# Automated Scheduler of Teaching Load

**[1]Virti Parekh, [2]Yash Jain, [3]Chintan Doshi**

*[1]Student, [2]Student, [3]Student*
*[1,2,3]Information Technology,*
*[1,2,3]Shah & Anchor Kutchhi Engineering College, Mumbai, India*

---

***Abstract :*** *Managing subjects', lectures, and classrooms are complex issues in universities and learning institutions. Programs, subjects' management, and timetabling are hectic and time-consuming tasks that create multi-dimensional and highly constrained issues in learning institutions. The scheduler generation system aims to generate a load sheet in excel format and a scheduler of teaching load in tabular format. It takes a number of semesters, subjects, labs, faculties, classrooms, and workloads as input. Creating an educational timetable manually required a lot of time, several resources, and numerous rounds of changes before it has been finalized. Assigning multiple subjects to one lecturer and allocating multiple classrooms for different subjects are tough tasks as well and need a solid framework and comprehensive solution. Various approaches have been made in the past decade to solve the problem of constructing timetables for colleges. In our system, this problem is formulated as a constraint satisfaction problem and we try various approaches that are capable of handling both hard and soft constraints. Automatic generation of the timetable is an attractive approach as compared to a manual approach. We have used many frameworks to design and develop an automatic timetable system that has a graphical user-friendly and attractive interface. The developed system has a flexible representation and appropriate methods to create a feasible, automatic timetable and avoid clashes. It is tedious undertakings that can prompt exceptionally obliged issues in foundations. Making a plan physically requires a few assets and various rounds of changes prior to affirming it. Allotting numerous subjects to anyone instructor and distributing various study halls for various subjects are troublesome undertakings to oversee and for this, we will require a strong arrangement and extensive arrangement. These issues and challenges will be explained in our proposed framework which will create the plan automatically.*

***IndexTerms*** **- Timetable, Django, MySql, Python, Web Application, Load sheet, Excel sheet, Genetic Algorithm.**

---

## I. INTRODUCTION

The timetabling in institutions are fundamentally the way toward booking and appointing the talks into suitable occasions parts and designate assets separately without causing time conflicts for understudies and instructors. At present, numerous managements are utilizing manual cycles for making plans which is an exceptionally confusing, convoluted, and troublesome assignment. It requires some investment for booking even the littlest imperatives and the case becomes far more detestable if the quantity of limitations increments. Indeed, even the impeccably created schedule is utilized for a long time with no changes. While making the schedule they need to think about accessible workforce, courses, rooms, labs, and working days. It turns out to be more muddled when variables like elective courses, instructor's time allotments, and the speaker study hall are to be overseen with no errors.[1] Therefore dependent on the referenced limitations planning a course schedule is a thorough, complex, and tedious assignment. The manual talk plan planning requests significant time and endeavours. The talk plan planning is an imperative fulfilment issue in which we discover an answer that fulfils the given arrangement of limitations. Indeed, even the completely planned time table is reused for the entire age with no changes, ending up being dull in such circumstances. Different cases are caused in light of the fact that the issue is the number of managers or faculties continues changing, this outcome in the rescheduling of time tables critically. [1][2]

Automated Scheduler of teaching load is a webapplication based software which is used to generate timetables automatically. Automatic Scheduler of Teaching Load is a web application based programming which is utilized to create plans automatically. It will assist with dealing with all the periods naturally. The most extreme and least outstanding task at hand for a Faculty for a day and week will be indicated for the proficient age of plan. Plan Scheduler focuses to create programming for institution to deal with the "Timetable Formation" for the staff. The top of each Department has an issue in assigning work to their subordinates and reacting to the work position. Additionally, information is accessible for the administration of schedules since paper based work isn't transferred to an incorporated framework.

Time taken for data collection reduces the effective time. The vast majority of the college work is PC based. Everything is done physically. This framework soaks up the nature of time needed for making the time table. This framework finds the answer for each issue as it is easy to use. Notice is given at whatever point an update is required. The Time age is the most Fundamental movement in any Educational organization. It is additionally the most troublesome and tedious cycle. [2]

Most departments have various courses and each course has various subjects. Presently there are restricted resources, every workforce showing more than one subject. In our Timetable Generation calculation we propose to use a schedule object. This item involves Classroom objects and the plan for each them similarly a wellness score for the schedule. Based on predefined data and with few changes regarding change in subjects and faculty a schedular is generated.

