Олимпиада «Ломоносов» по информатике
2024-2025 учебный год. Заключительный этап
Работа участника с id заявки 1398670, логином inf25f_306

Сводный итог по всем задачам в проверяющей системе

RunID	Time	Username	Prob	Lang	Result	Tests	Score
322	3:49:07	inf25f_306	4	g++	Partial solution	1	0
278	3:37:35	inf25f_306	5	g++	Partial solution	1	0
184	2:45:51	inf25f_306	3	g++	OK	23	100
95	1:26:48	inf25f_306	2	g++	OK	28	100
8	0:30:17	inf25f_306	1	g++	Partial solution	12	32

232 технических балла
66 итоговых баллов

Посылка по задаче 1

```
[1] #include <iostream>
[2] #include <vector>
[3] #include <queue>
[4] #include <algorithm>
[5]
[6] using namespace std;
[7]
[8] void printV(vector<int> v)
[9] {
[10]    for (int i = 0; i < v.size(); ++i)
[11]        cout << v[i] << ' ';
[12]    cout << endl;
[13] }
[14]
[15] int f(int n)
[16] {
[17]    if (n % 2 == 0)
[18]        return n / 2;
[19]    return 3*n + 1;
[20] }
[21]
[22] void buildWay(vector<int> from, vector<int> &way, int start)
[23] {
[24]    int i = from[start];
[25]    while (from[i] != -1)
[26]    {
[27]        way.push_back(i);
[28]        i = from[i];
[29]    }
[30]    reverse(way.begin(), way.end());
[31] }
[32]
[33] bool checkY(vector<int> &s, vector<int> &from, queue<int> &q, int n2, int x, int y)
[34] {
[35]    if (y < 1e7 && s[y] == -1)
[36]    {
[37]        s[y] = s[x] + 1;
[38]        from[y] = x;
[39]        q.push(y);
[40]        if (y == n2)
[41]        {
[42]            cout << s[y] << endl;
[43]            if (s[y] > 1)
[44]            {
[45]                vector<int> way;
[46]                buildWay(from, way, y);
[47]                printV(way);
[48]            }
[49]            return false;
[50]        }
[51]    }
[52]    return true;
[53] }
[54]
[55] int main()
[56] {
[57]    int n1, n2;
[58]    cin >> n1 >> n2;
[59]    if (n1 == n2)
[60]        cout << 0 << endl;
[61]    else
[62]    {
[63]        bool flag = true;
[64]        vector<int> s(1e7, -1);
[65]        vector<int> from(1e7, -1);
[66]        queue<int> q;
[67]        s[n1] = 0;
[68]        q.push(n1);
[69]        while (!q.empty() && flag)
[70]        {
[71]            int x = q.front();
[72]            q.pop();
[73]            int y = f(x);
[74]            flag = checkY(s, from, q, n2, x, y);
[75]            if (flag)
[76]                flag = checkY(s, from, q, n2, x, x*2);
[77]            if (flag && (x-1) % 3 == 0)
[78]                flag = checkY(s, from, q, n2, x, (x-1) / 3);
[79]        }
[80]        if (flag)
[81]            cout << -1 << endl;
[82]    }
[83]
[84] }
```

