

Олимпиада «Ломоносов» по информатике
2025-2026 учебный год. Заключительный этап
Работа участника с id заявки 1558375, логином inf26f_198

Сводный итог по всем задачам в проверяющей системе

Run ID	Time	User name	Problem	Language	Result	Tests	Score
1064	3:37:39	inf26f_198	7	g++	Partial solution	1	0
967	3:21:47	inf26f_198	6	g++	Partial solution	4	15
612	2:22:31	inf26f_198	5	g++	OK	103	100
537	2:11:08	inf26f_198	4	g++	OK	103	100
510	2:03:47	inf26f_198	3	python3	OK	23	100
279	1:11:44	inf26f_198	2	g++	Partial solution	51	96
84	0:35:14	inf26f_198	1	python3	OK	23	100

511 (пятьсот одиннадцать) технических баллов
73 (семьдесят три) итоговых балла



председатель Жюри кандидат физ.-мат. наук Малышко В. В.



зам. председателя Жюри кандидат физ.-мат. наук Корныхин Е. В.

Посылка по задаче 1

```
[1] def parse(digit):
[2]     res = 0
[3]
[4]     curI = 0
[5]     curX = 0
[6]
[7]     for c in digit:
[8]         if c == 'I':
[9]             res += 1
[10]            curI += 1
[11]        elif c == 'V':
[12]            res = res + (5 - curI)
[13]            res -= curI
[14]
[15]            curI = 0
[16]        elif c == 'X':
[17]            res = res + (10 - curI)
[18]            res -= curI
[19]
[20]            curI = 0
[21]            curX += 10
[22]        elif c == 'N':
[23]            continue
[24]        else:
[25]            res = res + (50 - curX)
[26]            res -= curX
[27]
[28]            curX = 0
[29]
[30]    return res
[31] alphabet = "A B C D E F G H I J K L M N O P Q R S T U V W X Y Z a b c d e f g h i j k l m n o p q r s t u v w x y z".split(' ')
[32]
[33] n = int(input())
[34]
[35] mn = ""
[36] mn_num = -1
[37]
[38] mx = ""
[39] mx_num = -1
[40]
[41] a = []
[42] for i in range(n):
[43]     s = input().split('.')
[44]     cur = ""
[45]     num = 0
[46]
[47]     for digit in s:
[48]         value = parse(digit)
[49]         cur += alphabet[value]
[50]         num = num * 52 + value
[51]
[52]     if mn_num == -1 or num < mn_num:
[53]         mn_num = num
[54]         mn = cur
[55]
[56]     if mx_num == -1 or num > mx_num:
[57]         mx_num = num
[58]         mx = cur
[59]
[60] print(mn)
[61] print(mx)
[62]
[63]
[64]
[65]
[66]
```

