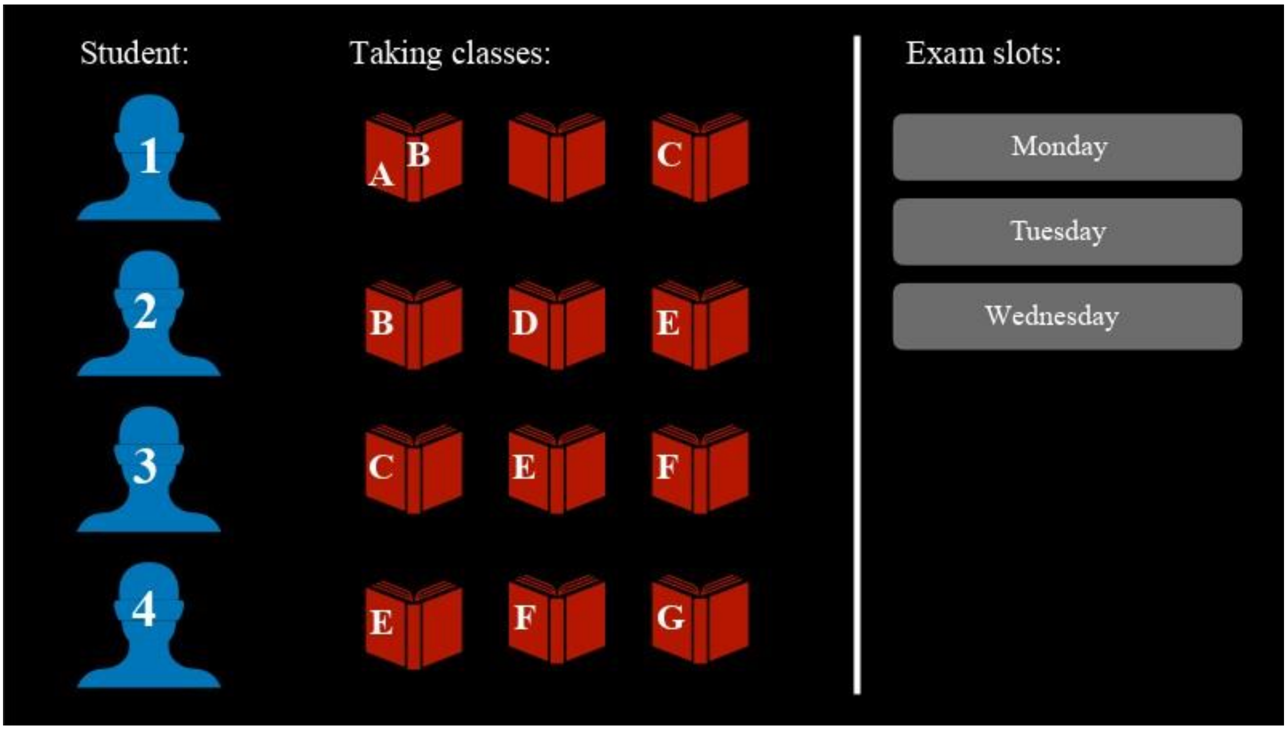
Exam Scheduling



Given Data

- There are 4 students in total
- There are 3 days to schedule exams
- There are total of 7 different exams to schedule.
- On each Day 2 or more exams can be scheduled (But not of same student)

```
VARIABLES = ["A", "B", "C", "D", "E", "F", "G"]
CONSTRAINTS = [
   ("A", "B"),
   ("A", "C"),
   ("B", "C"),
   ("B", "D"),
   ("B", "E"),
   ("C", "E"),
   ("C", "F"),
   ("D", "E"),
   ("E", "F"),
   ("E", "G"),
   ("F", "G")
]
```

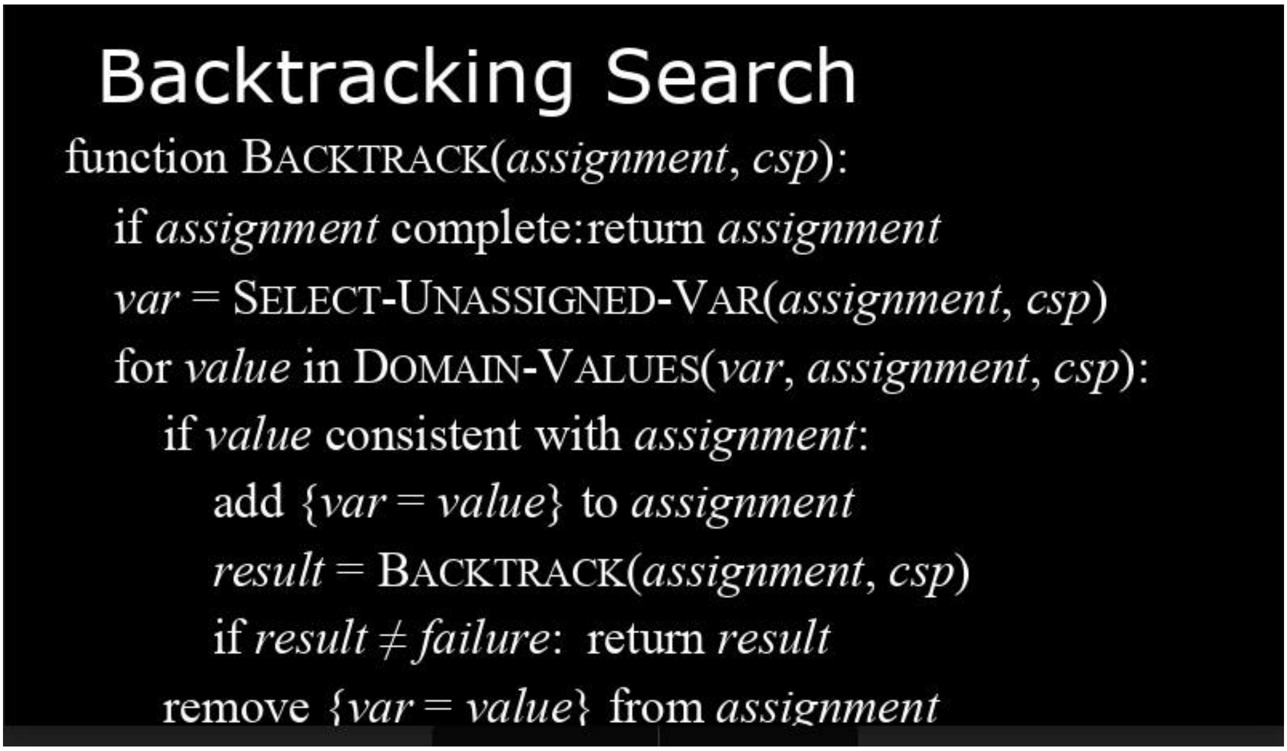
Task to perform

You need to implement hard constraints **(Complete Assignment)** on above given data to schedule exams on given 3 slots of all students using **Backtracking Algorithm (**Given below) so that no clashes would be there after final scheduling.

Exemplary Hard Constraint

Single exam must not be schedule at two different slots on same day. On
any day, not more than one exam can be scheduled of any student.

Backtracking Algorithm



Backtracking Search

```
function BACKTRACK(assignment, csp):
    if assignment complete:return assignment
    var = SELECT-UNASSIGNED-VAR(assignment, csp)
    for value in DOMAIN-VALUES(var, assignment, csp):
        if value consistent with assignment:
            add {var = value} to assignment
            result = BACKTRACK(assignment, csp)
            if result $\neq$ failure:  return result
        remove {var = value} from assignment
```