A black sign with white text

Description automatically generated

FINAL REPORT

**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 |

**Abstract**

Time table generation is a monotonous job when done manually for any administration of any educational institutions. Providing an automatic time table generator would help make it is easier and less time consuming. The system proposed by our project helps to generate the time table automatically with accuracy hence saving time. It avoids the complexity of making the time table manually. In our project we used algorithms like genetic heuristic, resource scheduling to reduce these difficulties of generating timetable. These algorithms incorporate a numeral of strategy, aimed to improve the operativeness of the search operation. The system will take various inputs like number of subjects, teachers, workload of a teacher, semester, priority of subject. By relying on these inputs, it will generate possible time tables for working days of the week for teaching faculty. This will integrate by making optimal use of all resources in a way that will best suit the constraints.

**Introduction**

Even though majority of the college organisation work has been transformed, the lecture timetable preparation is still manually done due to the underlying difficulties. The manual lecture-timetable scheduling is a limitation fulfilment problem in which we find a result that satisfies the given set of constraints.

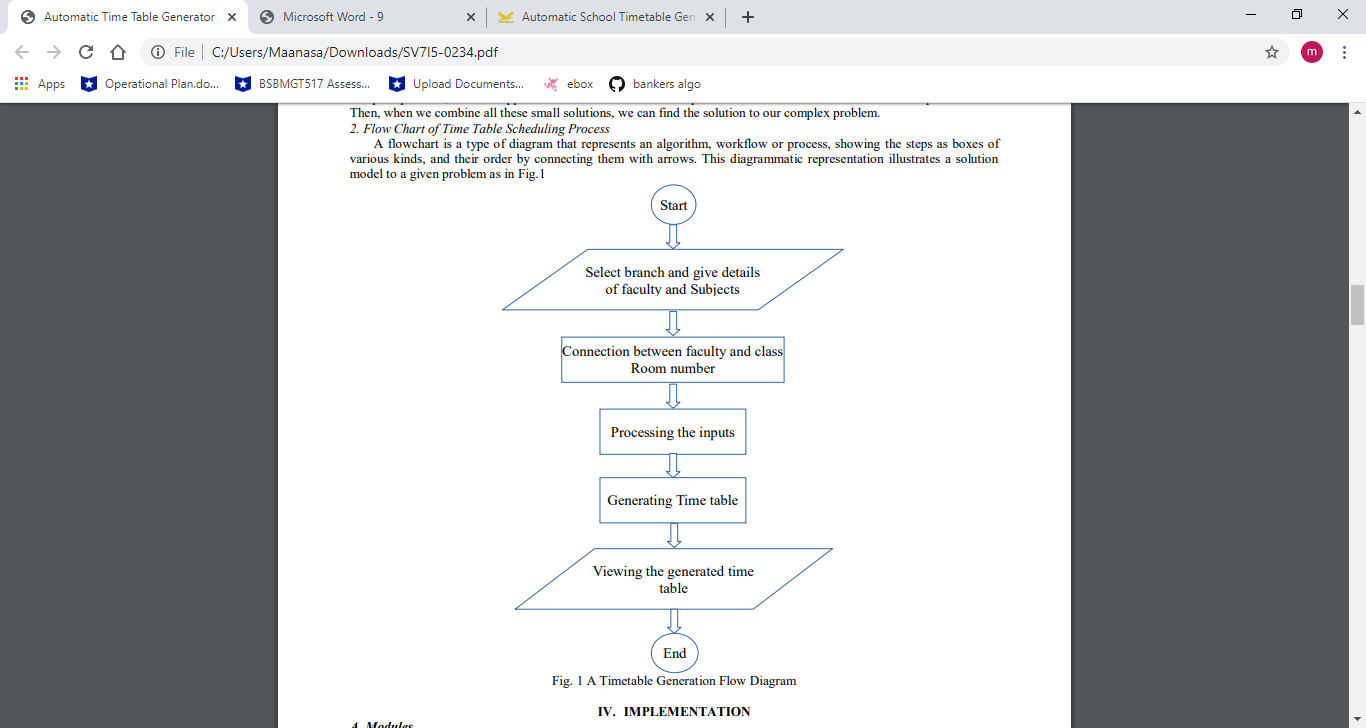
The process of preparing a timetable involves beneficial employment of resources which needs to be confronted each year by every educational institute. Most colleges have a number of different courses and each course has a number of subjects. Now there are limited faculties, each faculty teaching more than one subjects. So now the time table needed to schedule the faculty at provided time slots in such a way that their timings do not overlap and the time table schedule makes best use of all faculty subject demand. 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.