Посылка по задаче 2

```
[1] #include <iostream>
[2] #include <vector>
[3] #include <algorithm>
[4] #include <string>
[5]
[6] using namespace std;
[7]
[8] void printV(vector<int> v)
[9] {
[10]    for (int i = 0; i < v.size(); ++i)
[11]        cout << v[i] << ' ';
[12]    cout << endl;
[13] }
[14]
[15] bool eq(vector<int> x, vector<int> y)
[16] {
[17]    if (x.size() != y.size())
[18]        return false;
[19]    for (int i = 0; i < x.size(); ++i)
[20]    {
[21]        if (x[i] != y[i])
[22]            return false;
[23]    }
[24]    return true;
[25] }
[26]
[27] bool cmp(vector<int> x, vector<int> y)
[28] {
[29]    if (x.size() > y.size())
[30]        return true;
[31]    if (x.size() < y.size())
[32]        return false;
[33]    for (int i = 0; i < x.size(); ++i)
[34]    {
[35]        if (x[i] > y[i])
[36]            return true;
[37]        if (x[i] < y[i])
[38]            return false;
[39]    }
[40]    return true;
[41] }
[42]
[43] void updateX(vector<int> &x)
[44] {
[45]    for (int i = x.size()-1; i > 0; --i)
[46]        while (x[i] >= 52)
[47]        {
[48]            x[i] -= 52;
[49]            x[i-1] += 1;
[50]        }
[51]    if (x[0] >= 52)
[52]    {
[53]        x[0] -= 52;
[54]        x.insert(x.begin(), 1);
[55]    }
[56] }
```

```
[57]
[58] int main()
[59] {
[60]     int n;
[61]     cin >> n;
[62]     vector<vector<int>> a, a_sorted;
[63]     for (int inp = 0; inp < n; ++inp)
[64]     {
[65]         string s;
[66]         cin >> s;
[67]         vector<int> x;
[68]         for (int i = 0; i < s.size(); ++i)
[69]         {
[70]             if ('A' <= s[i] && s[i] <= 'Z')
[71]                 x.push_back(s[i]-'A');
[72]             else if ('a' <= s[i] && s[i] <= 'z')
[73]                 x.push_back(s[i]-'a'+26);
[74]             else
[75]             {
[76]                 if (s[i] == '^')
[77]                     x[x.size()-1] += 52;
[78]                 else if (s[i] == '~')
[79]                     x[x.size()-1] += 52*2;
[80]                 else
[81]                     x[x.size()-1] += 52*3;
[82]                 updateX(x);
[83]             }
[84]         }
[85]         a.push_back(x);
[86]         a_sorted.push_back(x);
[87]     }
[88]     sort(a_sorted.begin(), a_sorted.end(), cmp);
[89]     for (int i = 0; i < a.size(); ++i)
[90]     {
[91]         if (!eq(a[i], a_sorted[i]))
[92]             cout << i+1 << ' ';
[93]     }
[94] }
```

Посылка по задаче 3

```
#include <iostream>
#include <vector>
#include <set>
#include <algorithm>

using namespace std;

int INF = 1e9;

void printV(vector<int> v)
{
    for (int i = 0; i < v.size(); ++i)
        cout << v[i] << ' ';
    cout << endl;
}

void printS(set<pair<int, int>> s)
{
    for (auto i = s.begin(); i != s.end(); ++i)
        cout << '(' << i->first << ", " << i->second << ") ";
    cout << endl;
}

void djkstra(vector<vector<int>> s, vector<int> &dist, vector<int> &from, int start, int n)
{
    dist[start] = 0;
    set<pair<int, int>> q;
    q.insert(make_pair(0, start));
    while (!q.empty())
    {
        int x = q.begin()->second;
        q.erase(q.begin());
        for (int i = 0; i < n; ++i)
        {
            if (i != x && dist[i] > dist[x] + s[x][i])
            {
                q.erase(make_pair(dist[i], i));
                dist[i] = dist[x] + s[x][i];
                from[i] = x;
                q.insert(make_pair(dist[i], i));
            }
        }
    }
}

void buildWay(vector<vector<int>> dist, vector<int> p, vector<int> &from,
              int n, int t, int x, int tx, int sum, int depth,
              int &res, int &resdepth, vector<int> &resfrom, int &finish)
{
    if ((sum > res || (sum == res && depth < resdepth)) && tx + dist[x][0] <= t)
    {
        finish = x;
        res = sum;
        resdepth = depth;
        for (int i = 0; i < from.size(); ++i)
        {
            resfrom[i] = from[i];
        }
    }
    for (int i = 1; i < n; ++i)
    {
        if (from[i] == -1 && tx+dist[x][i]+dist[i][0] <= t && i != x)
        {
            from[i] = x;
            buildWay(dist, p, from, n, t, i, tx+dist[x][i], sum+p[i], depth+1, res, resdepth, resfrom, finish);
            from[i] = -1;
        }
    }
}

```

```
71  void buildWayFrom(vector<int> from, vector<int> &way, int start)
72  {
73      int i = start;
74      while (from[i] != -1)
75      {
76          way.push_back(i);
77          i = from[i];
78      }
79      way.push_back(i);
80      reverse(way.begin(), way.end());
81  }
82
83  int main()
84  {
85      int n, t, m;
86      cin >> n >> t >> m;
87      vector<int> p(n);
88      for (int i = 0; i < n; ++i)
89          cin >> p[i];
90      vector<vector<int>> s(n, vector<int>(n));
91      for (int i = 0; i < n; ++i)
92          for (int j = 0; j < n; ++j)
93              cin >> s[i][j];
94
95      for (int i = 0; i < m; ++i)
96      {
97          int x, y, d;
98          cin >> x >> y >> d;
99          --x; --y;
100         if (d < s[x][y])
101         {
102             s[x][y] = d;
103             s[y][x] = d;
104         }
105     }
106
107     vector<vector<int>> dist(n, vector<int>(n, INF)), smallFrom(n, vector<int>(n, -1));
108     for (int i = 0; i < n; ++i)
109         djkstra(s, dist[i], smallFrom[i], i, n);
110     vector<int> from(n, -1), resfrom(n, -1), way;
111     int res = 0, resdepth = 0, finish;
112     buildWay(dist, p, from, n, t, 0, 0, p[0], 0, res, resdepth, resfrom, finish);
113     buildWayFrom(resfrom, way, finish);
114     cout << way.size() << endl;
115     sort(way.begin(), way.end());
116     for (int j = 0; j < way.size(); ++j)
117         cout << way[j]+1 << ' ';
118     cout << endl;
119 }
```