## II. TYPES OF CONSTRAINTS

*Hard Constraints :*

In timetabling problem the hard constraints must be fulfilled to accomplish an attainable plan. The hard requirements in this difficult model can be seen from alternate points of view also. The limitations are as per the following:[3]

- Lectures should not have timeslot clashes.
- None of the room could be used for two different lectures at same time slot.
- None of the lecturer should have two classes at same time slot.
- No time clashes for the faculty.
- Every class must be scheduled exactly once.
- Each lecturer has their minimum and maximum limit of weekly working hours.
- The framework produces middle of the road level also numerous last reports including week after week time table, educator plan, room insightful time table, understudy time table, office level time table and so on.

*Soft Constraints:*

In timetabling, the soft issues make low values outputs. The infringement of the soft imperatives won't make the schedule lose its fitness. Soft Constraints are as per the needs of schedule in making and required predefined rules made by management:

- There should not be too many empty slots between the lectures.
- The working hours should be according to teaching load.
- Some classes require specific rooms. For instance, tests may be held specifically labs.[6]
- Break should be scheduled.
- The total number of available hours for every week.
- The workload on all teachers should be uniform.

## III. LITERATURE SURVEY

There exist different schedule age issues, for example, University Timetabling, Employee Timetabling, Sports Timetabling and Examination Timetabling, semester insightful timetabling, school shrewd timetabling.

There are different methods to generate scheduler such as – Group technique, Sequential strategy, Meta-Heuristics and Constraint Based strategy. Meta Heuristics is a more elevated level strategy which is utilized to give adequate answers for advancement issues. On some class of issues, they don't ensure an internationally ideal arrangement. This strategy is utilized when the traditional strategies are excessively ease back or neglect to give an answer. This is accomplished at the expense of optimality and exactness for speed. In this paper we consider the accompanying Meta-Heuristic techniques.[4][5]The various algorithms to carry such task are explained in brief:

*A. Simulated annealing (SA):*

Simulated annealing uses probability technique utilized for examples that has ideal given capacity or has low capacity. Deliberately, it is a metaheuristic to genuinely precise worldwide improvement in a tremendous inquiry space. It is regularly utilized when the inquiry space is particular. Simulated annealing is a procedure for finding a decent outcome to an advancement problem. On the off chance that there is where we need to expand or decrease something, our concern can almost certainly be tackled with reenacted strengthening.[3]

*B. Tabu Search (TS):*

Tabu Search is a Global Optimization calculation and a Metaheuristic or Meta-procedure for figuring an encompassed heuristic technique. Forbidden Search is a parent for relations of sub-ordinate methodology that set up memory structure in Metaheuristic, for example, Tabu Search and Parallel Tabu Search.

Forbidden Search fundamentally tries not to get caught at adjacent relations. That is the reason this inquiry permits non-improving moves when it is caught in nearby optimization. Another favorable position of Tabu Search strategy is that it forestalls cycling back to the recently visited arrangements by the utilization of recollections in this way making more odds of progress.

*C. Genetic Algorithm (GA) :*

Genetic Algorithms are inspired by Darwin's developmental hypothesis. GA goes under the class of Evolutionary calculations that utilization the guideline of normal determination to infer a bunch of arrangements towards the ideal arrangement. It is a pursuit heuristic which creates answers for advancement issues utilizing methods motivated by normal development like transformation, legacy, hybrid and choice.[5]

The productivity of the genetic algorithm calculation predominantly relies on fitness work. Change and hybrid are the two principal parts of the calculation. They are known as the administrators of GA. The term of Crossover is the acknowledgment of the deterministic part of algorithm. From the recently chosen arrangements, two-parent arrangements are chosen for breeding. The new arrangement is acquired by the strategy for hybrid and change and offers numerous qualities of the parent arrangement. Unusual parents are chosen for each new child and this cycle proceeds until a population of suitable predefined size is reproduced. Simply the best arrangements from the past arrangements are chosen for breeding, alongside few fewer fitness answers to guarantee hereditary diversity.[6] The making of another population can be halted when an answer that fulfills the base rules can be found as the fitness would be considered. For our situation, the cycle can be halted when a schedule fulfilling all the hard and soft constraints are fulfilled.

