

## APPENDIX 1

# Intelligent Emergency-Auto-Adjustments for Faculty

## PROJECT REPORT

by

Registration No.	Name	Roll No.
11803574	PURU RAJ	A33
11803497	ABHINANDAN R	A32
11907993	P. CHANDU	B34



**L** OVELY  
**P** ROFESSIOAL  
**U** NIVERSITY

**Department of Intelligent Systems**  
**School of Computer Science Engineering**  
**Lovely Professional University, Jalandhar**

**April-2020**

## APPENDIX 2

### Student Declaration

This is to declare that this report has been written by us. No part of the report is copied from other sources. All information included from other sources have been duly acknowledged. We aver that if any part of the report is found to be copied, we are shall take full responsibility for it.

Signature of Student

**PURU RAJ**

Roll number: A 33

**ABHINANDAN R**

Roll number: A 32

**PENTAKOTA CHANDU**

Roll number: B 34

Place: LOVELY PROFESSIONAL UNIVERSITY

Date: 6<sup>th</sup> April, 2020

## APPENDIX 3

### TABLE OF CONTENTS

TITLE	PAGE NO.
1. Abstract .....	4
2. Introduction.....	5
3. Description of Project .....	6
4. Algorithm and Programming .....	7
5. Library used and Screenshots.....	9
5. References.....	11

## **ABSTRACT**

This project introduces a practical timetabling algorithm capable of taking care of both strong and weak constraints effectively, used in an automated timetable Adjustment system for faculties. So that each teacher can view their timetable once they are finalized for a given semester but they can't edit them. Auto Adjust Timetable Generation System generates timetable for each class and teacher, in keeping with the availability calendar of teachers, availability and capacity of physical resources (such as classrooms, laboratories and computer room) and rules applicable at different classes, semesters, teachers and subjects level.

This paper presents a practical method for modeling and solving a dynamic resource allocation of automatic scheduling problem. Poor scheduling practices would cause double-assignments of lecturers, prolonged postponement and cancellations of lectures as well as inefficient use of time and resources. This method will follow a pre-assigned logic rules and algorithm to fit the optimization criteria's. The output of this research will be an automatic set of lecture schedule alternatives that will take into account all the constraints.

The proposed system performs satisfactorily in term of accuracy, data handling and adaptability on helping the faculty to arrange lectures more easily, yield a reliable record and increase efficient use of resources.

## INTRODUCTION

In a general educational timetabling problem, a set of events (e.g. courses and exams, etc) are assigned into a certain number of timeslots (time periods) subject to a set of constraints, which often makes the problem very difficult to solve in real-world circumstances. There are large numbers of commercial software packages available to assist with timetabling problem. However, due to the variety of characteristic, procedure and complexity of each university in the adjustment process make the study in this research area still interesting. In order to solve the best solution for this timetable adjustment problem, a widely research area has been studied in various complexity problems with different algorithm solution. This research will focus on how to build automated Adjustment that could also handle dynamic resource allocation problem.

Even though most college administrative work has been computerized, the lecture timetable adjustment is still mostly done manually due to its inherent difficulties. The manual lecture timetable adjustment demands considerable time and efforts. The lecture-timetable adjustment is a Constraint satisfaction problem in which we find a solution that satisfies the given set of constraints. An automated timetable adjustment is a temporal arrangement of a set of lectures and classrooms in which all given constraints are satisfied. Creating such timetables manually is complex and time-consuming process. By automating this process with computer assisted auto adjust timetable generator can save a lot of precious time of administrators who are involved in creating and managing course timetables.

## DESCRIPTION OF PROJECT

This project is based on the lecture make-up adjustments rules of faculties of LPU

In this teacher's attendance is taken on command line interface as value 0 (represents absence) and 1 (represents presence).

If a teacher is absent then further it is checked that the particular teacher has lecture on that particular day and then schedule of that teacher is checked and based on that time table a teacher is assigned who is free which is again checked by comparing time-tables of these two teachers.

