Name : Rasika Santosh Thakur

Class : S.Y. B.Tech

Branch : AI & DS 'B'

Roll No : 71

Subject : Operating Systems

Assignment on Unit III

Q.1 Consider following snapshot of System

Process	Allocation				Max				Available	Need			
	A	B	C	D	A	B	C	D		A	B	C	D
P_0	3	0	1	4	5	1	1	7	0,3,0,1	2	1	0	3
P_1	2	2	1	0	3	2	1	1		1	0	0	1
P_2	3	1	2	1	3	3	2	1		0	2	0	0
P_3	0	5	1	0	4	6	1	2		4	1	0	2
P_4	4	2	1	2	6	3	2	5		2	1	1	8

Using Banker's Algorithm determine whether or not each of following state is safe. Illustrate in which order process may completed. Otherwise illustrate why state is unsafe.

1) Available = (0, 3, 0, 1)

2) Available = (1, 0, 0, 2)

→ a) available (0,3,0,1)

work = available = 0,3,0,1.

Each time when process is in safe state, work is updated.

Need ≤ Work then Work = Work + Allocation

P_0 = Need = 2,1,0,3 & Work = 0,3,0,1

Need > Work

[Unsafe]

P_1 = Need = 1,0,0,1 & Work = 0,3,0,1

Need > Work

[unsafe]

P_2 = Need = 0,2,0,0 & Work = 0,3,0,1

Need < Work

[safe]

Work = Work + Allocation

= 0,3,0,1 + 3,1,2,1

= 3,4,2,2

finish [P2] = true

P_3 = Need = 4,1,0,2 & Work = 3,4,2,2

Need > Work

[unsafe state]

P_4 = Need = 2,1,1,3 & Work = 3,4,2,2

Need > work [unsafe state]