ADVANTAGES OF GENETIC ALGORITHM

- Does not need any by-product info (which might not be accessible for several real-world problems).
- Is faster and additional economical as compared to the standard ways.
- Has superb parallel capabilities.
- Optimizes each continuous and separate functions and additionally multi-objective issues.
- Provides a listing of "good" answers and not simply one solution.
- Always gets a solution to the matter that gets higher over the time.
- Useful once the search area is incredibly massive and measures an oversized range of parameters concerned.

LIMITATION OF GENETIC ALGORITHM

- GA's are not suited for all problems, especially problems which are simple and for which derivative information is available.
- Fitness Value is calculated repeatedly which might be computationally time taking for some problems.
- Having random probability distribution systems, there is no guarantee on the optimality or the quality of the solution.
- If not implemented properly, the Genetic Algorithm may not converge to the optimal solution.

STEPS OF GENETIC ALGORITHM

STEP 1 - This step starts with guessing of initial sets of a and b values which may or may not include the optimal values. These sets of values are called as 'chromosomes'.

STEP 2 - This step starts with rigorous calculation of initial sets of a and b values which can or might not embody the best values. These sets of values are known as 'chromosomes' and also the step is termed 'initialize population'

STEP 3 - This step is 'crossover'. During this step, chromosomes area unit expressed in terms of genes. This could be done by changing the values of a and b into binary strings which implies the values got to be expressed in terms of zero or one

STEP 4 - This step is named 'mutation'. Mutation is that the method of fixing the worth of factor i.e to switch the worth one with zero and vice-versa. For instance, if offspring body is [1,0,0,1], once mutation it becomes [1,1,0,1]. Here, 2d price of the offspring body is determined to urge mutated. It's got modified to one from zero.

STEP 5 - After mutation, binary chromosomes are regenerate into whole number type and fitness values are calculated. If anyone of the chromosomes produces target fitness price capable zero, we tend to stop there. Otherwise, the method is going to be recurrent from step - two to step - four by equation mutated chromosomes with new population

OVERVIEW:

Django is a popular python open-source web development and clean, pragmatic design. It is a robust and approachable framework that lets you focus on your application by having salient parts pre-baked that are fairly standard practice. This framework helps to focus on writing apps instead of reinventing the wheel. It is used in all sorts of tech stacks, including Instagram, Pinterest, Mozilla and Eventbrite.

We have made a user interface in which we have made a role based access which is very simple and easy for end users. In these, role based access is basically used to give access to only admin which will help him to edit, delete and other operations. Other users can only see the entries made but does not have the right to do any changes.

We have generated the timetable semester wise using some of the hard constraints. We have also added the theory slots and lab slots of two hours according to the slots given to faculty members. Currently we are making the timetable for only IT department that too from second year to fourth year. Once it gets approved we will make it for other department also. We have also filled the data of how many class will be having how many lectures in a week and what slots are allocated to which faculty member, following all these condition the timetable will be automatically generated.

## IV. PROPOSED WORK

- This framework will assist the universities with producing the schedule consequently with no manual outstanding burden and spares a ton of time that assumes a conspicuous part in the current way of life.
- The primary reason for the framework is to execute a compelling shrewd framework to create and keep up the schedules.
- A site and a legitimate UI was given to administrator and workforce to login with their certifications and do their individual duties.[3]

- The extent of the framework incorporates age of schedule, access the accessible timetable(s), alter/erase a current plan. Add/change personnel subtleties of an office. Add/change semester astute subjects subtleties. Add/change time booking and period assignment measure.
- By utilizing the Genetic Algorithm we can lessen the time needed to create a scheduler and it is more exact, exact, and liberated from human blunders.
- The First stage contains all the normal necessary classes of the establishment, which are planned by a focal group.[2][4]