Then the same is repeated for all weekday.

All above modules are repeated for each teacher.

## ALGORITHM AND PROGRAMMING

```
import datetime
import calendar

def tt():

    teacher1=([1,1,0,0,0,0,1,1],
               [0,1,0,1,1,0,1,1],
               [1,1,0,0,1,0,1,1],
               [1,1,0,0,1,0,1,1],
               [1,1,0,0,0,0,1,1],
               [0,0,0,0,0,0,0,0])
    teacher2=[]
    teacher2=([0,0,1,1,1,0,1,1],
               [0,0,0,1,1,0,0,1],
               [0,0,1,1,0,0,1,1],
               [0,0,1,1,0,1,0,0],
               [0,1,1,0,1,0,1,0],
               [0,0,0,0,0,0,0,0])
    teacher3=[]
    teacher3=([1,1,1,0,0,0,1,1],
               [1,0,0,0,0,0,1,1],
               [0,0,0,0,1,1,0,1],
               [1,1,0,0,1,1,1,0],
               [0,0,1,1,1,0,1,1],
               [0,0,0,0,0,0,0,0])
    teacher4=[]
    teacher4=([0,0,0,1,1,0,0,0],
               [0,1,1,0,0,0,0,0],
               [0,0,1,1,0,1,0,0],
               [0,0,1,0,1,0,1,0],
               [1,1,0,1,0,1,0,0],
               [0,0,0,0,0,0,0,0])

    days=5
    hours=8
    import datetime
    import calendar
    now = datetime.datetime.now().weekday()
    now=3
    # now+=1
    print(now)
    print("if teacher 1 is absent press: 1\nif teacher 2 is absent press: 2\nif teacher 3 is absent press : 3\nif teacher 4 is
absent press : 4")
    ta=int(input())
    if ta==1: #if teacher 1 is absent
        if now==1: #check if day is Monday

            if teacher1[0][0]==1:#check if teacher 1 has class at 9am
                if teacher2[0][0]== 0:
                    print("Teacher 2 will be substiute the lecture from 9-10 am")
```

```

elif teacher3[0][0]==0:
    print("Teacher 3 will be substitute the lecture from 9-10 am")

elif teacher4[0][0]==0:
    print("Teacher 4 will be substitute the lecture from 9-10 am")

# elif teacher5[0][0]==0:
#     print("Teacher 5 will be substitute the lecture from 9-10 am")

else:
    print("No teacher is Free")

```

---

```

if teacher2[4][0]==1:#check if teacher 1 has class at 9am
    if teacher1[4][0]== 0:
        print("Teacher 1 will be substitute the lecture from 9-10 am")

    elif teacher3[4][0]==0:
        print("Teacher 3 will be substitute the lecture from 9-10 am")

    elif teacher4[4][0]==0:
        print("Teacher 4 will be substitute the lecture from 9-10 am")

    # elif teacher5[4][0]==0:
    #     print("Teacher 5 will be substitute the lecture from 9-10 am")

    else:
        print("No teacher is Free")

```

---

```

if teacher4[4][7]==1:#check if teacher 1 has class at 4pm
    if teacher1[4][7]== 0:
        print("Teacher 1 will be substitute the lecture from 4-5 pm")

    elif teacher2[4][7]==0:
        print("Teacher 2 will be substitute the lecture from 4-5 pm")
    elif teacher3[4][7]==0:
        print("Teacher 3 will be substitute the lecture from 4-5 pm")
    # elif teacher5[4][7]==0:
    #     print("Teacher 5 will be substitute the lecture from 4-5 pm")
    else:
        print("No teacher is Free")
elif now==6:
    print("Not a lecture day")

```

```

r=int(input("Press 1 to run program\nPress 0 to exit program\n"))
while r==1:
    tt()
    r=int(input("Press 1 to run program\nPress 0 to exit program\n"))

