PART 1

In this part I decided to implement it and understand it on my own here. I defined facts for the things that I needed.

In order to make everything work I used 2 helper functions. One is called same which checks if there is a similar element within lists and the other is sublist which checks if one list is contained within another.

Here I am putting some of my test cases for all the parts we had to implement.

• Check whether there is any scheduling conflict.

```
auses
?- schedule_conflicts(math1, math2).
true.

?- schedule_conflicts(math1, physics1).
false.

?-
```

Here there is a conflict between math1 and math2 though there is no conflict between math and physics

• Check which classes a student can be assigned.

The main factor here will be in the decision of the equipment. If class is for handicapped students they can attend it otherwise they can't be assigned to it. For non handicapped students there is no problem.

```
?- enroll(sdnt3, physics1).
true.

?- enroll(sdnt3, math2).
false.

?-
```

Handicapped student 3 can only go to handicapped classes math1 and physics1, whereas sdudent 12 can go to any.

• Check which room can be assigned to a given class.

Here I checks what room provides and what a class needs of the equipment in order to implement this. Sublist comparison in here.

```
?- assignroom(programming1,X).
X = m1 ;
X = m2 ;
X = phy1 ;
X = prog2 ;
X = prog3 ;
false.
```

Here these rooms have projectors and programming requires projectors so that is assigned to it.

• Check which room can be assigned to which classes.

```
?- roomAssign(m2).
math1
true ;
programming1
true ;
physics1
true ;
false.
```

Again it check what is needed and provided to assign these rooms.

• Check which classes a student can be assigned.

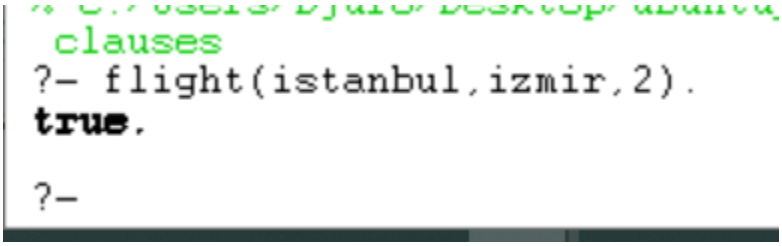
```
?- studentAssign(sdnt3).
math1
true ;
physics1
true ;
false.

?-
```

Classes supporting handicaped students

PART2

I used backtracking here in order to find the route I need alongside with the cost.

Some required examples given in the homework.



Otherwise it lists all the routes

```
X = erzincan,
C = 6 ;
X = antalya,
C = 9 ;
X = izmir,
C = 11 ;
X = diyarbakir,
C = 13 ;
X = istanbul,
C = 13 ;
X = ankara,
C = 17 ;
X = rize,
C = 17 ;
```