Посылка по задаче 2

```
[1] #include <bits/stdc++.h>
[2] using namespace std;
[3]
[4] using ld = double;
[5] using ll = long long;
[6] using ull = unsigned long long;
[7] using pii = pair<ll, ll>;
[8]
[9] inline void fio() {
[10]     cin.tie(0);
[11]     ios_base::sync_with_stdio(0);
[12] }
[13]
[14] const ll MAX = 2e5;
[15] const ll K = 25;
[16] ll lp[MAX], cnt[MAX];
[17] vector<ll> pr;
[18]
[19] inline void sieve(ll n) {
[20]     memset(lp, 0, sizeof(lp));
[21]     for (ll i = 2; i <= n; i++) {
[22]         if (!lp[i]) {
[23]             lp[i] = i;
[24]             pr.push_back(i);
[25]         }
[26]
[27]         for (auto p : pr) {
[28]             if (1ll * i * p > n || lp[i] < p) break;
[29]             lp[i * p] = p;
[30]         }
[31]     }
[32] }
[33]
[34] ll power(ll a, ll n) {
[35]     ll res = 1;
[36]
[37]     while (n > 0) {
[38]         if (n & 1) {
[39]             res *= a;
[40]         }
[41]
[42]         a *= a;
[43]         n >>= 1;
[44]     }
[45]
[46]     return res;
[47] }
[48]
[49] int main() {
[50]     fio();
[51]     sieve(MAX - 1);
[52]     memset(cnt, 0, sizeof(cnt));
[53]
[54]     auto f = [&](ll b, ll len) -> ll {
[55]         return (power(b, len) - 1) / (b - 1);
[56]     };
[57]
[58]     ll n; cin >> n;
[59]     for (ll i = 0; i < n; i++) {
[60]         ll x; cin >> x;
[61]
[62]         if (x < 2) continue;
[63]         if (lp[x] != x) continue;
[64]
[65]         for (ll len = 2; len <= K; len++) {
[66]             ll l = 2;
[67]             ll r = 2;
[68]
[69]             if (f(l, len) > x) break;
[70]             while (f(r, len) < x) {
[71]                 r <<= 1;
[72]             }
[73]
[74]             // cout << "len : " << len << ' ' << "l, r : " << l << ' ' << r << " -
```

```

[75] > " << f(r, len) << '\n';
[76]     ll base = -1;
[77]
[78]     while (l <= r) {
[79]         ll mid = (l + r) / 2;
[80]         ll val = f(mid, len);
[81]
[82]         if (val == x) {
[83]             base = mid;
[84]             break;
[85]         }
[86]
[87]         if (val < x) {
[88]             l = mid + 1;
[89]         } else {
[90]             r = mid - 1;
[91]         }
[92]     }
[93]
[94]     if (base != -1) {
[95]         cnt[x]++;
[96]         break;
[97]     }
[98] }
[99] }
[100]
[101] ll res = -1;
[102]
[103] for (ll i = 2; i < MAX; i++) {
[104]     if (cnt[i] == 0) continue;
[105]     if (res == -1) {
[106]         res = i;
[107]         continue;
[108]     }
[109]
[110]     if (cnt[res] <= cnt[i]) {
[111]         res = i;
[112]     }
[113] }
[114]
[115] cout << (res == -1 ? 0 : res) << '\n';
[116]
[117]     return 0;
}

```

Посылка по задаче 3

```
[1] def length(n, layer):
[2]     return max(1, 4 * (n - 1) - 8 * (layer - 1))
[3]
[4] n = int(input())
[5]
[6] j = int(input())
[7] i = int(input())
[8]
[9] layer = min(i if i <= (n + 1)//2 else n - i + 1, j if j <= (n + 1)//2 else n - j + 1)
[10]
[11] s = 0 if layer == 1 else 4 * (n - 1) * (layer - 1) - 4 * (layer - 2) * (layer - 1)
[12]
[13] if j == layer:
[14]     print(s + i - layer)
[15]     exit(0)
[16]
[17] s = s + n - 2 * (layer - 1) - 1
[18]
[19] if i == layer + n - 2 * (layer - 1) - 1:
[20]     print(s + j - layer)
[21]     exit(0)
[22]
[23] s = s + n - 2 * (layer - 1) - 1
[24]
[25] if j == layer + n - 2 * (layer - 1) - 1:
[26]     print(s + layer + n - 2 * (layer - 1) - 1 - i)
[27]     exit(0)
[28]
[29] s = s + n - 2 * (layer - 1) - 1
[30]
[31] print(s + layer + n - 2 * (layer - 1) - 1 - j)
```

Посылка по задаче 4

```
[1] #include <bits/stdc++.h>
[2] using namespace std;
[3]
[4] inline void fio() {
[5]     cin.tie(0);
[6]     ios_base::sync_with_stdio(0);
[7] }
[8]
[9] int binSearch(int L, int R, vector<int>& a, int value) {
[10]     int l = L, r = R;
[11]
[12]     while (l <= r) {
[13]         int mid = (l+r)/2;
[14]
[15]         if (a[mid] <= value) {
[16]             l = mid + 1;
[17]         } else {
[18]             r = mid - 1;
[19]         }
[20]     }
[21]
[22]     return l;
[23] }
[24]
[25] int f(vector<int>& a) {
[26]     int n = a.size();
[27]
[28]     vector<int> dp(n+1, INT_MAX); dp[0] = INT_MIN;
[29]     vector<int> prev(n, -1);
[30]     vector<int> pos(n+1, -1);
[31]
[32]     int res = 0;
[33]     for (int i = 0; i < n; i++) {
[34]         int idx = binSearch(0, n+1, dp, a[i]);
[35]         res = max(res, idx);
[36]
[37]         dp[idx] = a[i];
[38]         pos[idx] = i;
[39]         if (idx > 1) {
[40]             prev[i] = pos[idx - 1];
[41]         }
[42]     }
[43]
[44]     vector<int> lis;
[45]     int k = pos[res];
[46]     while (k != -1) {
[47]         lis.push_back(a[k]);
[48]         k = prev[k];
[49]     }
[50]
[51]     return lis.size();
[52] }
[53]
[54] int main() {
[55]     fio();
[56]
[57]     int n; cin >> n;
[58]     vector<int> a(n);
[59]
[60]     for (auto& c : a) cin >> c;
[61]     reverse(a.begin(), a.end());
[62]
[63]     cout << n - f(a) << '\n';
[64]
[65]     return 0;
[66] }
```