```

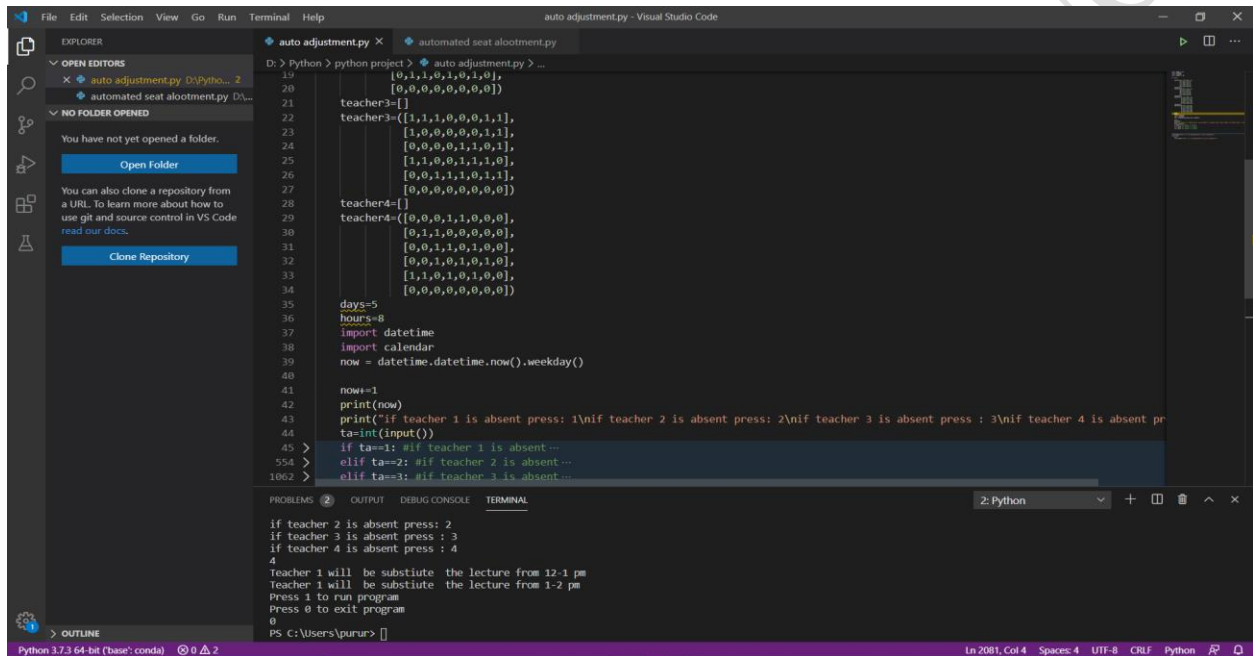


## LIBRARY USED

**Datetime Module:** - This module was imported to get the current date and time.

**Calendar Module:-** This module was imported to calculate the weekday by combining with datetime module.

## Screenshots



```
19 [0,1,1,0,1,0,1,0],
20 [0,0,0,0,0,0,0,0]]
21 teacher3=[]
22 teacher3=[[1,1,1,0,0,0,1,1],
23           [1,0,0,0,0,0,1,1],
24           [0,0,0,0,1,1,0,1],
25           [1,1,0,0,1,1,1,0],
26           [0,0,1,1,0,1,1,1],
27           [0,0,0,0,0,0,0,0]]
28 teacher4=[]
29 teacher4=[[0,0,0,1,1,0,0,0],
30           [0,1,1,0,0,0,0,0],
31           [0,0,1,1,0,1,0,0],
32           [0,0,1,0,1,0,1,0],
33           [1,1,0,1,0,1,0,0],
34           [0,0,0,0,0,0,0,0]]
35
36 days=5
37 hours=8
38 import datetime
39 import calendar
40 now = datetime.datetime.now().weekday()
41
42 now+=1
43 print(now)
44 print("if teacher 1 is absent press: 1\nif teacher 2 is absent press: 2\nif teacher 3 is absent press: 3\nif teacher 4 is absent pr
45 ta=int(input())
46 if ta==1: #if teacher 1 is absent--
554 elif ta==2: #if teacher 2 is absent--
1062 elif ta==3: #if teacher 3 is absent--
```

```
if teacher 2 is absent press: 2
if teacher 3 is absent press: 3
if teacher 4 is absent press: 4
4
Teacher 1 will be substitute the lecture from 12-1 pm
Teacher 1 will be substitute the lecture from 1-2 pm
Press 1 to run program
Press 0 to exit program
0
PS C:\Users\punur>
```

