Banker's Algo → Deadlock Avoidance
→ Deadlock Detection

Safe → If deadlock not occur
Unsafe → If deadlock occurs.

Total A = 10, B = 5, C = 7

Process	Allocation			Max Need			Available			Remaining Need		
	A	B	C	A	B	C	A	B	C	A	B	C
P_1	0	1	0	7	5	3	3	3	2	7	4	3
P_2	2	0	0	3	2	2	5	3	2	1	2	2
P_3	3	0	2	9	0	2	7	4	3	6	0	0
P_4	2	1	1	4	2	2	7	4	5	2	1	1
P_5	0	0	2	5	3	3	7	5	5	5	3	1
	7	2	5				10	5	7			

First available → (3, 3, 2)

P_2 fulfil so available = (5, 3, 2)

P_4 fulfil so " = (5, 3, 2) + (2, 1, 1) = (7, 4, 3)

P_5 fulfil " = (7, 4, 3) + (0, 0, 2) = (7, 4, 5)

P_1 fulfil " (7, 4, 5) + (0, 1, 0) = (7, 5, 5)

P_3 fulfil " (7, 5, 5) + (3, 0, 2) = (10, 5, 7)

Total

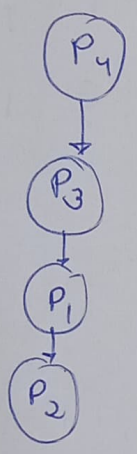
So ans is correct and

safe sequence = $P_2 \rightarrow P_4 \rightarrow P_5 \rightarrow P_1 \rightarrow P_3$

Total: A = 10, B = 8, C = 5

Process	Allocation			Max Need			Available			Current Need		
	A	B	C	A	B	C	A	B	C	A	B	C
P_1	2	1	0	8	6	3	4	3	2	6	5	3
P_2	1	2	2	9	4	3	7	3	3	8	2	1
P_3	0	2	0	5	3	3	7	5	3	5	1	3
P_4	3	0	1	4	2	3	9	6	3	1	2	2
	6	5	3									

First available = (4, 3, 2)



So rem avail = (4, 3, 2) + (3, 0, 1) = (7, 3, 3)

So rem avail = (7, 3, 3) + (0, 2, 0) = (7, 5, 3)

So rem avail = (7, 5, 3) + (2, 1, 0) = (9, 6, 3)

So rem avail = (9, 6, 3) + (1, 2, 2) = (10, 8, 5) Total

So Safe seq = $P_4 \rightarrow P_3 \rightarrow P_1 \rightarrow P_2$