**10.** Consider the following process table with number of processes that contains allocation field (for showing the number of resources of type: A, B and C allocated to each process in the table), max field (for showing the maximum number of resources of type: A, B, and C that can be allocated to each process). Write a program to calculate the entries of need matrix using the formula: (Need)i = (Max)i - (Allocation)i

**Program:**

**#include <stdio.h>**

**#define N\_PROCESSES 5**

**#define N\_RESOURCES 3**

**int main()**

**{**

**int allocation[N\_PROCESSES][N\_RESOURCES] = {{1, 1, 2}, {2, 1, 2}, {3, 0, 1}, {0, 2, 0}, {1,**

**1, 2}};**

**int max[N\_PROCESSES][N\_RESOURCES] = {{5, 4, 4}, {4, 3, 3}, {9, 1, 3}, {8, 6, 4}, {2, 2, 3}}; int available[N\_RESOURCES] = {3, 3, 2};**

**int need[N\_PROCESSES][N\_RESOURCES]; int i, j;**

**for(i=0; i<N\_PROCESSES; i++)**

**{**

**for(j=0; j<N\_RESOURCES; j++)**

|  |  |  |  |
| --- | --- | --- | --- |
| P1 | 2 1 2 | 4 3 3 |  |
| P2 | 3 0 1 | 9 1 3 |  |
| P3 | 0 2 0 | 8 6 4 |  |
| P4 | 1 1 2 | 2 2 3 |  |

**{**

**need[i][j] = max[i][j] - allocation[i][j];**

**}**

**}**

**printf("Need matrix:\n"); printf(" A B C\n"); for(i=0; i<N\_PROCESSES; i++)**

**{**

**printf("P%d ", i);**

**for(j=0; j<N\_RESOURCES; j++)**

**{**

**printf("%2d ", need[i][j]);**

**}**

**printf("\n");**

**}**

**return 0; }**

**Output:**

