



22^{ος} ΠΑΝΕΛΛΗΝΙΟΣ ΔΙΑΓΩΝΙΣΜΟΣ ΠΛΗΡΟΦΟΡΙΚΗΣ

ΛΥΣΕΙΣ ΘΕΜΑΤΩΝ ΤΕΛΙΚΗΣ ΦΑΣΗΣ

Θέμα 1ο: Lines man

[25 Μονάδες]

CPP

Κωστάλα Ερριέτα 12^ο ΓΕΛ Πειραιώς

```
/*
USER: 22pdp19
LANG: C++
TASK: lines_man
*/
#include <fstream>
#include <vector>
// #include <math.h>
using namespace std;
int abs (int n) {
    return (n >= 0?n:n * -1);
}
int main() {
    ifstream in("lines_man.in");
    ofstream out("lines_man.out");
    int A; //court size
    in >> A;
    int M; //phases
    in >> M;
//
    vector <int> width;
    int firstdude = 0, seconddude = 0;
    int firstdudepos = A / 2, seconddudepos = A / 2;
    for (int i = 0; i < M; i++) {
        int x;
        in >> x;
        if (x <= firstdudepos || x >= firstdudepos &&
x <= A/2) {
            firstdude += abs(firstdudepos - x);
            firstdudepos = x;
        }
        else {
            firstdude += abs (firstdudepos - A/2);
            firstdudepos = A/2;
        }
        if (x >= seconddudepos || x <= seconddudepos
&& x >= A/2) {
            seconddude += abs(seconddudepos - x);
            seconddudepos = x;
        }
        else {

```

Σελ. 1 από 8



```
seconddude += abs (seconddudepos -  
A/2);  
seconddudepos = A/2;  
}  
}  
out << firstdude << " " << seconddude << endl;  
return 0;  
}
```





ΘΕΜΑ 2ο: Servers

[35 Μονάδες]

CPP

Πόριος Χρήστος 3^ο Γ/σιο Αλεξανδρούπολης (28.6/35)

```
/*
USER: 22pdp33
LANG: C++
TASK: servers,
*/

#include <fstream>
#include <iostream>
#include <stdlib.h>
using namespace std;

struct site
{
    int id;
    long int hits;
};

int compare(const void *a, const void *b)
{
    site sa;
    site sb;
    sa = *(site*)a;
    sb = *(site*)b;
    return sb.hits-sa.hits;
}

int main()
{
    ifstream fin("servers.in");
    ofstream fout("servers.out");
    int M, N, K, i, s, j;
    long int h;
    site sites[10000];

    fin >> M >> N >> K;

    for(i=0;i<M;i++)
    {
        sites[i].id=i;
        sites[i].hits=0;
    }

    for(i=0;i<M;i++)
```

Σελ. 3 από 8





```
{
    for(j=0;j<N;j++)
    {
        fin >> s >> h;
        //cout << s << ' ' << h;
        sites[s].hits+=h;
        //cout << ' ' << sites[s].id << ' ' <<
sites[s].hits << endl;
    }

    qsort(sites,M,sizeof(site),compare);

    for(i=0;i<K;i++)
    {
        fout << sites[i].id << ' ' << sites[i].hits << '\n';
        //cout << sites[i].id << ' ' << sites[i].hits << endl;
    }

    fin.close();
    fout.close();

    return 0;
}
```





ΘΕΜΑ 3ο: Get out!

[40 Μονάδες]

CPP

Αρσένης Γεράσιμος 2° ΓΕΛ Μοσχάτου

```
/*
USER: 22pdp91
LANG: C++
TASK: getout
*/
#include <cstdio>
#include <set>
#include <queue>
#include <cstdlib>

using namespace std;

int n, r;
FILE *fin, *fout;

struct state {
    char park[9][9];
    int moves;
    bool operator==(const state &b) const {
        int i, j;
        for (i=1; i<=n; i++) for (j=1; j<=n; j++) if (park[i][j]!
=b.park[i][j]) return false;
        return true;
    }
    bool operator<(const state &b) const {
        int i, j;
        for (i=1; i<=n; i++) for (j=1; j<=n; j++)
            if (park[i][j]<b.park[i][j]) return true;
            else if (park[i][j]>b.park[i][j]) return false;
        return false;
    }
    void operator=(const state &b) {
        int i, j;
        for (i=1; i<=n; i++) for (j=1; j<=n; j++) park[i][j] =
b.park[i][j];
        moves = b.moves;
    }
};
set<state> s;
queue<state> q;
char dir[4][2] = { {-1, 0}, {1, 0}, {0, -1}, {0, 1} };

void print(state &tmp) {
    for (int i=n; i>=1; i--) {
        for (int j=1; j<=n; j++)
            printf("%d ", tmp.park[i][j]);
    }
}
```

