Олимпиада «Ломоносов» по информатике
2023-2024 учебный год. Заключительный тур
Работа участника с id заявки 1243404, логином inf24f_211

Сводный итог по всем задачам в проверяющей системе

Run ID Time User name Problem Language Result Tests Score
120 0:52:25 inf24f_211 1 g++ OK 28 100
379 2:30:01 inf24f_211 2 g++ OK 28 100
231 1:33:08 inf24f_211 3 pypy3 OK 28 100
71 0:32:31 inf24f_211 4 g++ OK 21 100
398 2:37:14 inf24f_211 5 g++ OK 22 100
681 3:58:20 inf24f_211 6 g++ OK 11 100
600 технических баллов
100 итоговых баллов

Посылка по задаче 1

```
[1]  #include <bits/stdc++.h>
[2]
[3]  using namespace std;
[4]
[5]  #define int long long
[6]
[7]  int32_t main() {
[8]      ios_base::sync_with_stdio(false);
[9]      cin.tie(nullptr);
[10]     vector<int> nums = {0, 0, 0, 1};
[11]     while (nums.back() < (int) 1e9) {
[12]         int sz = (int) nums.size();
[13]         nums.push_back(nums[sz - 1] + nums[sz - 2] + nums[sz - 3] + nums[sz- 4]);
[14]     }
[15]     map<int, string> mp;
[16]     mp[0] = "0";
[17]     mp[1] = "1";
[18]     mp[2] = "10";
[19]     mp[3] = "11";
[20]     auto f = [&](int x) {
[21]         string res;
[22]         for (int i = (int) nums.size() - 1; i >= 3; i--) {
[23]             res += mp[x / nums[i]];
[24]             x %= nums[i];
[25]         }
[26]         int c = count(res.begin(), res.end(), '1');
[27]         return c % 2 == 0;
[28]     };
[29]     /*
[30]     f(0);
[31]     f(3);
[32]     f(732);*/
[33]     int n;
[34]     cin >> n;
[35]     int ans = 0;
[36]     while (n--) {
[37]         int x;
[38]         cin >> x;
[39]         ans += f(x);
[40]     }
[41]     cout << ans;
[42]     return 0;
[43] }
```

Посылка по задаче 2

```
[1]  #include <bits/stdc++.h>
[2]
[3]  using namespace std;
[4]
[5]  string s;
[6]
[7]  // #define double long double
[8]
[9]  map<char, long double> ans;
[10] map<char, vector<int>> mp;
[11]
[12] void f(int& i, long double border, int dep) {
[13]     if (i == (int) s.size()) {
[14]         return;
[15]     }
[16]     if (s[i] != 'Q') {
[17]         ans[s[i]] += border * border * border;
[18]         mp[s[i]][dep]++;
[19]         i++;
[20]     } else {
[21]         i++;
[22]         for (int it = 0; it < 8; it++) {
[23]             f(i, border / 2, dep + 1);
[24]         }
[25]     }
[26] }
[27]
[28] int32_t main() {
[29]     ios_base::sync_with_stdio(false);
[30]     cin.tie(nullptr);
[31]     cin >> s;
[32]     int i = 0;
[33]     const string alp = "WROYGCBVD";
[34]     for (char ch : alp) {
[35]         mp[ch].resize(5000);
[36]     }
[37]     f(i, 1.0, 0);
[38]
[39]     for (char ch : alp) {
[40]         for (int j = 5000 - 1; j >= 1; j--) {
[41]             if (mp[ch][j] >= 8) {
[42]                 mp[ch][j - 1] += mp[ch][j] / 8;
[43]                 mp[ch][j] %= 8;
[44]             }
[45]         }
[46]     }
[47]     vector<int> mx(5000);
[48]     char bst = '!';
[49]     for (auto [x, y] : mp) {
[50]         if (y > mx) {
[51]             mx = y;
[52]             bst = x;
[53]         }
[54]     }
[55]     cout << bst << '\n';
[56]     string ans;
[57]     ans += to_string(mx[0]);
[58]     ans += '.';
[59]     for (int j = 1; j < 5000; j++) {
[60]         bitset<3> b(mx[j]);
[61]         ans += b.to_string();
[62]     }
[63]     while (ans.back() == '0') {
[64]         ans.pop_back();
[65]     }
[66]     if (ans.back() == '.') {
[67]         ans.push_back('0');
[68]     }
[69]     cout << ans << '\n';
[70]
[71]     return 0;
[72] }
```