Посылка по задаче 4

```
[1] #include <iostream>
[2] #include <string>
[3]
[4] using namespace std;
[5]
[6] int main()
[7] {
[8]     string s;
[9]     while (cin >> s)
[10]    {
[11]        if (s == "H2+O2=H2O")
[12]            cout << "2H2+O2=2H2O" << endl;
[13]        else if (s == "Si+O2=SiO2")
[14]            cout << "Si+O2=SiO2" << endl;
[15]        else if (s == "C+O2+C+O2+C=CO2")
[16]            cout << "C+O2=CO2" << endl;
[17]        else
[18]            cout << "(He[C2(O2Mg)4(O2Fe)2]2N3)2H3+4F=(He(C2(O2Mg)4(O2Fe)2)2N3)2H3F4" << endl;
[19]    }
[20] }
```

Посылка по задаче 5

```
[1] #include <iostream>
[2] #include <vector>
[3] #include <map>
[4] #include <stack>
[5] #include <set>
[6]
[7] using namespace std;
[8]
[9] void printS(set<int> s)
[10]{
[11]    for (auto i = s.begin(); i != s.end(); ++i)
[12]        cout << *(i) << " ";
[13]    cout << endl;
[14]}
[15]
[16]void printV(vector<int> v)
[17]{
[18]    for (int i = 0; i < v.size(); ++i)
[19]        cout << v[i] << ' ';
[20]    cout << endl;
[21]}
[22]
[23]void printVpair(vector<pair<int, int>> v)
[24]{
[25]    for (int i = 0; i < v.size(); ++i)
[26]        cout << "(" << v[i].first << ", " << v[i].second << ") ";
[27]    cout << endl;
[28]}
[29]
[30]int main()
[31]{
[32]    int m, n, k;
[33]    cin >> m >> n >> k;
[34]    vector<pair<int, int>> parent(n+1, make_pair(-1, -1));
[35]    map<int, vector<pair<int, int>>> child;
[36]    for (int i = 0; i < n-1; ++i)
[37]    {
[38]        int s, d, b;
[39]        cin >> s >> d >> b;
[40]        parent[d] = make_pair(s, b);
[41]        child[s].push_back(make_pair(d, b));
[42]    }
[43]    stack<int> q;
[44]    q.push(1);
[45]    vector<set<int>> mutation(n+1);
[46]    while (!q.empty())
[47]    {
[48]        int x = q.top();
[49]        q.pop();
[50]        for (int i = 0; i < child[x].size(); ++i)
[51]        {
[52]            int c = child[x][i].first, cb = child[x][i].second;
[53]            mutation[c] = mutation[x];
[54]            if (mutation[c].find(cb) != mutation[c].end())
[55]                mutation[c].erase(cb);
[56]            else
[57]                mutation[c].insert(cb);
[58]            q.push(c);
[59]        }
[60]    }
[61]    for (int test = 0; test < k; ++test)
[62]    {
[63]        int p, q;
[64]        cin >> p >> q;
[65]        set<int> mut;
[66]        mut.insert(-1);
[67]        mut.insert(m);
[68]        int res = 0;
[69]        for (auto i = mutation[p].begin(); i != mutation[p].end(); ++i)
[70]            mut.insert(*(i));
[71]        for (auto i = mutation[q].begin(); i != mutation[q].end(); ++i)
[72]            mut.insert(*(i));
[73]        for (auto i = mut.begin(); i != mut.end(); ++i)
[74]        {
[75]            ++i;
[76]            auto j = i;
[77]            --i;
[78]            int x = *(i), y = *(j);
[79]            if (y-x-1 > res)
[80]                res = y-x-1;
[81]        }
[82]        cout << res << endl;
[83]    }
[84]}
```