Σελ. 5 από 8



```
        printf("\n");
    } printf("\n");
}

void check(state &tmp) {
    int i;
    //print(tmp);
    for (i=1; i<=n; i++) if (tmp.park[r][i]!=0) return;
    fprintf(fout, "%d\n", tmp.moves);
    exit(0);

    //printf("ok with %d moves\n", tmp.moves);
    //if (tmp.moves==4) print(tmp);
}

int main() {
    fin=fopen("getout.in", "r");
    fout=fopen("getout.out", "w");
    state init, tmp, buk;
    int d=1, i, j, t, x, y, k, z;

    fscanf(fin, "%d %d", &n, &r);
    for (i=1; i<=n; i++) for (j=1; j<=n; j++) init.park[i][j] = 0;

    for (i=1; i<=n; i++) {
        fscanf(fin, "%d", &t);
        if (i==r && t!=0) { fprintf(fout, "-1\n"); return 0; }
        for (j=1; j<=t; j++) { fscanf(fin, "%d", &y);
init.park[i][y] = d; init.park[i][y+1] = d++; }
        }
        for (i=1; i<=n; i++) {
            fscanf(fin, "%d", &t);
            for (j=1; j<=t; j++) { fscanf(fin, "%d", &x);
init.park[x][i] = d; init.park[x+1][i] = d++; }
        }
        init.moves = 0;

        s.insert(init);
        q.push(init);
        check(init);

        while (!q.empty()) {
            char used[100]={0};
            tmp = q.front();
            q.pop();

            //print(tmp);

            tmp.moves++;
            buk = tmp;
            for (i=1; i<=n; i++)
```

Σελ. 6 από 8



```
for (j=1; j<=n; j++) {
    if (tmp.park[i][j]) {
        for (k=0; k<4; k++) {
            x = i + dir[k][0]; y = j + dir[k]
[1];
            if (i>=1 && i<=n && j>=1 && j<=n
&& tmp.park[i][j]==tmp.park[x][y] && !used[tmp.park[i][j]]) break;
        }
        if (k!=4) {
            used[tmp.park[i][j]] = 1;
            //printf("found (%d %d) (%d %d): %d
\n", i, j, x, y, tmp.park[i][j]);
            if (i==x) {
                //if (j<y) k = j; else k = y;
                k = j;
                for (z=k-1; z>=1; z--) {
                    if (!tmp.park[i][z]) {
                        tmp.park[i][z] = tm-
p.park[i][z+1];
                        tmp.park[i][z+2] = 0;
                    } else break;
                    if (s.find(tmp)==s.end()) {
                        q.push(tmp); s.insert(tmp); check(tmp);
                    }
                    tmp = buk;
                    for (z=k+2; z<=n; z++) {
                        if (!tmp.park[i][z]) {
                            tmp.park[i][z] = tm-
p.park[i][z-2];
                            tmp.park[i][z-2] = 0;
                        } else break;
                        if (s.find(tmp)==s.end()) {
                            q.push(tmp); s.insert(tmp); check(tmp);
                        }
                    }
                    tmp = buk;
                } else {
                    //if (i<x) k = i; else k = x;
                    k = i;
                    //printf("second\n");
                    //print(tmp);
                    for (z=k-1; z>=1; z--) {
                        if (!tmp.park[z][j]) {
                            //printf("ok\n");
                            tmp.park[z][j] = tm-
p.park[z+1][j];
                            tmp.park[z+2][j] = 0;
                        } else break;
                        if (s.find(tmp)==s.end()) {
                            q.push(tmp); s.insert(tmp); check(tmp);
                        }
                    }
                    tmp = buk;
                    for (z=k+2; z<=n; z++) {
```

Σελ. 7 από 8



```
if (!tmp.park[z][j]) {
    tmp.park[z][j] = tm-
    tmp.park[z-2][j] = 0;
} else break;
if (s.find(tmp)==s.end()) {
    tmp = buk;
}
}
}
}
}
//return 0;
}
fprintf(fout, "-1\n");
return 0;
}
```

