

PRACTICE PROBLEMS

Consider the following set of processes:

Set 1

Process	Allocation				Maximum				Available (Work)			
	R ₁	R ₂	R ₃	R ₄	R ₁	R ₂	R ₃	R ₄	R ₁	R ₂	R ₃	R ₄
P ₀	0	1	1	0	0	2	1	0	1	5	2	0
P ₁	1	2	3	1	1	6	5	2				
P ₂	1	3	6	5	2	3	6	6				
P ₃	0	6	3	2	0	6	5	2				
P ₄	0	0	1	4	0	6	5	6				

Set 2

Process	Allocation				Maximum				Available (Work)			
	R ₁	R ₂	R ₃	R ₄	R ₁	R ₂	R ₃	R ₄	R ₁	R ₂	R ₃	R ₄
P ₀	0	1	1	0	0	2	1	0	1	3	1	0
P ₁	1	4	4	1	1	6	5	2				
P ₂	1	3	6	5	2	3	6	6				
P ₃	0	6	3	2	0	6	5	2				
P ₄	0	0	1	4	0	6	5	6				

Set 3

Process	Allocation			Maximum			Available (Work)		
	R ₁	R ₂	R ₃	R ₁	R ₂	R ₃	R ₁	R ₂	R ₃
P ₀	1	1	2	4	3	3	2	1	0
P ₁	2	1	2	3	2	2			
P ₂	4	0	1	9	0	2			
P ₃	0	2	0	7	5	3			
P ₄	1	1	2	1	1	2			

Set 4

Process	Allocation			Maximum			Available (Work)		
	R ₁	R ₂	R ₃	R ₁	R ₂	R ₃	R ₁	R ₂	R ₃
P ₀	1	0	1	2	1	1	2	1	1
P ₁	2	1	2	5	4	4			
P ₂	3	0	0	3	1	1			
P ₃	1	0	1	1	1	1			

Set 5

Process	Allocation			Maximum			Available (Work)		
	R ₁	R ₂	R ₃	R ₁	R ₂	R ₃	R ₁	R ₂	R ₃
P ₀	2	1	2	1	0	1	2	1	2
P ₁	3	2	4	0	0	1			
P ₂	4	2	1	1	1	1			

For each set, compute whether the system is safe or not using the Banker's algorithm. Determine the sequence if it is safe.

PRACTICE PROBLEMS

Consider the following set of processes:

Set 1

Process	Allocation				Maximum				Available (Work)			
	R ₁	R ₂	R ₃	R ₄	R ₁	R ₂	R ₃	R ₄	R ₁	R ₂	R ₃	R ₄
P ₀	0	1	1	0	0	2	1	0	1	5	2	0
P ₁	1	2	3	1	1	6	5	2				
P ₂	1	3	6	5	2	3	6	6				
P ₃	0	6	3	2	0	6	5	2				
P ₄	0	0	1	4	0	6	5	6				

Process	Allocation				Maximum				Available (Work)			
	R ₁	R ₂	R ₃	R ₄	R ₁	R ₂	R ₃	R ₄	R ₁	R ₂	R ₃	R ₄
P ₀	0	1	1	0	0	2	1	0	1	5	2	0
P ₁	1	2	3	1	1	6	5	2	1	6	3	0
P ₂	1	3	6	5	2	3	6	6	2	8	6	1
P ₃	0	6	3	2	0	6	5	2	3	11	12	6
P ₄	0	0	1	4	0	6	5	6	3	17	15	8

3 17 16 12

>

Need				
R ₁	R ₂	R ₃	R ₄	
0	1	0	0	✓
0	4	2	1	✓
1	0	0	1	✓
0	0	2	0	✓
0	6	4	2	✓

P₀ → P₁ → P₂ → P₃ → P₄

Set 2

Process	Allocation				Maximum				Available (Work)			
	R ₁	R ₂	R ₃	R ₄	R ₁	R ₂	R ₃	R ₄	R ₁	R ₂	R ₃	R ₄
P ₀	0	1	1	0	0	2	1	0	1	3	1	0
P ₁	1	4	4	1	1	6	5	2	1	4	2	0
P ₂	1	3	6	5	2	3	6	6	2	8	6	1
P ₃	0	6	3	2	0	6	5	2	3	11	12	6
P ₄	0	0	1	4	0	6	5	6	3	17	15	8

3 17 16 12

Need

R ₁	R ₂	R ₃	R ₄	
0	1	0	0	✓
0	✓	1	1	✓
1	0	0	1	✓
0	0	✓	0	✓
0	6	4	2	✓

P₀ → P₁ → P₂ → P₃ → P₄

Set 3

Process	Allocation			Maximum			Available (Work)		
	R ₁	R ₂	R ₃	R ₁	R ₂	R ₃	R ₁	R ₂	R ₃
P ₀	1	1	2	4	3	3	2	1	0
P ₁	2	1	2	3	2	2	4	2	2
P ₂	4	0	1	9	0	2	5	3	4
P ₃	0	2	0	7	5	3	6	4	6
P ₄	1	1	2	1	1	2	10	4	7

10 6 7

Need

R ₁	R ₂	R ₃	
3	2	1	✓
1	1	0	✓
5	0	1	✓
7	3	3	✓
0	0	0	✓

P₁ → P₄ → P₀ → P₂ → P₃

Set 4

Process	Allocation			Maximum			Available (Work)		
	R ₁	R ₂	R ₃	R ₁	R ₂	R ₃	R ₁	R ₂	R ₃
P ₀	1	0	1	2	1	1	2	1	1
P ₁	2	1	2	5	4	4	3	1	2
P ₂	3	0	0	3	1	1	6	1	2
P ₃	1	0	1	1	1	1	7	1	3

9 2 5

Need			
R ₁	R ₂	R ₃	
1	1	0	✓
3	3	2	✗
0	1	1	✓
0	1	0	✓
P ₀ → P ₂ → P ₃ → P ₁			

Set 5

Process	Allocation			Maximum			Available (Work)		
	R ₁	R ₂	R ₃	R ₁	R ₂	R ₃	R ₁	R ₂	R ₃
P ₀	2	1	2	1	0	1	2	1	2
P ₁	3	2	4	0	0	1			
P ₂	4	2	1	1	1	1			

Need		
R ₁	R ₂	R ₃