

Олимпиада «Ломоносов» по информатике  
2023-2024 учебный год. Заключительный тур  
Работа участника с id заявки 1195825, логином inf24f\_109

Сводный итог по всем задачам в проверяющей системе

Run ID	Time	User name	Problem	Language	Result	Tests	Score
21	0:22:52	inf24f_109	1	python3	OK	28	100
57	1:03:27	inf24f_109	2	python3	OK	28	100
86	1:38:39	inf24f_109	3	python3	OK	28	100
144	2:14:35	inf24f_109	5	g++	Partial solution	20	90
218	2:57:01	inf24f_109	4	clang++	Partial solution	1	0
390 технических баллов							
78 итоговых баллов							

## Посылка по задаче 1

```
[1] class Ntri:
[2]     trib = [0, 0, 1]
[3]     while trib[-1] <= (1 << 24):
[4]         trib.append(trib[-1] + trib[-2] + trib[-3])
[5]
[6]     def __init__(self, n):
[7]         self.digits = []
[8]         self.pelmen = 0
[9]         for i in range(len(self.trib) - 1, 2, -1):
[10]             if n >= self.trib[i]:
[11]                 self.digits.append(1)
[12]                 n -= self.trib[i]
[13]                 self.pelmen ^= 1
[14]             else:
[15]                 self.digits.append(0)
[16]
[17]     def __str__(self):
[18]         res = ''
[19]         fl = True
[20]         for d in self.digits:
[21]             if fl and not d:
[22]                 continue
[23]             fl = False
[24]             res += str(d)
[25]         return res
[26]
[27]
[28] print(sum(Ntri(int(input())).pelmen for _ in range(int(input()))))
```

## Посылка по задаче 2

```
[1] ind = {'B': 0, 'C': 1, 'D': 2, 'G': 3, 'O': 4, 'R': 5, 'V': 6, 'W': 7, 'Y': 8}
[2] # lett = ['W', 'R', 'O', 'Y', 'G', 'C', 'B', 'V', 'D']
[3] lett = ['B', 'C', 'D', 'G', 'O', 'R', 'V', 'W', 'Y']
[4] # print(sorted(lett))
[5] area = [[0 for _ in range(700)] for _ in range(9)]
[6]
[7] T = input()
[8] # if len(T) == 1:
[9] #     print(T, '1.0', sep='\n')
[10] # else:
[11] st = [[0, -1]]
[12] for c in T:
[13]     while st[-1][1] == 0:
[14]         st.pop()
[15]         st[-1][1] -= 1
[16]         if c == 'Q':
[17]             st.append([st[-1][0] + 1, 4])
[18]         else:
[19]             area[ind[c]][st[-1][0] << 1] += 1
[20]
[21] ans = ['' for _ in range(9)]
[22] best = 0
[23] for i in range(len(area)):
[24]     for j in range(699, 0, -1):
[25]         area[i][j - 1] += area[i][j] >> 1
[26]         area[i][j] &= 1
[27]     ans[i] = (''.join(map(str, area[i])).rstrip('0')).ljust(2, '0')
[28]     ans[i] = ans[i][0] + '.' + ans[i][1:]
[29]     if ans[i] >= ans[best]:
[30]         best = i
[31] print(lett[best], ans[best], sep='\n')
```

---

### Посылка по задаче 3

```
[1] val = {'>': 1, '>!': 2, '>!!': 3, '>!!!': 4, '>?': 5, '<': 6, '<!': 7, '<!!': 8, '<!!!': 9, '<?': 10}
[2] dig = ['()', '>', '>!', '>!!', '>!!!', '>?', '<', '<!', '<!!', '<!!!', '<?']
[3]
[4] def n(s):
[5]     if s[0] == '(':
[6]         return 0
[7]     d = []
[8]     for c in s:
[9]         if c in '<>':
[10]            d.append(c)
[11]        else:
[12]            d[-1] += c
[13]     res = 0
[14]     pow10 = 1
[15]     for x in d:
[16]         res += val[x] * pow10
[17]         pow10 *= 10
[18]     return res
[19]
[20]
[21] def g(x):
[22]     if x == 0:
[23]         return dig[0]
[24]     res = []
[25]     while x:
[26]         if x % 10 == 0:
[27]             res.append(dig[10])
[28]             x -= 10
[29]         else:
[30]             res.append(dig[x % 10])
[31]             x //= 10
[32]     return ''.join(res)
[33]
[34]
[35] # while True:
[36] #     print(g(int(input())))
[37]
[38] imin, imax = 0, 0
[39] vmin, vmax = float('inf'), -float('inf')
[40] for i in range(int(input())):
[41]     v = n(input())
[42]     if v < vmin:
[43]         vmin = v
[44]         imin = i
[45]     if v > vmax:
[46]         vmax = v
[47]         imax = i
[48]
[49]
[50] if imin == imax:
[51]     print('>\n>!')
[52] else:
[53]     imin, imax = sorted((imin + 1, imax + 1))
[54]     print(g(imin), g(imax), sep='\n')
```

