



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

ОЛИМПИАДНАЯ РАБОТА

Наименование олимпиады школьников: **«Ломоносов»**

Профиль олимпиады: **Информатика**

ФИО участника олимпиады: **Насретдинов Тимур Ринатович**

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 <bits/stdc++.h>
```

```
using namespace std;
```

```
#define ios ios_base::sync_with_stdio(false); cin.tie(0); cout.tie(0)
```

```
#define ll long long
```

```
//#define int long long
```

```
int get_k(char c) {
```

```
    if (c <= '9') return c - '0';
```

```
    if ('a' <= c && c <= 'z') return (c - 'a') + 10;
```

```
    return (c - 'A') + 36;
```

```
}
```

```
char get_c(int k) {
```

```
    if (k <= 9) return char(k + '0');
```

```
    if (10 <= k && k <= 35) return char(k - 10 + 'a');
```

```
    return char(k - 36 + 'A');
```

```
}
```

```
signed main()
```

```
{
```

```
    //cout << get_c(4) << "\n";
```

```

int k, n;

cin >> k >> n;

vector <vector <int>> v(n, vector <int>(61, 0));

vector <int> biggest = {-1};

string s;

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

    cin >> s;

    for (int j = s.length() - 1; j >= 0; --j) {

        v[i][60 - (s.length() - 1 - j)] = get_k(s[j]);

    }

    bool ok = true;

    for (int j = 0; j <= k - 2; ++j) {

        if (v[i][60 - j] != 0) {

            ok = false;

            break;

        }

    }

    if (!ok) continue;

    if (i == 0) biggest = v[i];

    else {

        for (int j = 0; j < 61; ++j) {

            if (v[i][j] > biggest[j]) {

                biggest = v[i];

                break;

            }

        }

    }

}

```

```

    }
}
}
if (biggest[0] == -1) {
    cout << "-1";
    exit(0);
}
bool ok = false;
for (int i = 0; i < 61; ++i) {
    if (biggest[i] != 0) ok = true;
    if (ok) {
        cout << get_c(biggest[i]);
    }
}
if (!ok) cout << "0";
cout << "\n";
for (int i = 0; i < n; ++i) {
    ok = true;
    for (int j = 0; j < 61; ++j) {
        if (v[i][j] != biggest[j]) {
            ok = false;
            break;
        }
    }
}
if (ok) cout << i + 1 << "\n";

```


Задание 1. Попытка 2.

```
#include <iostream>
```

```
#include <bits/stdc++.h>
```

```
using namespace std;
```

```
#define ios ios_base::sync_with_stdio(false); cin.tie(0); cout.tie(0)
```

```
#define ll long long
```

```
//#define int long long
```

```
int get_k(char c) {
```

```
    if (c <= '9') return c - '0';
```

```
    if ('a' <= c && c <= 'z') return (c - 'a') + 10;
```

```
    return (c - 'A') + 36;
```

```
}
```

```
char get_c(int k) {
```

```
    if (k <= 9) return char(k + '0');
```

```
    if (10 <= k && k <= 35) return char(k - 10 + 'a');
```

```
    return char(k - 36 + 'A');
```

```
}
```

```
signed main()
```

```
{
```

```
    ios;
```

```

//cout << get_c(4) << "\n";

int k, n;

cin >> k >> n;

vector <vector <int>> v(n, vector <int>(61, 0));

vector <int> biggest = {-1};

string s;

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

    cin >> s;

    for (int j = s.length() - 1; j >= 0; --j) {

        v[i][60 - (s.length() - 1 - j)] = get_k(s[j]);

    }

    bool ok = true;

    for (int j = 0; j <= k - 2; ++j) {

        if (v[i][60 - j] != 0) {

            ok = false;

            break;