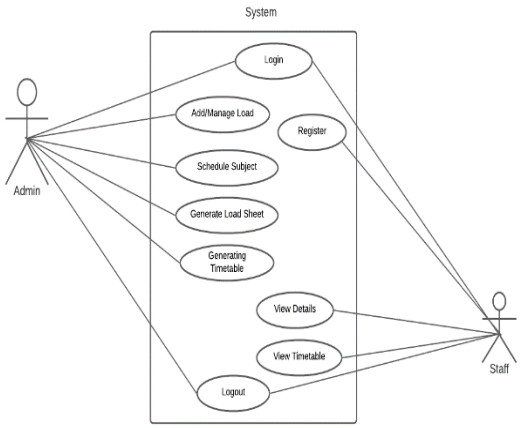



Fig 1: Use case diagram

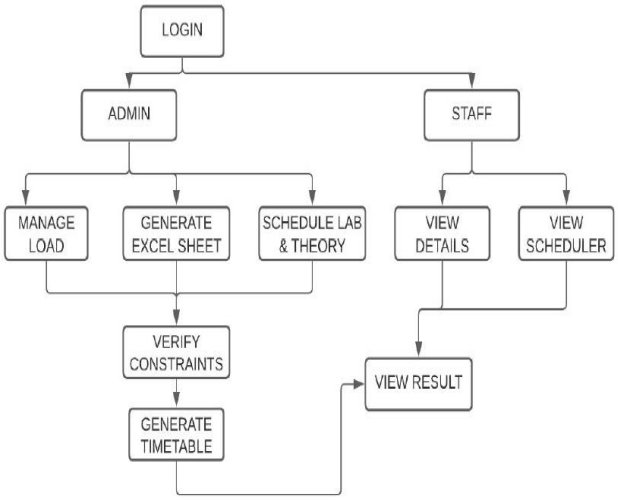


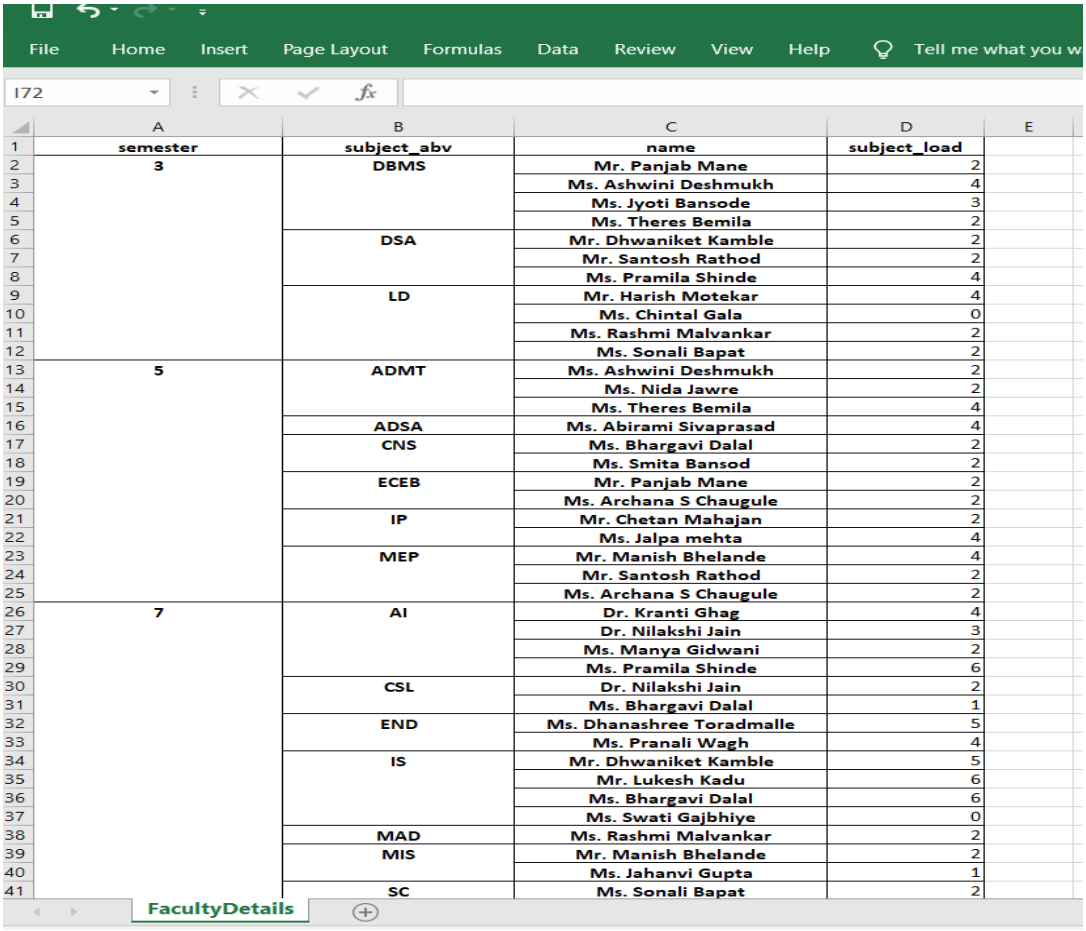

Fig 2: Project Model

- There are few points that justify the need of the system.
- User friendly
- Faster and better generation of timetable.
- Saving time and manpower.
- It saves Time and Effort.

## V. RESULT ANALYSIS

### 5.1 Results of Descriptive Statics of Study Variables

Automatic Schedular is basically soft computing but as a part of implementation we added some more functionalities which are like getting load of faculties in excel sheet through database and also altered changes made in data regarding faculties or lectures or load all changes or modification can be seen just by a click in an excel sheet which makes this project as a part of machine learning exaggerating it is supervised type of machine learning as we know the expected output of an institute schedule.



File Home Insert Page Layout Formulas Data Review View Help Tell me what you w

I72

| | A | B | C | D | E |
|---|---|---|---|---|---|
| 1 | semester | subject_abv | name | subject_load | |
| 2 | 3 | DBMS | Mr. Panjab Mane | 2 | |
| 3 | | | Ms. Ashwini Deshmukh | 4 | |
| 4 | | | Ms. Jyoti Bansode | 3 | |
| 5 | | | Ms. Theres Bemila | 2 | |
| 6 | | DSA | Mr. Dhwaniket Kamble | 2 | |
| 7 | | | Mr. Santosh Rathod | 2 | |