**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.



CODE

from tkinter import \*  
import os  
  
# Designing window for registration of new Faculty  
  
def register():  
    global register\_screen  
    register\_screen = Toplevel(main\_screen)  
    register\_screen.title("Register")  
    register\_screen.geometry("300x250")  
  
    global username  
    global password  
    global username\_entry  
    global password\_entry  
    username = StringVar()  
    password = StringVar()  
  
    Label(register\_screen, text="Staff Login", bg="orange").pack()  
    Label(register\_screen, text="").pack()  
    username\_lable = Label(register\_screen, text="Username \* ")  
    username\_lable.pack()  
    username\_entry = Entry(register\_screen, textvariable=username)  
    username\_entry.pack()  
    password\_lable = Label(register\_screen, text="Password \* ")  
    password\_lable.pack()  
    password\_entry = Entry(register\_screen, textvariable=password, show='\*')  
    password\_entry.pack()  
    Label(register\_screen, text="").pack()  
    Button(register\_screen, text="Register", width=10, height=1, bg="yellow", command=register\_user).pack()  
  
  
# Designing window for  login  
  
def login():  
    global login\_screen  
    login\_screen = Toplevel(main\_screen)  
    login\_screen.title(" Staff Login")  
    login\_screen.geometry("300x250")  
    Label(login\_screen, text="Please enter your Details ").pack()  
    Label(login\_screen, text="").pack()  
  
    global username\_verify  
    global password\_verify  
  
    username\_verify = StringVar()  
    password\_verify = StringVar()  
  
    global username\_login\_entry  
    global password\_login\_entry

    Label(login\_screen, text="Username \* ").pack()  
    username\_login\_entry = Entry(login\_screen, textvariable=username\_verify)  
    username\_login\_entry.pack()  
    Label(login\_screen, text="").pack()

    Label(login\_screen, text="Password \* ").pack()  
    password\_login\_entry = Entry(login\_screen, textvariable=password\_verify, show='\*')  
    password\_login\_entry.pack()  
    Label(login\_screen, text="").pack()  
    Button(login\_screen, text="Login", width=10, height=1, command=login\_verify).pack()  
  
  
# Implementing event on register button  
  
def register\_user():  
    username\_info = username.get()  
    password\_info = password.get()  
  
    file = open(username\_info, "w")  
    file.write(username\_info + "\n")  
    file.write(password\_info)  
    file.close()  
  
    username\_entry.delete(0, END)  
    password\_entry.delete(0, END)  
  
    Label(register\_screen, text="Registration Success", fg="green", font=("calibri", 11)).pack()  
  
  
# Implementing event on login button  
  
def login\_verify():  
    username1 = username\_verify.get()  
    password1 = password\_verify.get()  
    username\_login\_entry.delete(0, END)  
    password\_login\_entry.delete(0, END)  
  
    list\_of\_files = os.listdir()  
    if username1 in list\_of\_files:  
        file1 = open(username1, "r")  
        verify = file1.read().splitlines()  
        if password1 in verify:  
            login\_sucess()  
  
        else:  
            password\_not\_recognised()  
  
    else:  
        user\_not\_found()  
  
  
# Designing popup for login success  
  
def login\_sucess():  
    global login\_success\_screen  
    global delete\_login\_success\_screen  
     
    login\_success\_screen = Toplevel(login\_screen)  
    login\_success\_screen.title("Welcome to Faculty page")  
    login\_success\_screen.geometry("300x250")  
    Label(login\_success\_screen,text="Enter the  Subject and Department",bg="orange",width="300",height="2",font=("Calibri",13)).pack()  
    Label(login\_success\_screen,text="").pack()  
    Button(login\_success\_screen,text="Department", height="2", width="30", command=generate\_time\_table).pack()  
    Label(login\_success\_screen,text="").pack()  
    Button(login\_success\_screen,text="Subjects", height="2", width="30", command=generate\_time\_table).pack()

