**Поиск мостов**

const int MAXN = ...;

vector<int> g[MAXN];

bool used[MAXN];

int timer, tin[MAXN], fup[MAXN];

void dfs (int v, int p = -1) {

used[v] = **true**;

tin[v] = fup[v] = timer++;

for (size\_t i=0; i<g[v].size(); ++i) {

int to = g[v][i];

if (to == p) continue;

if (used[to])

fup[v] = min (fup[v], tin[to]);

else {

dfs (to, v);

fup[v] = min (fup[v], fup[to]);

if (fup[to] > tin[v])

IS\_BRIDGE(v,to);

}

}

}

void find\_bridges() {

timer = 0;

for (int i=0; i<n; ++i)

used[i] = **false**;

for (int i=0; i<n; ++i)

if (!used[i])

dfs (i);

}