| 8 | | | Ms. Pramila Shinde | 4 | |
| 9 | | LD | Mr. Harish Motekar | 4 | |
| 10 | | | Ms. Chintal Gala | 0 | |
| 11 | | | Ms. Rashmi Malvankar | 2 | |
| 12 | | | Ms. Sonali Bapat | 2 | |
| 13 | 5 | ADMT | Ms. Ashwini Deshmukh | 2 | |
| 14 | | | Ms. Nida Jawre | 2 | |
| 15 | | | Ms. Theres Bemila | 4 | |
| 16 | | ADSA | Ms. Abirami Sivaprasad | 4 | |
| 17 | | CNS | Ms. Bhargavi Dalal | 2 | |
| 18 | | | Ms. Smita Bansod | 2 | |
| 19 | | ECEB | Mr. Panjab Mane | 2 | |
| 20 | | | Ms. Archana S Chaugule | 2 | |
| 21 | | IP | Mr. Chetan Mahajan | 2 | |
| 22 | | | Ms. Jalpa mehta | 4 | |
| 23 | | MEP | Mr. Manish Bhelande | 4 | |
| 24 | | | Mr. Santosh Rathod | 2 | |
| 25 | | | Ms. Archana S Chaugule | 2 | |
| 26 | 7 | AI | Dr. Kranti Ghag | 4 | |
| 27 | | | Dr. Nilakshi Jain | 3 | |
| 28 | | | Ms. Manya Gidwani | 2 | |
| 29 | | | Ms. Pramila Shinde | 6 | |
| 30 | | CSL | Dr. Nilakshi Jain | 2 | |
| 31 | | | Ms. Bhargavi Dalal | 1 | |
| 32 | | END | Ms. Dhanashree Toradmalle | 5 | |
| 33 | | | Ms. Pranali Wagh | 4 | |
| 34 | | IS | Mr. Dhwaniket Kamble | 5 | |
| 35 | | | Mr. Lukesh Kadu | 6 | |
| 36 | | | Ms. Bhargavi Dalal | 6 | |
| 37 | | | Ms. Swati Gajbhiye | 0 | |
| 38 | | MAD | Ms. Rashmi Malvankar | 2 | |
| 39 | | MIS | Mr. Manish Bhelande | 2 | |
| 40 | | | Ms. Jahanvi Gupta | 1 | |
| 41 | | SC | Ms. Sonali Bapat | 2 | |