Посылка по задаче 5

```
[1] #include <bits/stdc++.h>
[2] using namespace std;
[3]
[4] using ld = double;
[5] using ll = long long;
[6] using ull = unsigned long long;
[7] using pii = pair<ll, ll>;
[8]
[9] inline void fio() {
[10]     cin.tie(0);
[11]     ios_base::sync_with_stdio(0);
[12] }
[13]
[14] vector<vector<int>> g, gt;
[15] vector<int> used;
[16]
[17] void dfs(int v, vector<int>& ts) {
[18]     used[v] = true;
[19]
[20]     for (auto u : g[v]) {
[21]         if (!used[u]) {
[22]             dfs(u, ts);
[23]         }
[24]     }
[25]
[26]     ts.push_back(v);
[27] }
[28]
[29] int main() {
[30]     fio();
[31]
[32]     int n; cin >> n;
[33]     vector<int> x(n), y(n), h(n);
[34]
[35]     for (int i = 0; i < n; i++) {
[36]         cin >> x[i] >> y[i] >> h[i];
[37]     }
[38]
[39]     g.assign(n, {});
[40]     gt.assign(n, {});
[41]     used.assign(n, 0);
[42]
[43]     for (int i = 0; i < n; i++) {
[44]         for (int j = 0; j < n; j++) {
[45]             if (i == j) continue;
[46]
[47]             if (x[j] > x[i] && y[j] > y[i] && h[j] < h[i]) {
[48]                 g[i].push_back(j);
[49]                 gt[j].push_back(i);
[50]             }
[51]         }
[52]     }
[53]
[54]     vector<int> ts;
[55]     for (int i = 0; i < n; i++) {
[56]         if (!used[i]) {
[57]             dfs(i, ts);
[58]         }
[59]     }
[60]
[61]     reverse(ts.begin(), ts.end());
[62]     vector<int> dp(n, 1);
[63]
[64]     int res = 1;
[65]
[66]     for (auto v : ts) {
[67]         for (auto u : gt[v]) {
[68]             dp[v] = max(dp[v], dp[u] + 1);
[69]         }
[70]
[71]         res = max(res, dp[v]);
[72]     }
[73]
[74]     cout << res << '\n';
[75]
[76]     return 0;
[77] }
```

