

Base - Case :

GenerateClassTimetable(class, Free Rooms, Busy Rooms)
Total courses = 5

x = 0

for i = 1 to 5

for j = 1 to 8

classesUsed = 0

for k = 1 to 3

while timetable[i][j][k] < 0 False

while classesUsed != 0

return timetable[0][0][0]

counter = 0

Inductive Step.

GenerateClassTimeTable (¹class, ¹freeRooms, ¹BusyRooms)

y = 1

curr Course = class.courses[1]

for i = 1 to S

for j = 1 to 8

classes used = 0

for k = 1 to 3 (no. of sections)

while TimeTable[i][j][k] < 0

if (True)

L = F.pop()

class Used = classes Used + 1 (1).

e.g (value)

counter

y++ = 1

while class 1 = 0.

return order, counter = 1

Conclusion:

GenerateClassTimeTable(ⁿclass, ⁿFreeRooms, ⁿBusyRooms)

$y = n$

allCourse = class.courses[n]

for $i = 1$ to 5

for $j = 1$ to 8

classesUsed = 0

for $k = 1$ to 3

while TimeTable[i][j][k] < 0

if (True)

$l = \text{FreeRooms.pop}()$

classesUsed = classesUsed + 1

(Subjects)

counter

$y++$

while class != 0

return = Subjects, counter = n