FacultyDetails

Fig 3: Load Sheet in Excel Format

The most important part is choosing framework which has a easy database retrievals and has systematic approach as ASTL needs a simple user interface hence we selected Django framework and started by creating forms for the input and when we had all the forms ready for adding the input we started with data entry and had a labelled data which would be required for the functioning of the algorithm. The data was used in such a way that if the management wanted to used data for generating timetable they would just modify the data and the whole load sheet would be ready.

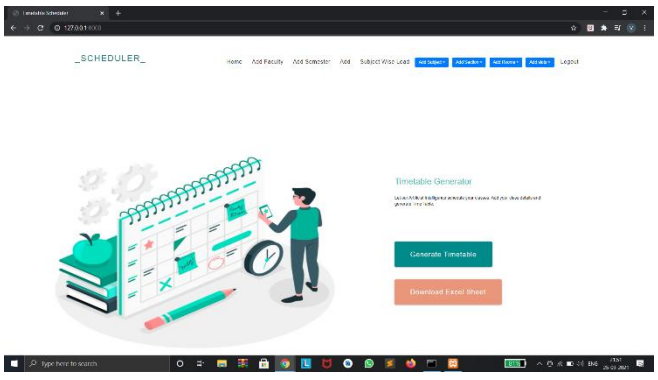



Fig 4: Home Page

Later, for getting the load sheet on excel having details of the faculties and their allotted subjects and the semester we created DataFrames which is little part of Data Science and revoked or removed the relentless data through numpy and pandas libraries through Joins in dataframe function merged the tables and created dataframes having details needed for an institution to keep a record of all the details of semesters, subjects, faculties, subject load, faculty load. On a second of click whole excel sheet is generated having all the required details.

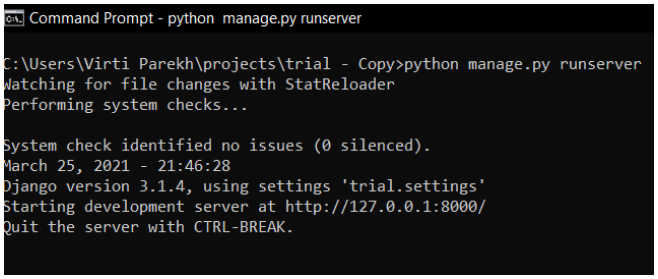



Fig 5: Executing application through command Prompt

Then proceeding with the main part applying algorithm to the data gathered we choosed it to be genetic algorithm as it works good with lot of data and is popular for such scheduling functions. Firstly, we declared the population size and with takin all the details of the lecture timings and faculties, number of classes and subjects for the specific semester this all needs to fed to the algorithm as a input also we have to declare the conditions for all the constraints which in terms is declared as fitness of the output for us as we were scheduling for three semester which had two divisions hence in total we had to check fitness for eight classes and we only choose output of the function of the fitness which had values zero or 0.1 we increase the values of fitness when the condition are not satisfied and hence lesser or lower the fitness values more is the expected output and through this an schedule of eight in a series were generated and the output was generated in both command prompt for detailed research where we can also check the fitness and test the conditions and the final output with fitness zero was selected and shown in the format of table and separate time schedule for all the three semester and eight classes in all was generated with no clashes in classrooms, faculties and faculties with their respective load and subject within specified time intervals.

