**Input**: An array of objects containing class information. Call each of these objects *Course.* “Class time” is a property of these objects.

There is another array of objects, call each object *Schedule,* which will hold the possible schedules.

Recursive heuristic solution —>

1. Find the *Course* object with the fewest number of “class times” offered.

2. Check the first “class time” against the time restrictions.

3. If it passes, insert this course and class time into the *Schedules* object.

4. If it doesn’t pass the time restrictions, pick a different “class time” in this object. If there is no more “class times” left, this course does not fit the schedule – hence the schedule is trashed.

4. Remove the *Course* object associate with this “class time” from future searches in this schedule build.

5. Go back to step 1 (this time without the *Course* object we previously used).

6. You are done making the first schedule when all *Course* objects in the initial input have a “class time” inserted in the *Schedule* object.

I think this is the fastest way to find the first solution possible schedule.

To find the next solution work backwards from this initial solution.

1. For the last *Course* object looked at (the object with the most number of class times), move onto the next class time and see if it fits the restrictions and the schedule we already have.

2. If it fits, add a new schedule