#### Посылка по задаче 4

```
[1] #include <iostream>
[2] #include <string>
[3] #include <vector>
[4]
[5] using namespace std;
[6]
[7] int ind(char c) { return c - 33; }
[8]
[9] int main() {
[10]     string t, goal; cin >> t >> goal;
[11]     int s = 0, e = 1, bs = 0, be = 100000;
[12]     vector<int> cur(94, 0), need(94, 0);
[13]     for (char c : goal) { ++need[ind(c)]; }
[14]     while (true) {
[15]         bool pelmen = true;
[16]         for (int i = 0; i < 94; ++i) {
[17]             if (cur[i] < need[i]) { pelmen = false; }
[18]         }
[19]         if (pelmen) {
[20]             if (e - s < be - bs) {be = e; bs = s;}
[21]             --cur[ind(t[s])]; ++s;
[22]         }
[23]         else {
[24]             if (e == t.size()) { break; }
[25]             ++cur[ind(t[e])]; ++e;
[26]         }
[27]     }
[28]     // cout << bs << ' ' << be << endl;
[29]     if (be < 100000) {
[30]         string ans;
[31]         for (int i = bs; i < be; ++i) {
[32]             ans.push_back(t[i]);
[33]         }
[34]         cout << ans << endl;
[35]     }
[36]     else {
[37]         cout << endl;
[38]     }
[39]     return 0;
[40] }
```

## Посылка по задаче 5

```
[1] #include <iostream>
[2] #include <string>
[3] #include <vector>
[4]
[5] using namespace std;
[6]
[7] int R, C;
[8]
[9] struct state {
[10]     int r1, r2, r3;
[11]     state() { r1 = 0; r2 = 0; r3 = 0; }
[12]     state(int x, int y, int z) { r1 = x; r2 = y; r3 = z; }
[13]     state(int x0) { r1 = x0 % R; r2 = (x0 / R) % R; r3 = x0 / (R * R); }
[14]     vector<int> get() {
[15]         vector<int> res = {r1};
[16]         if (r2 != r1) { res.push_back(r2); }
[17]         if ((r3 != r1) && (r3 != r2)) { res.push_back(r3); }
[18]         return res;
[19]     }
[20]     int ind() { return r1 + r2 * R + r3 * R * R; }
[21]     vector<state> poss() {
[22]         vector<int> poss1 = {r1}, poss2 = {r2}, poss3 = {r3};
[23]         if (r1 > 0) { poss1.push_back(r1 - 1); }
[24]         if (r2 > 0) { poss1.push_back(r2 - 1); }
[25]         if (r3 > 0) { poss1.push_back(r3 - 1); }
[26]         if (r1 < R - 1) { poss1.push_back(r1 + 1); }
[27]         if (r2 < R - 1) { poss1.push_back(r2 + 1); }
[28]         if (r3 < R - 1) { poss1.push_back(r3 + 1); }
[29]         vector<state> res;
[30]         for (int x1 : poss1) { for (int x2 : poss2) { for (int x3 : poss3) { res.push_back(state(x1, x2, x3)); }}}
[31]         return res;
[32]     }
[33] };
[34]
[35] int main() {
[36]     int res = 0;
[37]     cin >> R >> C;
[38]     state start; cin >> start.r1 >> start.r2 >> start.r3;
[39]     vector<vector<int>> f(C, vector<int>(R)), dp(C, vector<int>(R * R * R, 0));
[40]     for (int r : start.get())
[41]         dp[0][start.ind()] += f[0][r];
[42]     res = dp[0][start.ind()];
[43]     for (int i = 0; i < R; ++i) { for (int j = 0; j < C; ++j) {cin >> f[j][i];} }
[44]     for (int i = 1; i < C; ++i) {
[45]         for (int j = 0; j < R * R * R; ++j) {
[46]             state cur = state(j);
[47]             for (state st : cur.poss()) {
[48]                 dp[i][j] = max(dp[i][j], dp[i - 1][st.ind()]);
[49]             }
[50]             for (int r : cur.get()) {
[51]                 dp[i][j] += f[i][r];
[52]             }
[53]             res = max(res, dp[i][j]);
[54]         }
[55]     }
[56]     cout << res + 2;
[57]     return 0;
[58] }
```