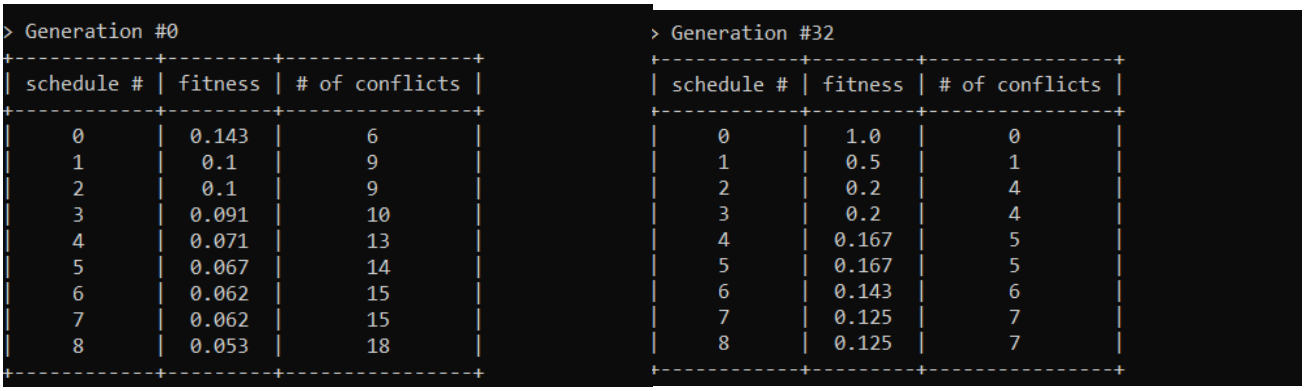



Fig 6: Generation 0 (as start) and Generation 32 (as end)
Indicates the fitness and conflicts.

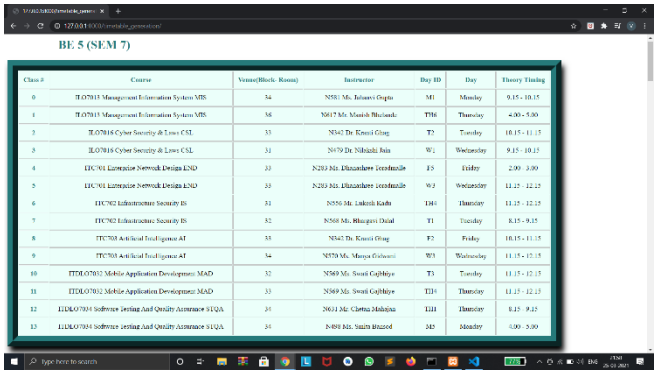



Fig 7: Timetable Schedule Semester and Division Wise

## VI. CONCLUSIONS

This paper describe the timetable scheduler problem which covers real time problems using genetic algorithm according to generations. It is tedious task to handle many faculty's database at the same time, also allocating courses, rooms, faculty according to the load is difficult. Our proposed system will help to overcome this difficulties and problems. We present an

algorithm which satisfy the hard constraints and college timetabling problem involving multiple contexts. An adaptive mutation plan has been applied for speeding up the convergency.

## VII. REFERENCES

**[1]** Adithya R Pai, Ashwitha S, Raksha Shetty, and Prof. Geethalaxmi ."Automated school schedule generator" International Journal of Scientific and Engineering Research, April 2018.

**[2]** Akshay puttaswamy , H M Arshad Ali Khan, Chandan S.V and Parka.A Department of Computer Science and Engineering, M S Ramaiah Institute of Technology, Bangalore ." An investigation on programmed plan generator". Global Journal of Innovative Research and Growth, May 2018.

**[3]** Sundresan Perumal, Mujahid Tabassum, Norita MD, Ganthan Samy, Sivananthan Perumal. "Advancement of a productive Timetable System". Global Conference on control framework, figuring and Engineering, 23-25 November 2018, Malaysia.

**[4]** Ping Guo, Jing-Xin Chen, Lin Zhu. "The Design and Implementation of schedule framework dependent on hereditary calculations". Worldwide Conference on Mechatronic Science, Electric Engineering and Computer August 19-22, 2011, Jilin.

**[5]** Saritha M, Pranav Kiran Vaze, Pradeep, Mahesh N R . Programmed plan generator. Worldwide Journal of Advanced Research in Computer Science and Software Engineering, May 2017.

**[6]** D. G. Maere, (2010). "How Working Group Automated Timetabling was established'", recovered from http://www.asap.ac.nott.ac.uk/, 2010, Last got to date9th December 2011.