On-line help desk for college departmental activities

Asha V.G

Department of Information Science,

Dayananda Sagar college of Engineering,
Bangalore

Dr. K. N. Rama Mohan Babu

Department of Information Science,

Dayananda Sagar college of Engineering,
Bangalore

Abstract- **Many college departmental activities are very arduous and time consuming task. To create timetable, it takes lots of patience and man hours. This paper proposes a general solution to resolve these issues. This proposal includes automation of activities like creation of class time table, department library management, student information management and internals timetable generation.**

Keywords: time tabling, scheduling, on-line help desk, student information management, library management

I. INTRODUCTION

On-line help desk for college departmental activities refers to the varied computer machinery and software used to digitally create, collect, store, manipulate, and transfer the information needed for accomplishing basic tasks in a college department. This includes storage of information, transfer of the digital data and management of data. These are the basic activities of an on-line help desk system. On-line help desk helps in optimizing or automating existing department procedures. Advantages of this are, it can get tasks accomplished much quicker, it eliminates the need for a large man power, less storage space is required to store data and multiple people can access and update data simultaneously.

The On-line help desk is a combination of multiple sub modules which are required for college department automation activities. This includes:

o Automatic Timetable generation: It is used for generating the time table for the course, which also includes adding, deleting and updating a course, faculty, classroom and lab to generate the timetable.

o Automatic Internals timetable generation: It used for generating the time table for the internal examination.

o Student profile management: This module includes adding, deleting and updating student's personal information, previous education information, academic information and other non-academic information.

o Library management: This is used for barrow and return of books from department library by students and faculty.

II. LITERATURE SURVEY

[1] Genetic algorithms was invented by John Holland and has described this idea in his book "Adaptation in natural and artificial systems" in the year 1975. Genetic Algorithms are inspired by Darwin's evolutionary theory. Genetic algorithm comes under the class of evolutionary algorithms that use the principle of natural selection to derive a set of solutions towards the optimal solution. [2] It is a search which generates solutions to optimization problems using techniques inspired by natural evolution like mutation, inheritance, crossover and selection. There will be a fitness function which calculates the number of constraints violated in the timetable. [3] A timetable will have highest fitness value if it violates less number of constraints. The process can be stopped when a timetable satisfying all the hard constraints (must satisfy) is found. Advantages of this are that, it provides diverse values of solutions and reaches global maxima. Mutation is used to induce diversity in the

timetable generation. Here saving best solution is always helpful. Main disadvantage is it is complex to implement.

[4] Genetic algorithm with Bacteria Foraging: In this proposed algorithm, a "virtual" bacterium represents a point in n dimensional search space where each point is a potential solution to the time-table problem, consisting of combination of various parameters. The chemotaxis of the bacteria is used to search for optimal solutions to the timetable problem.

[5] A Parallel Genetic Algorithm (PGA) is proposed with specific methods for chromosome representation and fitness evaluation, and specific recombination and mutation operators. It uses the coarse grain parallelism.

[6] Genetic Algorithms with heuristic search: This suggests selecting the best action among to generate the new generation depending on the probabilistic operator. [7] Subjects to be taught to a class are further classified being assigned to an integer value. Assignment of integer value is to ease computing in genetic algorithm based timetabling program.

[8] Heuristic optimization methods are explicitly aimed at good feasible solutions that may not be optimal where complexity of problem or limited time available does not allow exact solution. In its simplest form the scheduling task consists of randomly mapping class, teacher and room combinations onto time slots. Here initially a random pattern is being generated for a temporary timetable. Then, the availability of teacher and class is checked for each of the slot. If the teacher is available for allocated slot to teach the subject then it will be entered to output data structure as final. It also checks for the maximum lectures per teacher to avoid the conflict. Advantage of this method is it is simple to implement. Disadvantage is that this might not result in optimum result.

[9] Fred Glover proposed the Tabu Search in 1986. Tabu search can be directly applied to any kind of optimization problem. Tabu search technique is one of the popular local search method based on neighbourhood search algorithm. Tabu search basically avoids getting trapped at local maxima.