# Designing popup for login invalid password  
  
def password\_not\_recognised():  
    global password\_not\_recog\_screen  
    password\_not\_recog\_screen = Toplevel(login\_screen)  
    password\_not\_recog\_screen.title("Success")  
    password\_not\_recog\_screen.geometry("150x100")  
    Label(password\_not\_recog\_screen, text="Invalid Password ").pack()  
    Button(password\_not\_recog\_screen, text="OK", command=delete\_password\_not\_recognised).pack()  
  
  
# Designing popup for user not found  
  
def user\_not\_found():  
    global user\_not\_found\_screen  
    user\_not\_found\_screen = Toplevel(login\_screen)  
    user\_not\_found\_screen.title("Success")  
    user\_not\_found\_screen.geometry("150x100")  
    Label(user\_not\_found\_screen, text="User Not Found").pack()  
    Button(user\_not\_found\_screen, text="OK", command=delete\_user\_not\_found\_screen).pack()  
     
def generate\_time\_table():  
    global generate\_time\_table  
    global delete\_generate\_time\_table  
    generate\_time\_table=Toplevel(login\_screen)  
    generate\_time\_table.title("Time table Generator")  
    generate\_time\_table.geometry("800x600")  
    Label(generate\_time\_table, text="Enter the details below as per the slots given",bg="blue", width="300", height="2", font=("Calibri", 13)).pack()  
    global sub1  
    global sub1\_entry  
    global sub2  
    global sub2\_entry  
    global sub3  
    global sub3\_entry  
    global sub4  
    global sub4\_entry  
    global sub5  
    global sub5\_entry  
    global sub6  
    global sub6\_entry  
    global sub7  
    global sub7\_entry  
    global sub8  
    global sub8\_entry  
    global sub9  
    global sub9\_entry  
    global sub10  
    global sub10\_entry  
    global sub11  
    global sub11\_entry  
    global sub12  
    global sub12\_entry  
    """global sub13  
    global sub13\_entry  
    global sub14  
    global sub14\_entry  
    global sub15  
    global sub15\_entry  
    global sub16  
    global sub16\_entry  
    global sub17

    global sub17\_entry  
    global sub18  
    global sub18\_entry  
    global sub19  
    global sub19\_entry  
    global sub20  
    global sub20\_entry  
    """  
    sub1 = StringVar()  
    sub2 = StringVar()  
    sub3 = StringVar()  
    sub4 = StringVar()      
    sub5 = StringVar()  
    sub6 = StringVar()  
    sub7 = StringVar()  
    sub8 = StringVar()  
    sub9 = StringVar()  
    sub10 = StringVar()  
    sub11 = StringVar()  
    sub12= StringVar()  
    """sub13 = StringVar()  
    sub15= StringVar()  
    sub16 = StringVar()  
    sub17 = StringVar()  
    sub18= StringVar()  
    sub19 = StringVar()  
    sub20= StringVar()  
    """  
    #Monday Time table  
    Label(generate\_time\_table, text="Monday",bg="red", fg="white").pack()  
    Label(generate\_time\_table,text="Enter Subject 1").pack()  
    sub1= Entry(generate\_time\_table, textvariable=sub1)  
    sub1.pack()  
    Label(generate\_time\_table,text="Enter Subject 2").pack()  
    sub2= Entry(generate\_time\_table, textvariable=sub2)  
    sub2.pack()  
    sub2\_entry = Entry(generate\_time\_table, textvariable=sub2)  
    sub2.pack()  
    Label(generate\_time\_table,text="Enter Subject 3").pack()  
    sub3= Entry(generate\_time\_table, textvariable=sub3)  
    sub3.pack()  
    sub3\_entry = Entry(generate\_time\_table, textvariable=sub3)  
    sub3.pack()  
    Label(generate\_time\_table,text="Enter Subject 4").pack()  
    sub4= Entry(generate\_time\_table, textvariable=sub4)  
    sub4.pack()  
    sub4\_entry = Entry(generate\_time\_table, textvariable=sub4)  
    sub4.pack()  
     
