#include <stdio.h>

int main(){

int nr, pr, i, j, k; // no.of resources, no of processess

printf("Enter no.of resources: ");

scanf("%d", &nr);

int avail[nr];

printf("Enter total allocation of resources: ");

for(i = 0; i < nr; i++){

scanf("%d", &avail[i]);

}

printf("Enter no.of processes: ");

scanf("%d", &pr);

int initial[pr][nr], max\_allocM[pr][nr], needM[pr][nr];

printf("Enter intial allocation matrix: ");

for(i = 0; i < pr; i++){

for(j = 0; j < nr; j++)

scanf("%d", &initial[i][j]);

}

printf("Enter the max need of each process: ");

for(i = 0; i < pr; i++){

for(j = 0; j < nr; j++)

scanf("%d", &max\_allocM[i][j]);

}

for (int i = 0; i < pr; i++) {

for (int j = 0; j < nr; j++) {

avail[j] -= initial[i][j];

}

}

int visited[pr];

for (int i = 0; i < pr; i++) {

visited[i] = 0;

for (int j = 0; j < nr; j++)

needM[i][j] = max\_allocM[i][j] - initial[i][j];

}

//check if a process can be allocated with the avail resources

printf("Safe sequence: ");

for (int k = 0; k < pr; k++) {

for (int i = 0; i < pr; i++) {

if (visited[i] == 0) {

int flag = 1;

for (int j = 0; j < nr; j++) {

if (needM[i][j] > avail[j]) {

flag = 0;

break;

}

}

if (flag) {

printf("%d ", i + 1);

visited[i] = 1;

// update available

for (int j = 0; j < nr; j++)

avail[j] += initial[i][j];

}

}

}

}

return (0);

}