

**Course Code:** INT 404

**Submitted To:** Nandini

|  |  |  |
| --- | --- | --- |
| Name | Reg ID | Roll no |
| Vummiti Muni Maanasa Murali | 11803861 | 17 |
| Srijan Dubey | 11805682 | 18 |
| Vivek Reddy | 11805765 | 19 |
| Ravi Shankar | 11805828 | 20 |

**Introduction**

The manual timetable scheduling demands considerable time and efforts along with lots of paperwork. The timetable scheduling can also be considered as a Constraint satisfaction problem (CSP), which is a unique concept in Artificial Intelligence, in which we find a solution that satisfies the given set of constraints. Time table scheduling has been in human requirements since they thought of managing time effectively It is widely used in schools, colleges and other fields of teaching. In early days, time table scheduling was done manually with a single person or some group involved in task of scheduling it with their hands, which take lot of effort and time. While scheduling, even the smallest constraints can take a lot of time and the case is even worse when the number of constraints or the amount of data to deal with increases. Other cases that can cause problem is when the number of Faculties (Teaching Staff) are less, resulting in rescheduling of time table or they need to fill on empty seats urgently. In such cases automated timetable, scheduling can be a very convenient method for managing it in computers with algorithms also proving to be eco-friendly for no paperwork.

**Proposed System**

The final system should able to generate time tables in completely automated way which will save a lot of time and effort of an institute administration. To make a timetable system generic so that I can work equally well for different School, Colleges and Universities. User defined constraints handling. Ease of use for user of system so that he/she can make automatic time table. Focus on optimization of resources i.e. teachers, labs and rooms etc. Provide a facility for everyone to view timetable. Generate multiple useful views from time table. Outcomes depends on

A. Interface for input

The system will be having an easy to use and interactive interface to enter all the inputs like the teacher name, the data for the rooms and data for the labs and the data for subject.

B. Database Capabilities

The system will have well designed database to store all the information which will be entered in as the input. Separate database maintaining basic information, subjects, teachers, batches and their associations and other details Database for holding generated timetable and for storing required timetables.

C. Processing Capabilities

The system will have algorithms to process all the data present in the database and keeping in view the various constraints like that a teacher should not have two consecutive lectures/labs, students have minimum one hour gaps,

**Project Statement**

The difficulty faced during timetabling can be represented as a constraint satisfaction problem with loose parameters and many constraints. These constraints can be replicated in a format which can be managed by the scheduling algorithm in an organized manner.

**Solution**

Timetable generation included a tedious process of assigning each subject to staff manually and scheduling the Timetable as in a way so that no clashes occur. But this process also took great use of time and also us of paper- work which is cost-ineffective. For this approach we decide a solution of using our computing skills and technology to generate the Timetable. It is done using the Automated Timetable generator which contains the involvement of Evolutionary Algorithms (EAs) called Genetic Algorithm. The Genetic Algorithm involves the process of Chromosome Representation to generate the Timetable. The above Solution gives a block model of following processes: The user will enter each of the data as counts of subjects, class-rooms, labs, lectures, students. The admin will assign each subject to their respective staff and assign them classrooms and the students whom they will teach. The Admin will use constraints as given in the algorithm so that no constraints occur. After assigning the Admin will do a verification check so that no anomalies are missed out. If the Admin encounters any mistake or clash that had been gone unnoticed earlier, he/she has the option to edit and then regenerate. After successful reviews the Timetable is uploaded on the college website for the staffs and students to view.

**Methodology**

The methodology used in this application enables us to approach Evolutionary Algorithms. Evolutionary Algorithms are a class of direct, probabilistic search and optimisation algorithms gleaned from the model of organic evolution. A Genetic Algorithm (GA) is a type of EA and is regarded as being the most widely known EA in recent times. A GA differs from other search techniques in the following ways:

GAs optimises the trade-off between exploring new points in the search space and exploring the information discovered thus far. GAs is randomised algorithms, in that they use operators whose results are governed by probability. The results for such operations are based on the value of a random number. This means GAs use probabilistic transition rules, not deterministic rules.

GA was combined with a heuristic specific greedy algorithm to take advantage of the global search of feasible solutions and specific technique efficiency in local solution optimization. This approach resulted in considerably smaller execution times.

**Summary**

This paper addresses the Timetabling Problem, which covers a very broad range of real problems faced continually in educational institutions, and we describe how Evolutionary Algorithms (EAs) can be employed to Effectively address arbitrary instances of automated timetabling problem. Automated Timetable Generator application will simplify the process of time table generation smoothly which may otherwise needed to done using spread sheet manually possibly leading to constraints problem that are difficult to determine when time table is generated manually. It helps to provide an optimal solution.