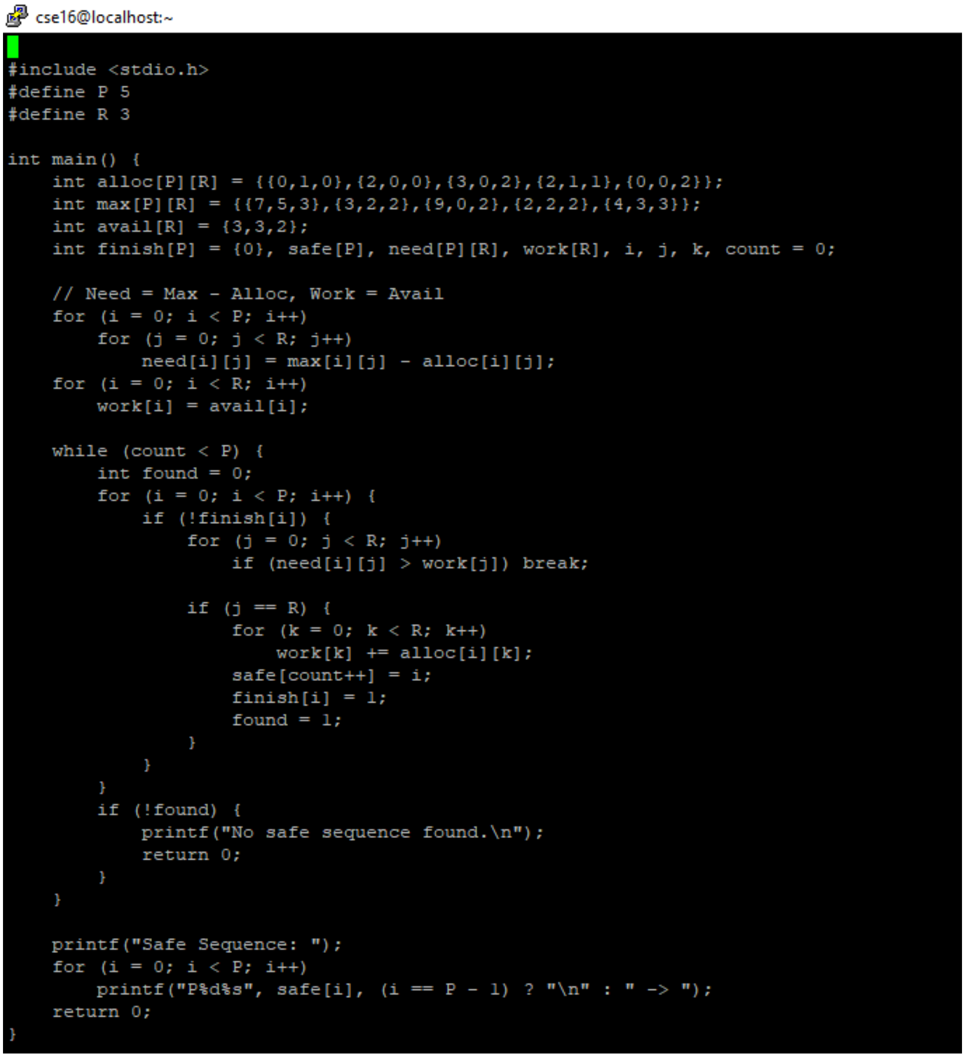
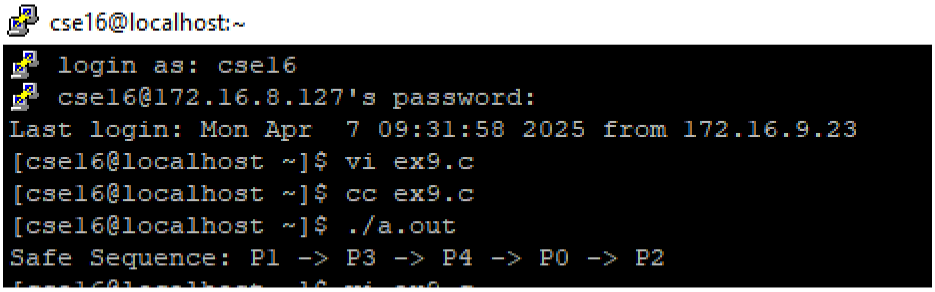
9.

CODE-



```
#include <stdio.h>
#define P 5
#define R 3

int main() {
    int alloc[P][R] = {{0,1,0},{2,0,0},{3,0,2},{2,1,1},{0,0,2}};
    int max[P][R] = {{7,5,3},{3,2,2},{9,0,2},{2,2,2},{4,3,3}};
    int avail[R] = {3,3,2};
    int finish[P] = {0}, safe[P], need[P][R], work[R], i, j, k, count = 0;

    // Need = Max - Alloc, Work = Avail
    for (i = 0; i < P; i++)
        for (j = 0; j < R; j++)
            need[i][j] = max[i][j] - alloc[i][j];
    for (i = 0; i < R; i++)
        work[i] = avail[i];

    while (count < P) {
        int found = 0;
        for (i = 0; i < P; i++) {
            if (!finish[i]) {
                for (j = 0; j < R; j++)
                    if (need[i][j] > work[j]) break;

                if (j == R) {
                    for (k = 0; k < R; k++)
                        work[k] += alloc[i][k];
                    safe[count++] = i;
                    finish[i] = 1;
                    found = 1;
                }
            }
        }
        if (!found) {
            printf("No safe sequence found.\n");
            return 0;
        }
    }

    printf("Safe Sequence: ");
    for (i = 0; i < P; i++)
        printf("P%d%s", safe[i], (i == P - 1) ? "\n" : " -> ");
    return 0;
}
```

OUTPUT-



```
login as: cse16
cse16@172.16.8.127's password:
Last login: Mon Apr  7 09:31:58 2025 from 172.16.9.23
[cse16@localhost ~]$ vi ex9.c
[cse16@localhost ~]$ cc ex9.c
[cse16@localhost ~]$ ./a.out
Safe Sequence: P1 -> P3 -> P4 -> P0 -> P2
```