/\*

PROG: surround

LANG: C++

ID: hayk.sa1

\*/

#include

#include

#include

using namespace std;

#define inf 1000000000

int n, I;

int l[510][510];

vector g[510];

void init()

{

freopen("surround.in", "r", stdin);

freopen("surround.out", "w", stdout);

int p, q, i;

scanf("%d", &n);

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

{

scanf("%d%d", &p, &q);

g[--p].push\_back(--q);

g[q].push\_back(p);

}

for (p = 0; p < n; p++)

for (q = 0; q < n; q++)

scanf("%d", l[p]+q);

}

int t[510];

void dfs(int p)

{

int i;

t[p] = I;

for (i = 0; i < g[p].size(); i++)

if (t[g[p][i]] == -1)

dfs(g[p][i]);

}

void components()

{

int i;

memset(t, -1, sizeof(t));

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

if (t[i] == -1)

{

dfs(i);

I++;

}

}

int d[510][510];

void construct()

{

int i, j;

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

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

d[i][j] = inf;

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

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

if (d[t[i]][t[j]] > l[i][j])

d[t[i]][t[j]] = l[i][j];

}

int solve()

{

int ret=inf, s, i, j;

for (i = 0; i < I; i++)

{

s = 0;

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

s += d[i][j];

if (ret > s)

ret = s;

}

return ret\*2;

}

int main()

{

init();

components();

construct();

printf("%d\n", solve());

return 0;

}