Correctness And Complexity

Correctness:-

Initialization:-

The function is initialized by checking that we have available slots of classRooms, classes (like sessions 17,18, 19), timeTable and the course we are dealing with and the teachers available in those time slots.

Maintenance:-

If we have all slots available that we have mentioned above then generateTable function will start inserting the courses iteratively into the timeTable according to their creditHours and if all credit hours are been used for that course then the function will delete that course specific from the array of courses in the respective iteration. One iteration will work keeping in view the upper limits and insert the course in respective positions.

Termination:-

The function will be terminated when all slots of all classRooms are used or no other teacher is available or all courses are used which means the courses list is empty.

Complexity:-

The program has the main function as Generate timeTable which is further using sub modules as checkTeacher and checkCourse. An instance of the generateTable problem is a four-tuple (cl, t.len, y.len, i*j)

where

cl is the number of classRooms,

t.len is the number of teachers

y.len is the number of courses

And timeTable i*j denoting the number of courses(y) taught by teacher t to class cls.(i is the number days and j is number of hours in ith day)

with ‘y’ denoting the number of lessons taught by teacher ‘t’ to class ‘cl’. The time slot assignment (subject+teacher positioning) in the timetable is modeled by a set of variables (cls, t, y) , (for cls = 1…….cl; t = 1…….t.length; y= 1…….y.length) with (cls, t, y) = 1 if a lesson for class

'cls' taught by teacher 't' is taught at specific time slot, and (cls, t, y) = 0 otherwise. The objective is to find a time slot assignment satisfying the following: