

Pre-LAB:-

(1) We need to Unify  $P(X, Y, Y)$  with  $P(a, Z, b)$ .

Unification:-

Unification is a process of making two different logical atomic expressions identical by finding a substitution.

Unification depends on the substitution process.

It takes 2 literals as input and makes them identical using substitution.

Let  $\psi_1$  and  $\psi_2$  be two atomic sentences and  $\alpha$  be a unifier such that  $\psi_1 \alpha = \psi_2 \alpha$  then it can be expressed as  $\text{UNIFY}(\psi_1, \psi_2)$ .

then it can be expressed as

$\text{UNIFY}(\psi_1, \psi_2)$

No, let  $\psi_1 = \text{king}(x)$ ,  $\psi_2 = \text{king}(\text{john})$

Substitution  $\theta = \{ \text{john}/x \}$  is a unifier for these atoms and applying this substitution, and both expressions will be identical.

We have some conditions for

Unification :- like

- \* Predicate symbol must be same, atoms or expression with different predicate symbol can never be unified.

- \* Number of arguments in both expressions must be identical.

- \* Unification will fail if there are 2 similar variables present in the same expression.



## INLAB:-

We are a problem where:-

A, B, C, D & E are 5 thieves living on different floors of a 5 storied apartment. They escaped from the police and are hiding in same building.

The problem statements are:-

- \* A doesnot live on top floor
- \* B doesnot live on bottom floor
- \* C doesnot live on either top or bottom floor.
- \* D lives on higher floor than B
- \* E doesnot lie on a floor adjacent to C
- \* C doesnot live on a floor adjacent to B.

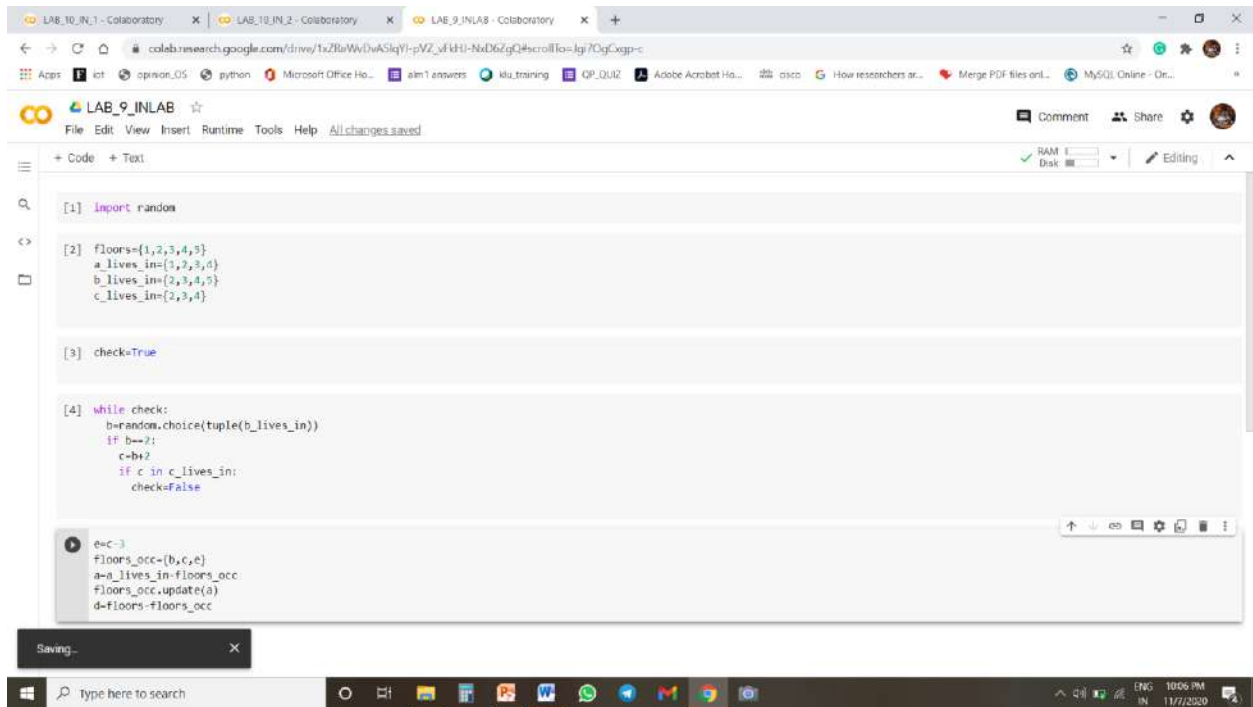
We need to write the python code, to help the policemen to find out where do the thieves live, so let's implement the problem using the given problem statements and the output comes in such a way that, we will get the order in which thieves stay in the apartment.

### POST LAB:-

sol We have 2 expressions  $P(x)$  and  $P(y)$ . One way to unify these is to substitute any constant expression for  $x$  and  $y$  :  $\{fred/x, fred/y\}$ . But this is not the most general unifier, because if we substitute any variable for  $x$  &  $y$  we get a more general unifier.

$\{z/x, z/y\}$ . The first unifier  
is a valid unifier, but it would lessen  
the generality of inferences that  
we might want to make.

## 190030117-AILAB 9 INLAB:OUTPUT:



The screenshot shows a Google Colaboratory notebook titled "LAB\_9\_INLAB". The notebook contains four code cells. The first cell imports the random module. The second cell initializes a list of floors and the initial lives for three players (a, b, and c). The third cell sets a check variable to True. The fourth cell is a while loop that continues as long as the check is True. Inside the loop, a random choice is made from the tuple of lives, and if it's player b, their lives are decremented by 2. If player c is chosen, their lives are also decremented by 2, and the check is set to False. The output of the code is displayed in a separate box at the bottom of the notebook, showing the final state of the game after 3 iterations.

```
[1] import random

[2] floors={1,2,3,4,5}
a_lives_in={1,2,3,4}
b_lives_in={2,3,4,5}
c_lives_in={2,3,4}

[3] check=True

[4] while check:
    b=random.choice(tuple(b_lives_in))
    if b==2:
        c=b+2
        if c in c_lives_in:
            check=False

e=0
floors_occ={b,c,e}
a=a_lives_in-floors_occ
floors_occ.update(a)
d=floors-floors_occ
```

Saving...