**Эйлер**

int m, pos;

VEC <int> gr[MAXN];

VEC <int> ans;

vector<int>::iterator it;

int circuit[1050];

void find\_circuit(int i)

{

while(gr[i].SZ > 0){

it = min\_element(gr[i].begin(), gr[i].end());

int j = (\*it);

gr[i].erase(it);

gr[j].erase(find(gr[j].begin(),gr[j].end(),i));

find\_circuit(j);

}

circuit[pos++] = i;

}

void in()

{

ifstream cin("fence.in");

cin >> m;

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

int a, b;

cin >> a >> b;

gr[a].PB(b);

gr[b].PB(a);

}

}

void solution()

{

int s = 0, s1 = -1, s2;

for(int i = 1; i <= 500; i++)

{

if(gr[i].SZ > 0 && !s) s = i;

if(gr[i].SZ % 2) s1 == -1 ? s1=i : s2=i;

}

s = (s1 == -1) ? s : min(s1,s2);

ans.PB(s);

find\_circuit(s);

for(int i = m-1; i >= 0; i--)

ans.PB(circuit[i]);

}

void out()

{

ofstream cout("fence.out");

for (int i = 0; i < ans.SZ; i++)

cout << ans[i] << "\n";

}