Посылка по задаче 6

```
[1] #include <bits/stdc++.h>
[2] using namespace std;
[3]
[4] using ld = double;
[5] using ll = long long;
[6] using ull = unsigned long long;
[7] using pii = pair<ll, ll>;
[8]
[9] inline void fio() {
[10]     cin.tie(0);
[11]     ios_base::sync_with_stdio(0);
[12] }
[13]
[14] vector<vector<int>> field;
[15] set<vector<vector<int>>> res;
[16] int cnt = 0;
[17]
[18] bool full(int n, int m) {
[19]     for (int i = 0; i < n; i++) {
[20]         for (int j = 0; j < m; j++) {
[21]             if (field[i][j] == 0) {
[22]                 return false;
[23]             }
[24]         }
[25]     }
[26]
[27]     return true;
[28] }
[29]
[30] void rec(int n, int m, int a, int b, int c, int d) {
[31]     if (full(n, m)) {
[32]         res.insert(field);
[33]         cnt++;
[34]
[35]         return;
[36]     }
[37]
[38]     if (a != 0) {
[39]         for (int i = 0; i < n; i++) {
[40]             for (int j = 0; j < m; j++) {
[41]                 if (field[i][j] != 0) continue;
[42]
[43]                 field[i][j] = 1;
[44]                 rec(n, m, a - 1, b, c, d);
[45]                 field[i][j] = 0;
[46]             }
[47]         }
[48]     }
[49]
[50]     if (b != 0) {
[51]         for (int i = 0; i < n; i++) {
[52]             for (int j = 0; j < m; j++) {
[53]                 if (field[i][j] != 0) continue;
[54]
[55]                 if (j + 1 < m && field[i][j + 1] == 0) {
[56]                     field[i][j] = field[i][j + 1] = 2;
[57]                     rec(n, m, a, b - 1, c, d);
[58]                     field[i][j] = field[i][j + 1] = 0;
[59]                 }
[60]
[61]                 if (i + 1 < n && field[i + 1][j] == 0) {
[62]                     field[i + 1][j] = field[i + 1][j] = 2;
[63]                     rec(n, m, a, b - 1, c, d);
[64]                     field[i + 1][j] = field[i + 1][j] = 0;
[65]                 }
[66]             }
[67]         }
[68]     }
[69]
[70]     if (c != 0) {
[71]         for (int i = 0; i < n; i++) {
[72]             for (int j = 0; j < m; j++) {
[73]                 if (field[i][j] != 0) continue;
[74]
```

```

[75]         if (j + 2 < m && field[i][j + 1] == 0 && field[i][j + 2] == 0) {
[76]             field[i][j] = field[i][j + 1] = field[i][j + 2] = 3;
[77]             rec(n, m, a, b, c - 1, d);
[78]             field[i][j] = field[i][j + 1] = field[i][j + 2] = 0;
[79]         }
[80]
[81]         if (i + 2 < n && field[i + 1][j] == 0 && field[i + 2][j] == 0) {
[82]             field[i][j] = field[i + 1][j] = field[i + 2][j] = 3;
[83]             rec(n, m, a, b, c - 1, d);
[84]             field[i][j] = field[i + 1][j] = field[i + 2][j] = 0;
[85]         }
[86]     }
[87] }
[88] }
[89]
[90] if (d != 0) {
[91]     for (int i = 0; i < n; i++) {
[92]         for (int j = 0; j < m; j++) {
[93]             if (field[i][j] != 0) continue;
[94]
[95]             if (j + 3 < m && field[i][j + 1] == 0 && field[i][j + 2] == 0 && field[i]
[96] [j + 3] == 0) {
[97]                 field[i][j] = field[i][j + 1] = field[i][j + 2] = field[i][j + 3] = 4;
[98]                 rec(n, m, a, b, c, d - 1);
[99]                 field[i][j] = field[i][j + 1] = field[i][j + 2] = field[i][j + 3] = 0;
[100]             }
[101]
[102]             if (i + 3 < n && field[i + 1][j] == 0 && field[i + 2][j] == 0 && field[i + 3]
[103] [j] == 0) {
[104]                 field[i][j] = field[i + 1][j] = field[i + 2][j] = field[i + 3][j] = 4;
[105]                 rec(n, m, a, b, c, d - 1);
[106]                 field[i][j] = field[i + 1][j] = field[i + 2][j] = field[i + 3][j] = 0;
[107]             }
[108]         }
[109]     }
[110] }
[111] }
[112]
[113] int main() {
[114]     fio();
[115]
[116]     int a, b, c, d; cin >> a >> b >> c >> d;
[117]     int n, m; cin >> n >> m;
[118]
[119]     field.assign(n, vector<int>(m));
[120]
[121]     for (int i = 0; i < n; i++) {
[122]         for (int j = 0; j < m; j++) {
[123]             char t; cin >> t;
[124]
[125]             field[i][j] = -(t != '#');
[126]         }
[127]     }
[128]
[129]     rec(n, m, a, b, c, d);
[130]     cout << res.size() << '\n';
[131]
[131]     return 0;
}

```

Посылка по задаче 7

```
[1] #include <bits/stdc++.h>
[2] using namespace std;
[3]
[4] inline void fio() {
[5]     cin.tie(0);
[6]     ios_base::sync_with_stdio(0);
[7] }
[8]
[9] random_device rd;
[10] mt19937 gen(rd());
[11]
[12] int rand(int l, int r) {
[13]     return uniform_int_distribution<int>(l, r)(gen);
[14] }
[15]
[16] int main() {
[17]     fio();
[18]
[19]     cout << "34339.0462" << '\n';
[20]
[21]
[22]     return 0;
[23] }
```