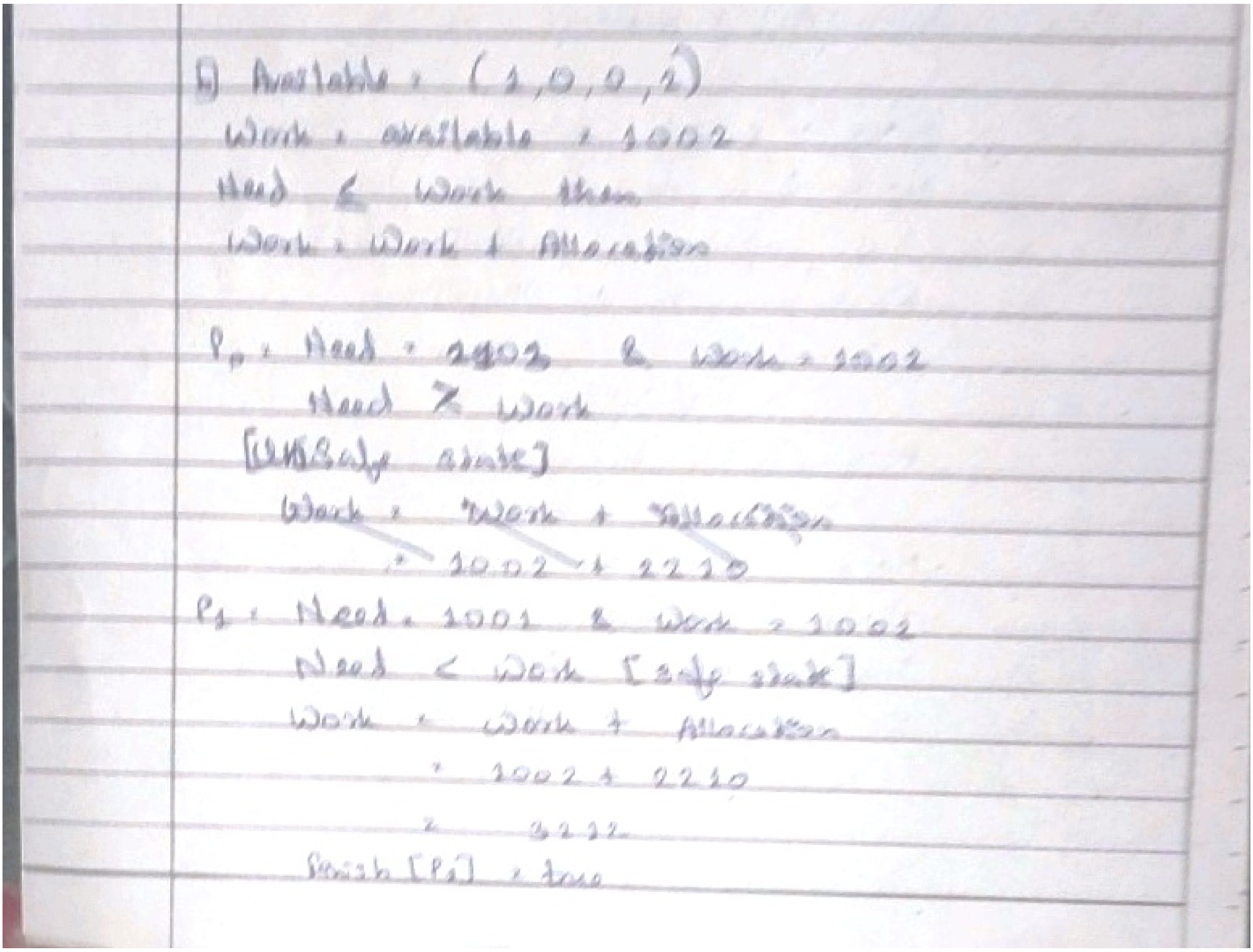
P_1 = Need = 1,0,0,1 & Work = 3,4,2,2

Need < work [safe state]

Work = 3,4,2,2 + 2,2,1,0

= 5,6,3,2

As, Need of Resource type 'D' of process 'P_0' & 'P_4' is greater than need, it will never complete it's execution. So, system is in unsafe state.



b) Available = (1,0,0,2)
Work = available = 1002
Need ≤ Work then
Work = Work + Allocation

P_0 = Need = 2102 & Work = 1002
Need > Work
[UnSafe state]
~~Work = Work + Allocation~~
~~= 1002 + 2210~~

P_1 = Need = 1001 & Work = 1002
Need < Work [safe state]
Work = Work + Allocation
= 1002 + 2210
= 3212
finish [P_1] = true

P_2 = Need = 0200 & Work = 3212

Need < Work

[Safe State]

Work = Work + Allocation

= 3212 + 3121

= 6333

Finish [P_2] = true

P_3 = Need = 4102 & Work = 6333

Need < Work

[Safe State]

Work = Work + Allocation

= 6333 + 0510

= 6843

Finish [P_3] = true

P_4 = Need = 2113 & Work = 6843

Need < Work

[Safe State]

Work = Work + Allocation

= 6843 + 4212

= 1010 55

Finish [P_4] = true

P_0 = Need = 2103 & Work = 101055

Need < Work

[Safe State]

Work = Work + Allocation

= 10 10 5 5 + 3 0 1 4

= 13 10 6 9

Finish[P_0] = true

∴ Safe Sequence = P_1, P_2, P_3, P_4, P_0

Q.2 Write a short note on Semaphore

→ - Semaphore is a hardware Solution.

- It is written or given to critical section problem.
- It is a synchronization tool used in concurrent programming to access control of shared resources among multiple threads & process.
- Main purpose is to access provide synchronization & mutual exclusion in multi-threaded environment.
- It prevents race condition & data corruption.
- It has 2 fundamental operations:

  i] Wait &

  ii] Signal
- Wait decrements semaphore value to stop until value becomes non-negative where as signal increments semaphore value to unblock process.

- There are two types of Semaphore

i] Binary Semaphore

- Also known as mutex & it has two states 0 & 1 to controll critical section

0 - locked

1 - unlocked

ii] Counting semaphore

- It is used when multiple instances of resources can execute it's critical section based on number of instances available.