Tabu search method is that it prevents cycling back to the previously visited solutions using memories thus creating more chances of improvement. Tabu search uses a search space which is the space of all the possible solutions. In Tabu search one of the basic elements is the "Tabu". Tabus are used to prevent cycling while moving away from local optima and through non-improving moves. Tabus are also used to move away from the previously visited portions of the search space and thus help in exploring other regions. Tabus are stored in a short-term memory called the Tabu list. Usually the Tabu list is implemented as a circular list of a predefined length. The moves are considered as Tabu only for a certain number of iterations and this number is called the "tabu tenure".

[10] The bee's algorithm is an optimization algorithm which is inspired by the food foraging behaviour of honey bees and performs a kind of neighbourhood search combined with random search to enable it to locate the global optimum. This algorithm provides a better optimal result than the tabu search. The algorithm is controlled by several parameters.

o Those are the number of scout bees (n)
o number of sites selected for neighbourhood search (out of n visited sites) (m)
o number of elite bees among m selected sites (e)
o number of bees recruited for patches visited by e elite bees (nep)
o number of bees recruited for the other (m-e) selected patches (nsp), size of patches (ngh).

[11] Kennedy and Eberhart developed Particle swarm optimization algorithm in 1995. It is a technique based on particles, each particle has its own velocity in search space to find feasible solution and adjust its position according its own previous experience and neighbours' experiences. The initial position and velocity of each particle is automatically determined. When each particle to moves to a new position then it will remember its current personal best (Pbest) and own particle information. Each particle will exchange its velocity information to other particles and it also remembers its global best (Gbest). Each particle has fitness value. Fitness value of each particle's current position is compared with Pbest and Gbest. If it is better than Pbest and Gbest then update the Pbest and Gbest.

[12] Usage integer linear programming is another method for timetable scheduling. This has become popular due to improvements in computer hardware and heuristic branch

choice. This method clearly identifies the hard constrains and soft constraints with different set of variables. The disadvantage of this method are high processor utilization and equation formation.

III. PROPOSED APPROACH

The On-line help desk for college departmental activities is a combination of multiple sub modules which are required for automation of college departmental activities. The sub modules are as listed below:

- Library management
- Student profile management
- Automatic timetable generation
- Automatic internals timetable generation

A. Library management:

This module contains the implementation of library management. It includes features like adding a book to a library, updating the book information, searching for a book, deleting a book information, issuing of a book and return of the book. The elements of the book table are title, author, category, publication and number of copies.

Algorithm:

Procedure Issue_books()

    Books=Get_book_details()

    Validate_book_data(Books)

    IF book_is_present THEN

        Issue_book()

    ELSE

        print "error, book not found"

Procedure return_books()

    Books=Get_book_details()

    Validate_book_data(Books)

    IF book_is_present THEN

        return_book()

    ELSE

        print "error, book not found"

B. Student information management:

This module contains the implementation of student information management. It includes features like adding, updating and searching for student information. The student information is split into 3 tables namely student_basic_information, student_academic_information and student_non_academic_information.

Algorithm:

Procedure Add_student_info()

    Student_info=Get_student_details()

    Validate_student_data(Student_info)

    IF Student_USN_present THEN

        Update_student_info(Student_info)

    ELSE

        Add_student_info(Student_info)

Procedure Search_Student_info()

    usn_no=Get_student_usn_no()

    IF usn_is_present THEN

        display_the_student_detail()

    ELSE

        print "error, student information not found"

C. Automatic timetable generation:

This module contains the implementation of the automated timetable generation. This module allows user to add and delete a resource, here resource can be a faculty, classroom or class. This module allows user to create a new time table by specifying the name, days and intervals per day. It also allows user to assign an activity to a time table. Here the activity contains subject name, faculty who is teaching that subject, credit points for the subject, information about

whether it is an elective subject or it is been taught by another department faculty. This also has features of viewing the time table for class and faculty and print the time table.

