intelligence-F**

|  |  |
| --- | --- |
| Members Names | Roll no |
| Muhammad Shazar | 17L-4177 |
| Muhammad Hammad Maqsood | 17L-4417 |

# **Introduction:**

In this programming assignment, genetic algorithm combined with local search was used to solve the scheduling problem.

A **genetic algorithm** is a search heuristic that is inspired by Charles Darwin’s theory of natural evolution. This algorithm reflects the process of natural selection where the fittest individuals are selected for reproduction in order to produce offspring of the next generation.

There were five phases that were used to compute results.

1. Initial population
2. Fitness function
3. Selection
4. Crossover
5. Mutation
6. Local Search

# **Techniques used:**

# **Population:**

In the file general.info there were three days and seven slots per day given to us, so we made 2d array of [3][7] and at every index of it there was an array list representing the courses ids whose exams were to be held. That was our single chromosome which was populated using random numbers.

* **Fitness Function:**

While computing evaluation of given chromosome we kept in mind following points and assign fitness value according to its importance

* All exams must be scheduled within the given number of days.
* Total students taking exam in one given slot must be less than the total room capacity.
* Number students having two exams in one given slot must be minimized.
* Number of students having exams on two consecutive slots must be minimized.
* Not even a single student can have more than two exams in one slot.
* Not even a single student can have more than two exams in consecutive lot.
* Not even a single student can have more than three exams in one given day.

**Approach:**

1. In case number 1 mention in above points, we have set the limit to by making of array of given number of days in GeneralInfo.txt and populate it accordingly and we make sure it that each and every exam must be scheduled in given days.
2. In case number 2, as I discuss above our approach to solve this problem. We basically make List type array which contain students of given courses List at given time slot we find the count of total student and compare the room capacity with it.
3. In case 3 and case 5 we basically use same approach. We use same technique as mention in point 2, get the list of students of each course in given slot and then simply compare it.
4. In case 7, we simply count the student id count in given day, if it exceeds then 3, we simply assume that this particular student have three exams in given day.

All these points which are mentioned above are taken into consideration while computing fitness of chromosome.

# **Selection:**

Since our population consisted of numerous chromosomes, we choose the best for further evaluation using their fitness values.

# **Crossover:**

Crossover was done using two of the best chromosomes, their courses present in the array lists were swapped with each other.

# **Mutation:**

Using some random probability mutation was done on the fittest chromosome by swapping courses within.

# **Experimental Result:**

Here we generate random population of almost 40 and represent it in the given table below with its fitness value.

|  |  |
| --- | --- |
|  | Initial Values |
|  | 5522 |
|  | 4312 |
|  | 4198 |
|  | 4339 |
|  | 4836 |
|  | 5394 |
|  | 3448 |
|  | 4701 |
|  | 4215 |
|  | 6239 |
|  | 3344 |
|  | 4735 |
|  | 3375 |
|  | 5155 |
|  | 3379 |
|  | 2459 |
|  | 2971 |
|  | 4502 |
|  | 6467 |
|  | 3765 |
|  | 1827 |
|  | 4828 |
|  | 4785 |
|  | 4384 |
|  | 4296 |
|  | 5841 |
|  | 4181 |
|  | 4176 |
|  | 2564 |
|  | 3716 |
|  | 3913 |
|  | 5339 |
|  | 4078 |
|  | 1828 |
|  | 4617 |
|  | 4742 |
|  | 4081 |
|  | 5219 |
|  | 3923 |
|  | 5023 |

Best Fitness value of Generation 2 = 7305

Best Fitness value of Generation 3 = 10052

Best Chromosome id= 22

Where 9000 fitness value was the threshold to accept the timetable.