DESCRIPTION OF THE SOLUTION

Juan Pablo Gómez
Santiago Alzate

We would describe our solution as a Greedy algorithm. The main idea of the algorithm is to find the best possible available room for each lesson, having a manual of priorities:

1. Complete the hard restriction if possible, taking into account the bes soft restriction.
2. That the room capacity is the best for the lesson and the group.
3. If the hard restriction is not possible, get the best soft restriction available.

This process can be seen in the class ClassRoom, in the methods getClassroom and getFirstClassroom.
We also have priorities at the moment of organizing the rooms:

1. We first organize the lessons which does not depend on a previous lesson, but they have lessons after that depend on them.
2. We organize the lessons that depend on a previous lesson.
3. We organize lessons that doesn't depend on any lesson, and doesn't have lessons depending on them.

There is also a process that happens before al the scheduling which is the reading and interpretation of the files
We stored all the data we get from the files, and classified them as useless data, or important data
For a data to be useless it has to apply to one of the following:

1. Its original classroom is 00000
2. There is no one register to that classroom(list of students is 0)
3. There is no record of them in pa2019.csv

This is the basic components behind our algorithm, and the results can be seen in pa2019new.csv

The link to the repository we work on in github is the following:
https://github.com/jpgomezt/ClassroomScheduling