Assignment2 | HY-487

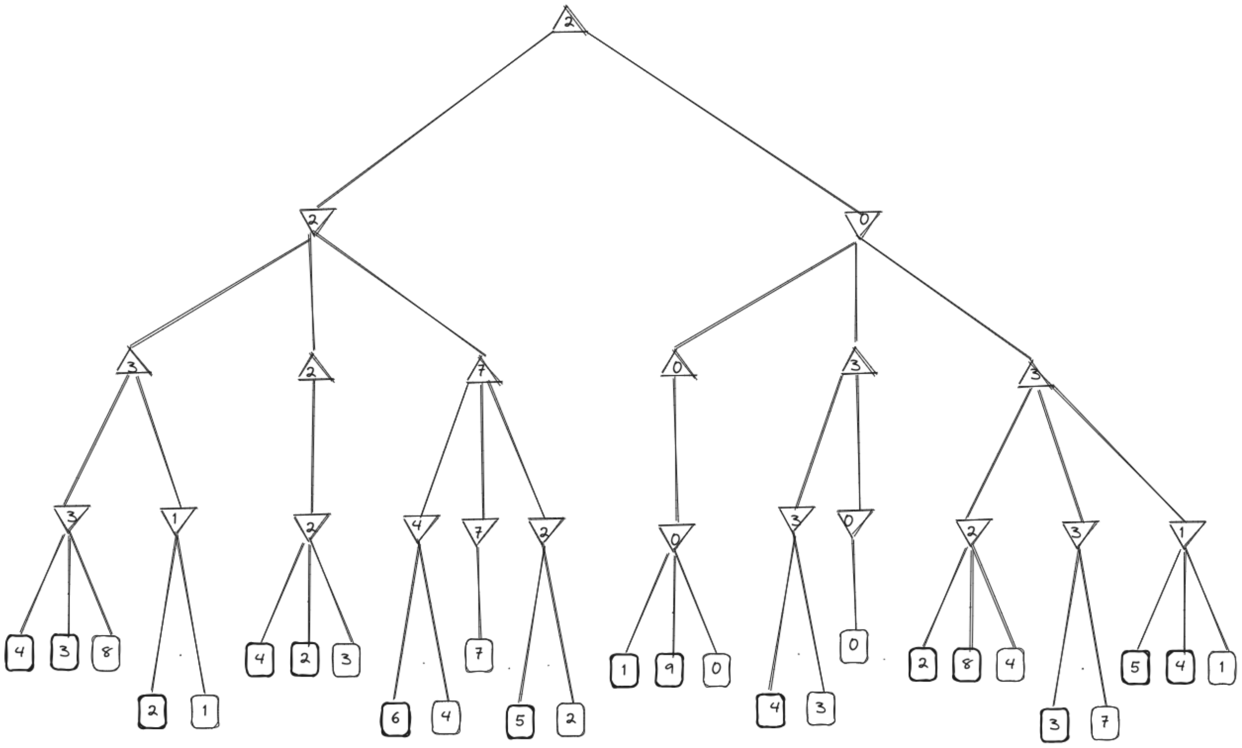
Stivaktakis Giorgos | csd4300

Exercise 1

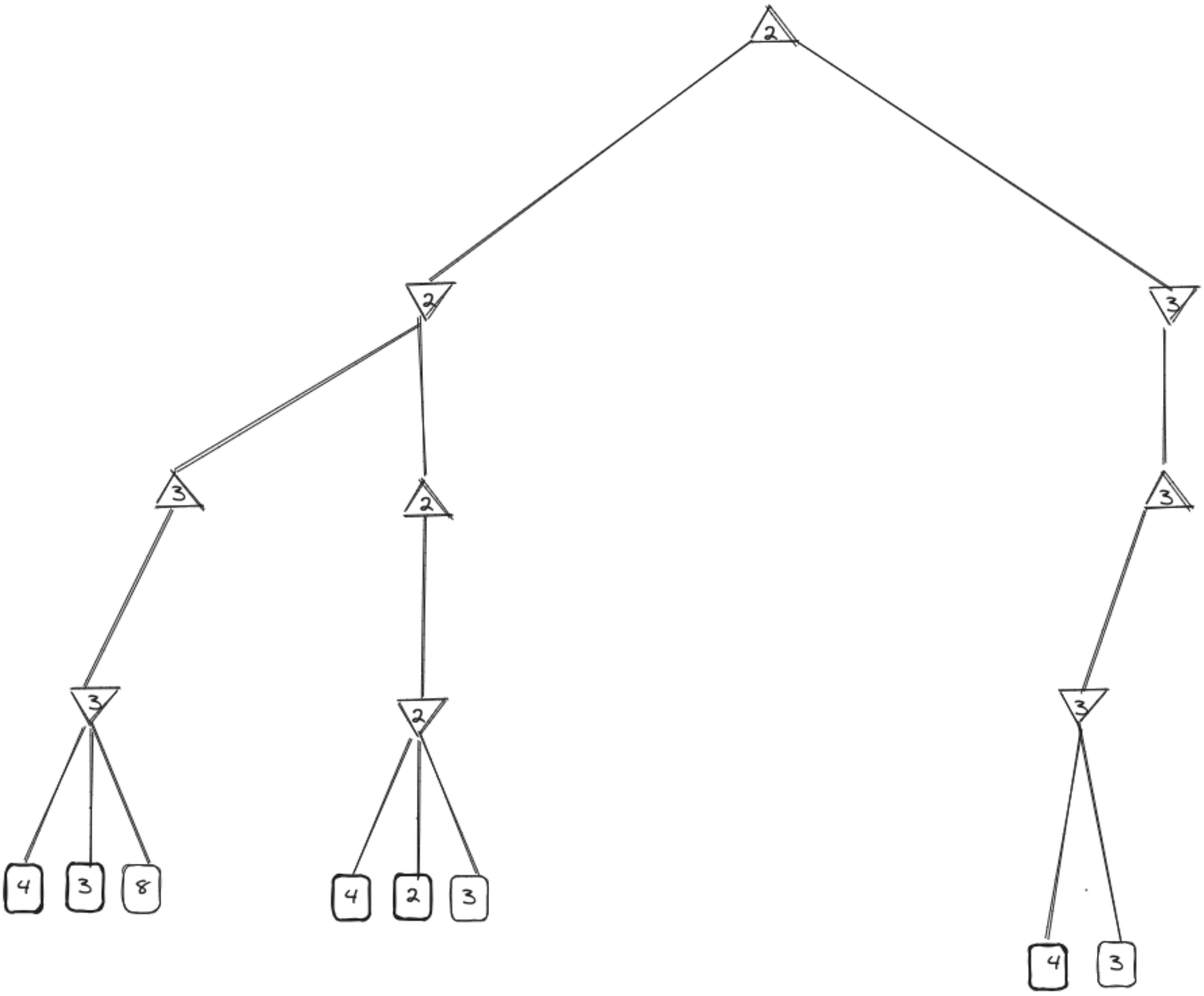
1. **True**. Because in a fully observable, turn-taking, zero-sum game between two perfectly rational players, each player will make the best decision possible (perfect strategy).Therefore knowing the opponents strategy doesn't provide any advantage since the player will still make the same moves.
2. **False.** Since RNG(Random Number Generator) exist in the game ,when you rolling the dice , it doesnt matter if you make the best moves possible you can still lose.

Exercise 2

1. Without alpha-beta pruning. **[2]**



2. With alpha-beta pruning. **[2]**



Exercise 3

1. Variables : The Students (S1,S2,S3,S4,S5)

Domains :