    #Tuesday  
    Label(generate\_time\_table, text="").pack()  
    Label(generate\_time\_table, text="Tuesday",bg="red", fg="white").pack()  
    Label(generate\_time\_table,text="Enter Subject 1").pack()  
    sub5= Entry(generate\_time\_table, textvariable=sub5)  
    sub5.pack()  
    sub5\_entry = Entry(generate\_time\_table, textvariable=sub5)  
    sub5.pack()  
    Label(generate\_time\_table,text="Enter Subject 2").pack()  
    sub6 = Entry(generate\_time\_table, textvariable=sub6)  
    sub6.pack()  
    sub6\_entry = Entry(generate\_time\_table, textvariable=sub6)  
    sub6.pack()

    Label(generate\_time\_table,text="Enter Subject 3").pack()  
    sub7= Entry(generate\_time\_table, textvariable=sub7)  
    sub7.pack()

    sub7\_entry = Entry(generate\_time\_table, textvariable=sub7)  
    sub7.pack()  
    Label(generate\_time\_table,text="Enter Subject 4").pack()  
    sub8= Entry(generate\_time\_table, textvariable=sub8)  
    sub8.pack()  
    sub8\_entry = Entry(generate\_time\_table, textvariable=sub8)  
    sub8.pack()  
     
    #Wednesday  
    Label(generate\_time\_table, text="").pack()  
    Label(generate\_time\_table, text="Wednesday",bg="red", fg="white").pack()  
    Label(generate\_time\_table,text="Enter Subject 1").pack()  
    sub9= Entry(generate\_time\_table, textvariable=sub9)  
    sub9.pack()  
    sub9\_entry = Entry(generate\_time\_table, textvariable=sub9)  
    sub9.pack()  
    Label(generate\_time\_table,text="Enter Subject 2").pack()  
    sub10 = Entry(generate\_time\_table, textvariable=sub10)  
    sub10.pack()  
    sub10\_entry = Entry(generate\_time\_table, textvariable=sub10)  
    sub10.pack()  
    Label(generate\_time\_table,text="Enter Subject 3").pack()  
    sub11= Entry(generate\_time\_table, textvariable=sub11)  
    sub11.pack()  
    sub11\_entry = Entry(generate\_time\_table, textvariable=sub11)  
    sub11.pack()  
    Label(generate\_time\_table,text="Enter Subject 4").pack()  
    sub12= Entry(generate\_time\_table, textvariable=sub12)  
    sub12.pack()  
    sub12\_entry = Entry(generate\_time\_table, textvariable=sub12)  
    sub12.pack()  
    """#Thursday  
    Label(generate\_time\_table, text="").pack()  
    Label(generate\_time\_table, text="Thursday",bg="red", fg="white").pack()  
    Label(generate\_time\_table,text="Enter Subject 1").pack()  
   sub13= Entry(generate\_time\_table, textvariable=sub13)  
    sub13.pack()  
    sub13\_entry = Entry(generate\_time\_table, textvariable=sub13)  
    sub13.pack()  
    Label(generate\_time\_table,text="Enter Subject 2").pack()  
    sub14 = Entry(generate\_time\_table, textvariable=sub10)  
    sub14.pack()  
    sub14\_entry = Entry(generate\_time\_table, textvariable=sub14)  
    sub14.pack()  
    Label(generate\_time\_table,text="Enter Subject 3").pack()  
    sub15= Entry(generate\_time\_table, textvariable=sub15)  
    sub15.pack()  
    sub15\_entry = Entry(generate\_time\_table, textvariable=sub15)

    sub15.pack()  
    Label(generate\_time\_table,text="Enter Subject 4").pack()  
    sub16= Entry(generate\_time\_table, textvariable=sub16)  
    sub16.pack()  
    sub16\_entry = Entry(generate\_time\_table, textvariable=sub16)  
    sub16.pack()      
    #Friday  
    Label(generate\_time\_table, text="",bg="red", fg="white").pack()

