

Pseudocode for checking safe state in Banker's Algorithm.

- ① initialise p as no. of process (no. of process) \rightarrow global variable
initialise e as no. of elements in one process - global variable
initialise i, j, k, z as a loop variable -
initialise cc, inn_count to 0 for counter variable.
take 2D array all, need, and one-D array avail, pro.
- ② process_check (allocation, need, available, pro, count)
- ③ Repeat step 4 for $z=0$ to $z < p$
- ④ Repeat step for $i=0$ to $i < p$
- ⑤ Repeat step for $j=0$ to $j < e$
- ⑥ \rightarrow check if $pro[i] = 1$
- ⑦ \rightarrow check if $available[j] \geq need[i][j]$
- ⑧ \rightarrow increase inn_count to 1
- ⑨ \rightarrow check if $inn_count = e$
- ⑩ \rightarrow $pro[i] = 0$
- ⑪ \rightarrow print the message "Process : i is terminated"
- ⑫ \rightarrow Count increase to 1
- ⑬ Repeat Step for $k=0$ to $k < p$.
- ⑭ \rightarrow New $avail[k] = avail[k] + allocation[i][k]$.
- ⑮ Initialise inn_count to 0.

- // outside All for loop -
- ①⑥ Repeat step for $i=0$ to $i < P$
 - ①⑦ $\rightarrow cc = cc + pro[i]$
 - ①⑧ check if $cc = 0$
 - ①⑨ \rightarrow print the message "system is in safe state"
 - ②⑦ else print the message "system not in safe state".