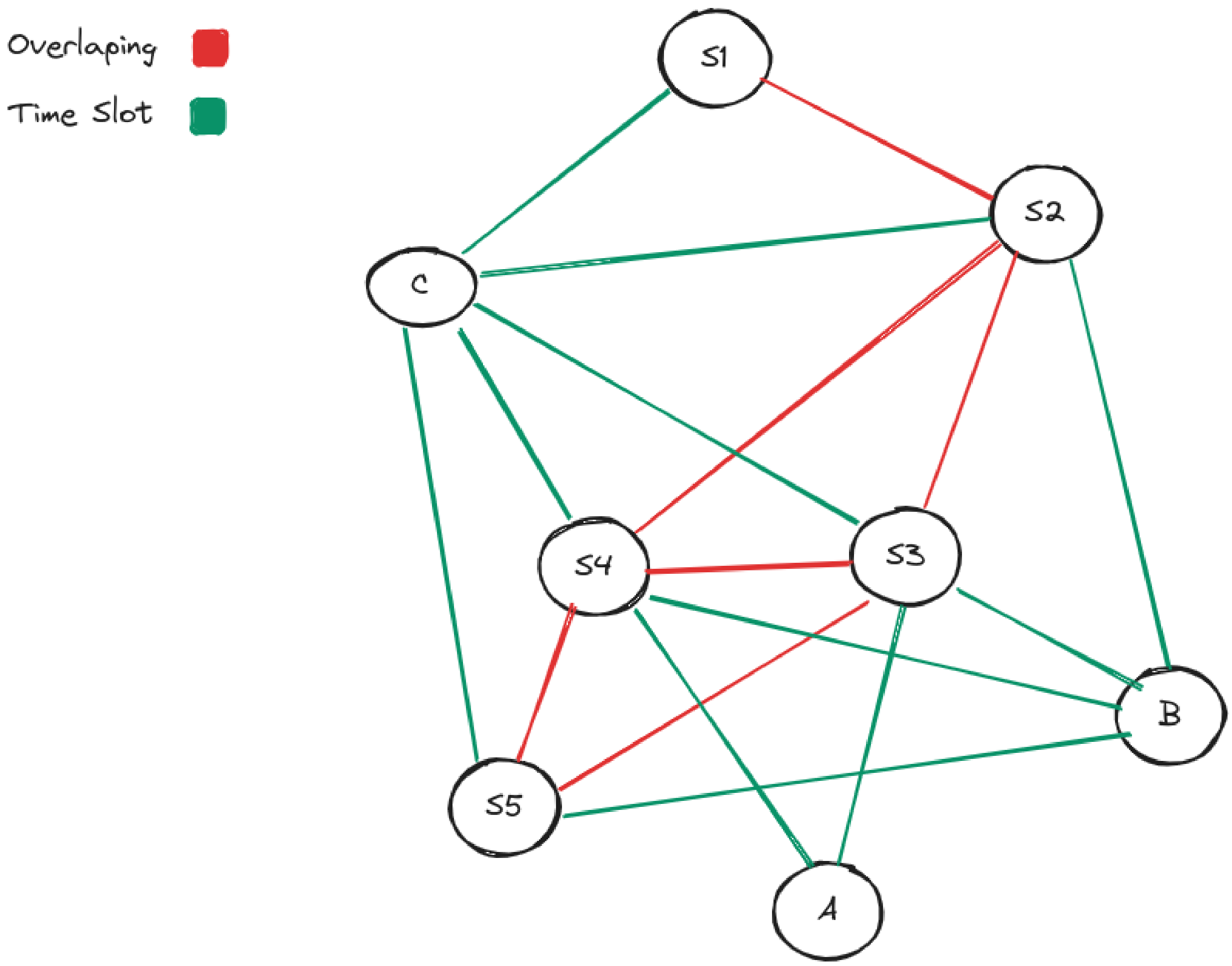
S	TA
S1	{C}
S2	{B, C}
S3	{A, B, C}
S4	{A, B, C}
S5	{B, C}

Constraints :

- Time slot : TA shouldnt be assigned on overlapping time
- Expertise : TAs must be assign to students within their expertise

2. Nodes : Students , Edges : a constraint that must be consider when assign a TA

- Time slot (Overlapping) :
    - S1-S2 { 8:00-9:00, 8:30-9:30}
    - S2-S3-S4 { 8:30-9:30, 9:00-10:00, 9:00-10:00}
    - S3-S4-S5 {9:00-10:00, 9:00-10:00, 9:30-10:30}
- Expertise :
    - S1-S2-S3-S4-S5 {C}
    - S2-S3-S4-S5 {C,B}
    - S3-S4 {A,B,C}



3. Domains : S TA

- S1 {C}
- S2 {B} (Overlapping with S1)
- S3 {A, C} (depending on S4)
- S4 {A, C} (depending on S3)
- S5 {B} (Overlapping with S3,S4)

4. One Solution could be :

    - S1 : 8:00-9:00am with **C**
    - S2 : 8:30-9:30am with **B**
    - S3 : 9:00-10:00am with **A**
    - S4 : 9:00-10:00am with **C**
    - S5 : 9:30-10:30am with **B**

5. Each decision affects only one branch of the tree (so can be solved w/o backtracking)

6. Tree composition.

Exercise 4

The performance of each algorithm is a trade-off between time complexity and space complexity. Backtracking with heuristics seems to be better, offering improved time efficiency without too much memory use.The most suitable algorithm depends on the nature of the puzzles. For simpler puzzles, DFS might work just fine, but for more complex puzzles, backtracking with heuristics generally works much better.