Algorithm:

Procedure Generate_timetable()

Timetable_name= Get_timetable_name()

Delete_already_existing_timetable(Timetable_nam

e)

Actvities_list=Get_all_activity(Timetable_name)

Intialize_prof_class_list();

IF activity_is_an_elective THEN

Save_scheduled_activity()

IF faculty_from_other_department THEN

Save_scheduled_activity()

Assign_lab()

WHILE not_all_classes_scheduled DO

For each class

generate_random_pattern()

current_fitness=calculate_fitness()

IF current_fitness > best_fitness THEN

best_fitness=current_fitness

Save_scheduled_activity()

Display_the_timetable()

D. Automatic internals timetable generation:

This module is used for generating the internal examination timetable. It takes input as faculty names for invigilation and how many sessions they will be available, classroom details and course details. It generates random combinations of course and allocates a room for the same. It also generates random order for faculty for invigilation.

Algorithm:

Procedure Generate_internals_timetable()

Faculty_details=Get_Faculty_details()

Room_details=Get_Room_details()

Class_details=Get_class_details()

timetable=Generate_Random_pattern_for_classom (Room_details, Class_details)

Invigilator_timetable=Generate_random_pattern(F aculty_details)

Display_timetable(timetable)

Display_timetable(Invigilator_timetable)

IV.   RESULTS

The on-line help desk for departmental activities after implementation, results in better utilization of resources. The creation of a time table of class helps in reducing lot of manual work involved in doing the same. The timetable output is displayed in a table format with the details about the class room and faculty. Below figure shows the timetable generated for "4thsem-Asec" class, which has classes scheduled for 6 days with 7 intervals on each day. For each interval, what is the subject been taught and who is the faculty. For the laboratory sessions, it shows which batch has which lab and two faculty associated with each lab. This also has provision for signature of head of the department and the principal.

Fig: Class timetable output

It also generates faculty-wise time table which helps the faculty to schedule their work more effectively. Here each interval will have the subject been taught and the class room information. Below is the result for timetable generated for faculty named "FACULTY1".

Fig: Faculty time table output

The above-mentioned algorithm for timetable generation takes an average 1ms to create time table for 6 theory subjects and 2 lab sessions. Experimental data is tabulated as follows:

	1	2	3	4	5	6	7	8	9	10	Average
Instance 1	1.010219097	1.336635113	1.116934776	1.060538095	1.020517111	1.070580006	0.995179176	1.070633173	0.901996136	0.982973814	1.05722065
Instance 2	1.104769945	0.955060005	1.061628103	1.015060186	1.084637165	1.039307833	1.174778938	1.043198109	1.176429033	1.142654896	1.079752421
Instance 3	1.144597054	1.118840933	0.910571098	1.164182901	1.570677996	0.995975971	0.943542957	1.111727953	1.060983181	0.986532211	1.100763226
Instance 4	1.094002962	1.118336916	1.422373056	1.038676023	1.149082899	0.970551968	1.086786032	1.13923596	1.014873981	1.174124956	1.120803475
Instance 5	1.177643061	1.068789959	1.237665984	1.065061092	1.270783901	1.013537884	1.124449968	1.160953045	0.962274075	1.043345928	1.11245029
Instance 6	1.241620064	1.374849081	1.036990881	0.954895973	1.088627815	1.156619881	1.143990993	1.024448872	1.071043015	1.109274149	1.120436072
Instance 7	1.321905851	1.03408885	1.265860081	1.087218046	1.063874006	1.310595036	0.938912153	1.116734028	1.151888847	1.059487104	1.1350564
Instance 8	1.146332026	1.265554905	1.11849308	1.163560867	1.028420925	1.060390949	1.159734011	1.185916901	0.971457958	1.190351009	1.129021263
											1.109937975

Fig: Experimental data

The internals timetable generation makes sure students of different classes are set to write the internal examination in a single room which improves the authenticity of the examination. It also allows the assigning invigilators for different sessions.