        }

    }

    if (!ok) continue;

    if (i == 0) biggest = v[i];

    else {

        for (int j = 0; j < 61; ++j) {

            if (v[i][j] < biggest[j]) break;

            if (v[i][j] > biggest[j]) {

//                cout << i << " BIGGER:\n";

```



```

//         for (auto t: biggest) cout << t << " ";
//         cout << "\n";
//         for (auto t: v[i]) cout << t << " ";
//         cout << "\n";
        biggest = v[i];

        break;
    }
}
}
}
if (biggest[0] == -1) {
    cout << "-1";
    exit(0);
}
bool ok = false;
for (int i = 0; i < 61; ++i) {
    if (biggest[i] != 0) ok = true;
    if (ok) {
        cout << get_c(biggest[i]);
    }
}
if (!ok) cout << "0";
cout << "\n";
for (int i = 0; i < n; ++i) {

```

```
ok = true;
for (int j = 0; j < 61; ++j) {
    if (v[i][j] != biggest[j]) {
        ok = false;
        break;
    }
}
if (ok) cout << i + 1 << "\n";
}
}
```

Задание 2. Попытка 1.

```
#include <iostream>
```

```
#include <bits/stdc++.h>
```

```
using namespace std;
```

```
#define ios ios_base::sync_with_stdio(false); cin.tie(0); cout.tie(0)
```

```
#define ll long long
```

```
//#define int long long
```

```
int get_k(char c) {
```

```
    if (c <= '9') return c - '0';
```

```
    if ('a' <= c && c <= 'z') return (c - 'a') + 10;
```

```
    if ('A' <= c && c <= 'Z') return (c - 'A') + 36;
```

```
    return -1;
```

```
}
```

```
char get_c(int k) {
```

```
    if (k <= 9) return char(k + '0');
```

```
    if (10 <= k && k <= 35) return char(k - 10 + 'a');
```

```
    return char(k - 36 + 'A');
```

```
}
```

```
vector <int> cnt;
```

```
vector <int> tmp_cnt;
```

```

vector<int> generate(int len) {
    cnt = tmp_cnt;
    int l = 61;
    vector<int> ans(len, 0);
    for (int j = 0; j < len; ++j) {
        while (l >= 0 && (j + 1 > len || cnt[l] == 0)) l--;
        if (l < 0 || j + 1 > len || cnt[l] == 0) return {-1};
        cnt[l]--;
        ans[j] = 1;
    }
    if (ans[0] == 0 && len != 1) return {-1};
    return ans;
}

```

```

signed main()
{
    cnt.resize(62, 0);
    int n;
    char c;
    cin >> n;
    for (int i = 0; i < n; ++i) {
        cin >> c;
        int t = get_k(c);
        if (t != -1) cnt[get_k(c)]++;
    }
}

```

```
}  
tmp_cnt = cnt;  
vector<int> ans = {-1};  
for (int len = min(n, 61); len >= 1; --len) {  
    vector<int> t = generate(len);  
    if (t[0] != -1) {  
        ans = t;  
        break;  
    }  
}  
if (ans[0] == -1) {  
    cout << -1;  
}  
else {  
    for (auto i: ans) cout << get_c(i);  
}  
}
```

Задание 4. Попытка 1.

```
#include <iostream>
```

```
#include <bits/stdc++.h>
```

```
using namespace std;
```

```
#define ios ios_base::sync_with_stdio(false); cin.tie(0); cout.tie(0)
```

```
#define ll long long
```

```
signed main()
```

```
{
```

```
    int n, m, a, b;
```

```
    cin >> n >> m;
```

```
    vector <vector <int>> g(n + 1);
```

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

```
        cin >> a >> b;
```

```
        if (a == b) continue;
```

```
        g[a].push_back(b);
```

```
        g[b].push_back(a);
```

```
    }
```

```
    int best = 2;
```

```
    for (int i = 3; i <= n; ++i) {
```

```
        if (g[i].size() < g[best].size()) {
```

```
            best = i;
```

```
        }
```

```
}  
vector <pair <int, int>> ans;  
for (auto i: g[best]) {  
    if (i > best) ans.push_back({best, i});  
    else ans.push_back({i, best});  
}  
sort(ans.begin(), ans.end());  
cout << ans.size() << "\n";  
for (auto p: ans) {  
    cout << p.first << " " << p.second << "\n";  
}  
}
```