



**МОСКОВСКИЙ ГОСУДАРСТВЕННЫЙ УНИВЕРСИТЕТ
имени М.В. ЛОМОНОСОВА**

ОЛИМПИАДНАЯ РАБОТА

Наименование олимпиады школьников: **«Ломоносов»**

Профиль олимпиады: **Информатика**

ФИО участника олимпиады: **Патык Дмитрий Леонидович**

Класс: **11 класс**

Технический балл: **80**

Дата проведения: **17 марта 2022 г.**

Результаты проверки:

Оценка участника строится из 3 частей:

1. оценка за задание - рассчитывается путем запуска тестов и определения правильности работы программы на тестах, до 100 баллов по каждой задаче;
2. дополнительные баллы за полностью правильное решение задания со 2 по 5 - в случае прохождения всех тестов по заданию к оценке прибавляется 55 баллов;
3. нормализация оценки - если полученная из пунктов 1 и 2 сумма баллов превышает 500, то итоговая оценка - 100, если не превышает 500, но превышает 400 - 99 баллов, если не превышает 400 - делится на 3.9 и округляется до целого.

Оценки за задания:

№	1	2	3	4	5
Оценка	100	100	0	56	0

Дополнительный балл: 55

Задание 1. Попытка 1.

```
#include <iostream>
```

```
#include <vector>
```

```
#include <utility>
```

```
#include <algorithm>
```

```
using namespace std;
```

```
int syms[255];
```

```
int foo(pair<string, int> a, pair<string, int> b) {
```

```
    int na = a.first.size(), nb = b.first.size(), n = min(na, nb), d = abs(na - nb);
```

```
    if (na < nb)
```

```
        for (int i = 0; i < d; i++)
```

```
            if (b.first[i] != '0')
```

```
                return 1;
```

```
    if (nb < na)
```

```
        for (int i = 0; i < d; i++)
```

```
            if (a.first[i] != '0')
```

```
                return 0;
```

```
    for (int i = n; i >= 1; i--) {
```

```
        if (syms[a.first[na - i]] < syms[b.first[nb - i]])
```

```
            return 1;
```

```
        if (syms[a.first[na - i]] > syms[b.first[nb - i]])
```

```
    return 0;
}
```

```
return 0;
}
```

```
bool eq(string a, string b) {
    int na = a.size(), nb = b.size(), n = min(na, nb), d = abs(na - nb);
    if (na < nb)
        for (int i = 0; i < d; i++)
            if (b[i] != '0')
                return false;
    if (nb < na)
        for (int i = 0; i < d; i++)
            if (a[i] != '0')
                return false;

    for (int i = n; i >= 1; i--)
        if (a[na - i] != b[nb - i])
            return false;

    return true;
}
```

```
int main() {
```

```
int k, n;

for (int i = 0; i < 10; i++) {
    syms['0' + i] = i;
}

for (int i = 0; i < 26; i++) {
    syms['a' + i] = 10 + i;
}

for (int i = 0; i < 26; i++) {
    syms['A' + i] = 36 + i;
}

cin >> k >> n;

vector<pair<string, int>> a(n);

for (int i = 0; i < n; i++) {
    string x;
    cin >> x;

    a[i] = make_pair(x, i + 1);
}

std::sort(a.begin(), a.end(), foo);
```

```

int i;

bool found = false;

string maxi;

for (i = a.size() - 1; i >= 0; i--) {

    int nn = a[i].first.size();

    found = true;

    for (int j = nn - 1; j >= max(nn - k + 1, 0); j--) {

        if (a[i].first[j] != '0') {

            found = false;

            break;

        }

    }

    if (found) {

        maxi = a[i].first;

        break;

    }

}

if (!found) {

    cout << -1;

    return 0;

}

vector<int> res;

while (i >= 0 && eq(maxi, a[i].first)) {

```

```
    res.push_back(a[i].second);
    i--;
}

sort(res.begin(), res.end());

for (i = 0; i < maxi.size() - 1 && maxi[i] == '0'; i++) {}
for (; i < maxi.size(); i++)
    cout << maxi[i];
cout << endl;

for (i = 0; i < res.size(); i++)
    cout << res[i] << endl;

return 0;
}
```