    Label(generate\_time\_table, text="Friday").pack()  
    Label(generate\_time\_table,text="Enter Subject 1").pack()  
    sub17= Entry(generate\_time\_table, textvariable=sub17)  
    sub17.pack()  
    sub17\_entry = Entry(generate\_time\_table, textvariable=sub17)  
    sub17.pack()  
    Label(generate\_time\_table,text="Enter Subject 2").pack()  
    sub18 = Entry(generate\_time\_table, textvariable=sub18)  
    sub18.pack()  
    sub18\_entry = Entry(generate\_time\_table, textvariable=sub18)  
    sub18.pack()  
    Label(generate\_time\_table,text="Enter Subject 3").pack()  
    sub19= Entry(generate\_time\_table, textvariable=sub19)  
    sub19.pack()  
    sub19\_entry = Entry(generate\_time\_table, textvariable=sub19)  
    sub19.pack()  
    Label(generate\_time\_table,text="Enter Subject 4").pack()  
    sub20= Entry(generate\_time\_table, textvariable=sub20)  
    sub20.pack()  
    sub20\_entry = Entry(generate\_time\_table, textvariable=sub20)  
    sub20.pack()  
    """  
    Label(generate\_time\_table,text="").pack()  
    Label(generate\_time\_table, text="CSE").pack()  
    Button(generate\_time\_table, text="Submit",command=time\_table\_input).pack()  
     
  
  
def time\_table\_input():  
    if os.path.exists("Time\_table"):  
        os.remove("Time\_table")  
    else:  
        print("The file does not exist")  
    sub1\_info=sub1.get()  
    sub2\_info=sub2.get()  
    sub3\_info=sub3.get()  
    sub4\_info=sub4.get()  
    sub5\_info=sub5.get()  
    sub6\_info=sub6.get()  
    sub7\_info=sub7.get()  
    sub8\_info=sub8.get()  
    sub9\_info=sub9.get()  
    sub10\_info=sub10.get()  
    sub11\_info=sub11.get()  
    sub12\_info=sub12.get()  
    """sub13\_info=sub13.get()  
    sub14\_info=sub14.get()  
    sub15\_info=sub15.get()"""  
    file=open("Time\_table",'at')  
    file.write("Monday")  
    file.write("\n")

    file.write(sub1\_info + " ")  
    file.write(sub2\_info + " ")  
    file.write(sub3\_info + " ")  
    file.write(sub4\_info + " ")  
    file.write("\nTuesday")  
    file.write("\n")  
    file.write(sub5\_info + " ")  
    file.write(sub6\_info + " ")  
    file.write(sub7\_info + " ")  
    file.write(sub8\_info + " ")  
    file.write("\nWednesday")

    file.write("\n")  
    file.write(sub9\_info + " ")  
    file.write(sub10\_info + " ")  
    file.write(sub11\_info + " ")  
    file.write(sub12\_info + " ")  
    file.close()  
    file = open("Time\_table", 'r')  
    for x in file:  
        print(x)  
    file.close()  
  
  
# Deleting popups  
  
def delete\_login\_success():  
    login\_success\_screen.destroy()  
     
def delete\_password\_not\_recognised():  
    password\_not\_recog\_screen.destroy()  
  
  
def delete\_user\_not\_found\_screen():  
    user\_not\_found\_screen.destroy()  
  
def delete\_generate\_time\_table():  
    generate\_time\_table.destroy()  
     
# Designing Main(first) window  
  
def main\_account\_screen():  
    global main\_screen  
    main\_screen = Tk()  
    main\_screen.geometry("300x250")  
    main\_screen.title("Account Login")  
    Label(text="Intelligent Time Table Maker", bg="blue", width="300", height="2", font=("Calibri", 13)).pack()  
    Label(text="").pack()  
    Button(text="Faculty Login", height="2", width="30",command=login).pack()  
    Label(text="").pack()  
    Button(text="Student Time Table", height="2", width="30",command=delete\_login\_success).pack()  
    Label(text="").pack()      
    Button(text="Register new ID Login", height="2", width="30",command=register).pack()  
    Label(text="").pack()  
    main\_screen.mainloop()  
  
  
main\_account\_screen()

**Outputs**

