AOS
~~DATE~~

Resource -Reqti

DATE 01/2/24

Eg:	Allocation			Max			Available			Need		
	A	B	C									
P_0	0	1	0	7	5	3	~~7~~ ~~3~~	~~5~~ ~~3~~	~~5~~ ~~2~~	7	4	3
P_1	2	0	0	3	2	2	~~5~~	~~3~~	~~2~~	1	2	2
P_2	3	0	2	9	0	2	10	5	7	6	0	0
P_3	2	1	1	2	2	2	~~7~~	~~4~~	~~3~~	0	1	1
P_4	0	0	2	4	3	3	~~7~~	~~4~~	~~5~~	4	3	1

work = 3 3 2 , finish = [$\underset{P_0}{\overset{T}{\cancel{F}}}$ $\underset{P_1}{\overset{T}{\cancel{F}}}$ $\underset{P_2}{\overset{T}{\cancel{F}}}$ $\underset{P_3}{\overset{T}{\cancel{F}}}$ $\underset{P_4}{\overset{T}{\cancel{F}}}$]
(available).

⇒ 7 4 3 ≤ 3 3 2 × → so P_0 have to wait.

1 2 2 ≤ 3 3 2 ✓ → P_1 will be allocated resource. and finished.

∴ work = work + allocation
= 3 3 2 + 2 0 0
= [5 3 2]

~~{7 4 3 ≤ 5 3 2 × wait}~~ ×

6 0 0 ≤ 5 3 2 × wait

0 1 1 ≤ 5 3 2 ✓ → P_3 will be allocated resource finished

∴ work = work + allocation
= 5 3 2 + $\overset{2}{\cancel{0}}$ 1 1

work = [$\overset{7}{\cancel{8}}$ 4 3]

4 3 1 ≤ 7 4 3 ✓ → P_4 will be allocated.

∴ work = work + allocation
= 7 4 3 + 0 0 2
= [7 4 5]

⇒ 7 4 3 ≤ 7 4 5 ✓ → P_0 executes.

∴ work = work + allocation
= 7 4 5 + 0 1 0
= [7 5 5]

→ 6 0 0 ≤ 7 5 5 ✓ → P_2 resource will be allocated.

∴ work = 7 5 5 + 3 0 2
= [10 5 7]

⇒ Resource- Request Algo^r:

→ when A request is made by Pi, following actions are taken

① if (request ≤ need), go to step ②
otherwise raise the error
(Exceeded max^m limit);

② if (request < available), go to step ③
otherwise Pi must wait.

③ Allocate resources to Pi (pretend).

④ (Available = Available - request,)
then (allocation$[i]$ = allocation$[i]$ + request[i])

& {need[i] = need [i] - request [i]}

→ Check the resulting resource allocation is safe or not.
If yes allocate resource and if no then old state is restored.

Q. Suppose in the prev. ques^s. P_1 request 1 additional instance of A and 2 instances of C,

P_1 = [1 0 2]

req^s(P_1)
→ here, 1 0 2 ≤ need(P_1)
1 0 2 ≤ 1 2 2 ✓

then check, $\text{req}(P_i) < \overset{\text{available}}{\text{req}(P_1)}$

1 0 2 < 3 3 2 ✓

means; allocate resources (pretend).

$\Rightarrow$ $\underset{(P_1)}{\text{available}}$ = availb − reqs
= 3 3 2 − 1 0 2
= [2 3 0] → work.

$\underset{(P_1)}{\text{allocation}}$ = alloc + reqs
= 2 0 0 + 1 0 2
= [3 0 2]

$\underset{(P_1)}{\text{need}}$ = need − reqs
= 1 2 2 − 1 0 2
= [0 2 0]

$\Rightarrow$ now check for safe state,

need ≤ available.

0 2 0

7 4 3 ≤ 2 3 0 X → P_0 wait

0 2 0 ≤ 2 3 0 ✓ → P_1 exes.

work = work + allocation
= $\overset{2}{\cancel{3}}\ \overset{3}{\cancel{3}}\ \overset{0}{\cancel{2}}$ + 3 0 2
= $\cancel{6}\ \cancel{3}\ \cancel{4}$ 5 3 2

6 0 0 ≤ $\overset{5}{\cancel{6}}\ \overset{3}{\cancel{3}}\ \overset{2}{\cancel{4}}$ X → P_2 ~~exe~~c wait

~~work = work + alloc~~

= ~~6 3 4 + 3 0 2~~

= ~~9 3 6~~

0 1 1 ≤ $\overset{5}{\cancel{9}}\ \overset{3}{\cancel{3}}\ \overset{2}{\cancel{6}}$ ✓ → P_3 exec

work = 5 3 2 + 2 1 1
= 7 4 3

Disk scheduling X

4 3 1 ≤ 7 4 3 → P4 exe

work = 7 4 3 + 0 0 2

= 7 4 5

⇒ 7 4 3 ≤ 7 4 5 → P0 exe

work = 7 4 5 + 0 1 0

= 7 5 5

6 0 0 ≤ 7 5 5 → P2 exe

work = 7 5 5 + 3 0 2

= [10 5 7] Ans.

Safe seq found so it is in safe state.

[P1 → P3 → P4 → P0 → P2]

Eg.

	Allocation A B C	Max	Available	Need
P0	1 0 1	4 3 1	3 3 0	3 3 0
P1	1 1 2	2 1 4	~~4 3 1~~	1 0 2
P2	1 0 3	1 3 3	~~5 4 3~~	0 3 0
P3	2 0 0	5 4 1	~~8 4 6~~	3 4 1

Finish = [~~F~~ T ~~F~~ T ~~F~~ T ~~F~~ T]

⇒ Total resources : 8 4 6

∴ Available = 8 4 6 − (Total allocation)

= 8 4 6 − 5 1 6

= 3 3 0.

⇒ 3 3 0 ≤ 3 3 0 → P0 executes.

work = work + allocation

= 3 3 0 + 1 0 1

= 4 3 1

1 0 ② ≤ 4 3 1 ~~≤~~ P1 ~~exe~~ wait

work = ~~4 3 1 + 1 1 2~~

= ~~5 4 3~~

$0\ 3\ 0 \leq \overset{4}{\cancel{5}}\ \overset{3}{\cancel{4}}\ \overset{1}{\cancel{3}} \rightarrow P_2\ exe^c$

work $= \overset{4}{\cancel{5}}\ \overset{3}{\cancel{4}}\ \overset{1}{\cancel{3}} + 1\ 0\ 3$

$= \cancel{6}\ 4\ \cancel{6}\ 5\ 3\ 4$

$\cancel{6}\ 2\ 0\ 0 \leq \overset{5}{\cancel{6}}\ \overset{3}{\cancel{4}}\ \overset{4}{\cancel{6}} \rightarrow P_3\ exe^c$

work $= \overset{5}{\cancel{6}}\ \overset{3}{\cancel{4}}\ \overset{4}{\cancel{6}} + 2\ 0\ 0$

$= [\overset{7}{\cancel{8}}\ \overset{3}{\cancel{4}}\ \overset{4}{\cancel{6}}]$ Ans.

=) safe seq^n : $[P_0 \rightarrow P_1 \rightarrow P_2 \rightarrow P_3]$ ✓.

$1\ 0\ 2 \leq 7\ 3\ 4 \rightarrow P_1\ exe^c$

work $= 1\ \cancel{0}\ 1\ 2 + 7\ 3\ 4$

$= [8\ \overset{4}{\cancel{3}}\ 6]$ Ans