Pic(1):- This picture shows the array in which data entry which is based on teachers' timetable.

```

149
150     if teacher1[1][0]==1:check if teacher 1 has class at 9am
151     if teacher2[1][0]== 0:
152         print("Teacher 2 will be substitute the lecture from 9-10 am")
153
154     elif teacher3[1][0]==0:
155         print("Teacher 3 will be substitute the lecture from 9-10 am")
156
157     elif teacher4[1][0]==0:
158         print("Teacher 4 will be substitute the lecture from 9-10 am")
159
160     # elif teacher5[1][0]==0:
161     #     print("Teacher 5 will be substitute the lecture from 9-10 am")
162
163     else:
164         print("No teacher is Free")
165
166     if teacher1[1][1]==1:check if teacher 1 has class at 10am
167     if teacher2[1][1]== 0:
168         print("Teacher 2 will be substitute the lecture from 10-11 am")
169     elif teacher3[1][1]==0:
170         print("Teacher 3 will be substitute the lecture from 10-11 am")
171     elif teacher4[1][1]==0:
172         print("Teacher 4 will be substitute the lecture from 10-11 am")
173     # elif teacher5[1][1]==0:
174     #     print("Teacher 5 will be substitute the lecture from 10-11 am")
175
176     else:
177         print("No teacher is Free")
178
179     if teacher1[1][2]==1:check if teacher 1 has class at 11am

```

Pic(2):- This picture shows the if-else logic to assign a teacher as a substitute.

```

d:\> Python> python project> automated seat allotment.py> booking
Windows PowerShell
Copyright (C) Microsoft Corporation. All rights reserved.

Try the new cross-platform PowerShell https://aka.ms/pscore6

PS C:\Users\punur> conda activate base
PS C:\Users\punur> & C:\Users\punur\Anaconda3\python.exe "d:/Python/python project/auto adjustment.py"
Press 1 to run program
Press 0 to exit program
1
1
if teacher 1 is absent press: 1
if teacher 2 is absent press: 2
if teacher 3 is absent press : 3
if teacher 4 is absent press : 4
3
Teacher 2 will be substitute the lecture from 9-10 am
Teacher 2 will be substitute the lecture from 10-11 am
Teacher 1 will be substitute the lecture from 11-12 pm
Teacher 4 will be substitute the lecture from 3-4 pm
Teacher 4 will be substitute the lecture from 4-5 pm
Press 1 to run program
Press 0 to exit program
1
1
if teacher 1 is absent press: 1
if teacher 2 is absent press: 2
if teacher 3 is absent press : 3
if teacher 4 is absent press : 4
3
Teacher 2 will be substitute the lecture from 9-10 am
Teacher 2 will be substitute the lecture from 10-11 am
Teacher 1 will be substitute the lecture from 11-12 pm
Teacher 4 will be substitute the lecture from 3-4 pm
Teacher 4 will be substitute the lecture from 4-5 pm
Press 1 to run program
Press 0 to exit program

```

Pic(3):- This Picture shows the output of the code.

## REFERENCES

- A Timetable Prediction For Technical Educational System Using Genetic Algorithm
- Bhaduri a “university timetable scheduling using genetic algorithm”
- Silberschatz, H. Korth and S. Sudarshan, “Database System Concepts”, 4th Edition, McGraw-Hill International, 2002.
- Researchgate.net

ARTIFICIAL INTELLIGENCE