The library management module will generate a report of how many books are available, for whom the book is been issued to, how many copies of the books are available etc. This helps in keeping track of books in the library.

The student information module provides the students information on-line, which makes the accessibility of information at ease. It also provides report of overall students' performance in a year which will be very helpful in evaluating the quality of education.

V. FUTURE SCOPE

In library management, inclusion of a feature of reading from ISBN pattern would be beneficial. In student information management inclusion of feature for connecting to alumni database at a college level. In time table generation providing an option of students to choose their subject of interest and attend only those classes is futuristic. In internals timetable generation inclusion of feature to automate the question paper creation would add more value.

VI. CONCLUSION

The intention of the providing the on-line help desk for college departmental activities are satisfied. This incorporates several techniques, aimed to improve the efficiency of the staff. The timetable generation algorithm addresses the important clashes between the availability of faculty. The timetable generation also considers the institutional, cluster and departmental elective to be scheduled as part of the timetable generation. It also considers the subjects which is been taught by another department faculty. The library management reduces the manual book-keeping information about the books available in the department. It makes it less cumbersome to manage library. Keeping the student information on-line, helps in ease of access also reduces lot of paper work and ease of access to the student information from anywhere. Internals time table generation makes sure students of different classes are set to write the internal examination in a room which improves the authenticity of the internal examination.

REFERENCES

[1] "Automatic Timetable Generation using Genetic Algorithm" by Dipesh Mittal, Hiral Doshi, Mohammed Sunasra, Renuka Nagpure -International Journal of Advanced Research in Computer and Communication Engineering Vol. 4, Issue 2, February 2015

[2] "Genetic Algorithm to Generate the Automatic Time-Table – An Over View" by Asif Ansari, Prof Sachin Bojewar- IEEE publication 2012

[3] "Development of a Timetabling Software Using Soft-computing Techniques with a Case Study" by M.F.M. Sabri, M.H. Husin, S.K. Chai- IEEE publication 2010

[4] "Optimal Time-Table Generation by hybridized Bacterial Foraging and Genetic Algorithms" - Om Prakash Verma, Rohan Garg and Vikram Singh Bisht - International Conference on Communication Systems and Network Technologies 2012

[5] "The Application of a Parallel Genetic Algorithm to Timetabling of Elementary School Classes" by Nadim Šrndić , Mirza Dervišević , Emir Pandžo , Samim Konjicija -IEEE publication 2009

[6] "An Algorithm to Automatically Generate Schedule for School Lectures Using a Heuristic Approach"- by Anirudha Nanda, Manisha P. Pai, and Abhijeet Gole- International Journal of Machine Learning and Computing, Vol. 2, No. 4, August 2012

[7] "A Novel Greedy Heuristic Algorithm for University Course Timetabling Problem" - by Dezhen Zhang, Saijun Guo, Weishi Zhang, Shujuan Yan –IEEE publication 2014

[8] "Efficient Design of University Timetable" by M.V. Padmini, K. Athre- IEEE publication 2010

[9] "Automating a Real-world University Timetabling Problem with Tabu Search algorithm" by Khang Nguyen, Nguyen Dang, Khon Trieu, Nuong Tran- IEEE publication 2010

[10] "The Bees Algorithm for A Practical University Timetabling Problem in Vietnam" – By Nguyen Tan Tran Minh Khang, Nguyen Ba Phuc , Tran Thi Hue Nuong – IEEE publication-2011

[11] "AUTOMATED TIMETABLE GENERATOR USING PARTICLE SWARM OPTIMIZATION" by Andeep Kumar Kawaljeet Singh, Neeraj Sharma- International Journal on Recent and Innovation Trends in Computing and Communication-2013

[12]" Modelling Constraints in School Timetabling using Integer Linear Programming" by Samir Ribi´,Razija Turˇcinhoˇ zi´ ,Amela Muratovi´c-Ribi´ - International conference on information, communication and automation technologies-2015