Задание 2. Попытка 1.

```
#include <iostream>
```

```
#include <vector>
```

```
#include <utility>
```

```
#include <algorithm>
```

```
using namespace std;
```

```
int syms[255];
```

```
int cmpl(char a, char b) {  
    return less<int>()(syms[a], syms[b]);  
}
```

```
int cmpg(char a, char b) {  
    return greater<int>()(syms[a], syms[b]);  
}
```

```
int main() {  
    for (int i = 0; i < 255; i++)  
        syms[i] = -1;  
  
    for (int i = 0; i < 10; i++)  
        syms['0' + i] = i;  
  
    for (int i = 0; i < 26; i++)
```



```
syms['a' + i] = 10 + i;  
for (int i = 0; i < 26; i++)  
    syms['A' + i] = 36 + i;
```

```
vector<char> a;
```

```
int n;
```

```
cin >> n;
```

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

```
    char c;
```

```
    cin >> c;
```

```
    if (syms[c] != -1)
```

```
        a.push_back(c);
```

```
}
```

```
sort(a.begin(), a.end(), cmpl);
```

```
int j;
```

```
for (j = 1; j < a.size() + 1 && j <= 61; j++)
```

```
    if (syms[a[j - 1]] > j)
```

```
        break;
```

```
j--;
```

```
if (j == 0) {
```

```
    cout << -1;
```

```
    return 0;
}

sort(a.begin(), a.end(), cmpg);

string res;

int i = 0;

while (i < a.size() && j >= 1) {
    if (syms[a[i]] > j) {
        i++;
    }
    else {
        res += a[i];
        j--;
        i++;
    }
}

for (i = 0; i < res.size() - 1 && res[i] == '0'; i++) {}
for (; i < res.size(); i++)
    cout << res[i];

return 0;
```


Задание 4. Попытка 1.

```
#include <iostream>
```

```
#include <vector>
```

```
#include <utility>
```

```
#include <algorithm>
```

```
using namespace std;
```

```
const int INF = 1e9;
```

```
class Graph {
```

```
public:
```

```
    int n, m;
```

```
    vector<vector<int>> a;
```

```
    Graph(int n, int m) : n(n), m(m) {
```

```
        a.resize(n);
```

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

```
            a[i].resize(n);
```

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

```
            int x, y;
```

```
            cin >> x >> y;
```

```
            x--, y--;
```

```
            if (x == y)
```

```

        continue;
    if (y < x)
        swap(x, y);
    a[x][y]++;
}
}

bool are_connected(int x, int y) {
    if (y < x)
        swap(x, y);
    return a[x][y] > 0;
}

bool check() {
    vector<bool> flags(n);
    vector<int> s;
    s.push_back(0);

    while (!s.empty()) {
        int x = s.back(); s.pop_back();
        if (flags[x])
            continue;

        for (int i = 0; i < n; i++) {
            if (!flags[i] && are_connected(x, i))

```

```

        s.push_back(i);
    }

    flags[x] = true;
}

for (int i = 0; i < n; i++)
    if (flags[i] == false)
        return true;
return false;
}
};

int mini = INF;
vector<pair<int, int>> answ;
vector<pair<int, int>> tmp;

void foo(Graph &g, int k, int x, int y) {
    if (g.check())
        if (k < mini)
            mini = k, answ = tmp;

    for (int i = x; i < g.n; i++) {
        for (int j = y; j < g.n; j++) {
            if (g.a[i][j] > 0) {

```

```
g.a[i][j]--;
int tmpi = i, tmpj = j;
if (tmpj < tmpi)
    swap(tmpi, tmpj);
tmp.emplace_back(tmpi, tmpj);
foo(g, k + 1, i, j);
tmp.pop_back();
g.a[i][j]++;
}
}
}
}
```

```
int main() {
    int N, M;

    cin >> N >> M;

    Graph g(N, M);

    foo(g, 0, 0, 0);

    sort(answ.begin(), answ.end());

    cout << mini << "\n";
```

```
for (int i = 0; i < mini; i++)
```

```
    cout << answ[i].first + 1 << " " << answ[i].second + 1 << "\n";
```

```
return 0;
```

```
}
```