Посылка по задаче 3

```
[1]  n = int(input())
[2]  val = dict()
[3]  val['i'] = 1
[4]  val['i('] = 2
[5]  val['i(('] = 3
[6]  val['I'] = 4
[7]  val['I('] = 5
[8]  val['I(('] = 6
[9]  val['J'] = 7
[10] val['J)'] = 8
[11] val['J))'] = 9
[12] val['j'] = 10
[13] val['j)'] = 11
[14] val['j))'] = 12
[15] val['[]'] = 0
[16] nums = []
[17] for i in range(n):
[18]     s = input()
[19]     j = len(s) - 1
[20]     a = []
[21]     while j >= 0:
[22]         for (x, y) in val.items():
[23]             if j - len(x) + 1 >= 0 and s[j - len(x) + 1:j + 1] == x:
[24]                 a.append(y)
[25]                 j -= len(x)
[26]                 break
[27]     a = a[::-1]
[28]     p = 1
[29]     s = 0
[30]     for i in a:
[31]         s += p * i
[32]         p *= 12
[33]     nums.append(s)
[34]
[35] i = 0
[36] j = 1
[37] abb = abs(nums[i] - nums[j])
[38] for ii in range(n):
[39]     for jj in range(ii + 1, n):
[40]         ab = abs(nums[ii] - nums[jj])
[41]         if ab > abb or (ab == abb and ii + jj > i + j):
[42]             i = ii
[43]             j = jj
[44]             abb = ab
[45] # print(nums)
[46] # print(i)
[47] # print(j)
[48]
[49] i += 1
[50] j += 1
[51]
[52] inv = dict()
[53] for (x, y) in val.items():
[54]     inv[y] = x
[55] # print(inv)
[56] def f(x: int):
[57]     if x == 0:
[58]         return "[]"
[59]     res = ""
[60]     while x > 0:
[61]         ost = x % 12
[62]         if ost == 0:
[63]             ost = 12
[64]         res += inv[ost]
[65]         x = (x - 1) // 12
[66]     return res
[67]
[68] print(f(i))
[69] print(f(j))
[70]
[71] # print(f(0))
[72] # print(f(12))
[73] # print(f(2024))
```

Посылка по задаче 4

```
[1]   #include <bits/stdc++.h>
[2]
[3]   using namespace std;
[4]
[5]   int32_t main() {
[6]       ios_base::sync_with_stdio(false);
[7]       cin.tie(nullptr);
[8]       int r, c;
[9]       cin >> r >> c;
[10]      int r1, r2, r3;
[11]      cin >> r1 >> r2 >> r3;
[12]      vector<vector<int>> pole(r, vector<int> (c));
[13]      for (int i = 0; i < r; i++) {
[14]          for (int j = 0; j < c; j++) {
[15]              cin >> pole[i][j];
[16]          }
[17]      }
[18]      const int inf = 1e9;
[19]      vector<vector<vector<int>>> dp(r, vector<vector<int>> (r, vector<int> (r, -inf)));
[20]      auto get = [&](int i, int x, int y, int z) {
[21]          if (x ==  y && x == z) {
[22]              return pole[x][i];
[23]          }
[24]          if (x == y && x != z) {
[25]              return pole[x][i] + pole[z][i];
[26]          }
[27]          if (x == z && x != y) {
[28]              return pole[x][i] + pole[y][i];
[29]          }
[30]          if (z == y && z != x) {
[31]              return pole[z][i] + pole[x][i];
[32]          }
[33]          return pole[x][i] + pole[y][i] + pole[z][i];
[34]      };
[35]      const vector<int> var = {-1, 0, 1};
[36]      dp[r1][r2][r3] = get(0, r1, r2, r3);
[37]      for (int it = 0; it < c - 1; it++) {
[38]          vector<vector<vector<int>>> ndp(r, vector<vector<int>> (r, vector<int> (r, -inf)));
[39]          for (int i = 0; i < r; i++) {
[40]              for (int j = 0; j < r; j++) {
[41]                  for (int k = 0; k < r; k++) {
[42]                      for (int ii : var) {
[43]                          int ni = i + ii;
[44]                          if (ni < 0 || ni >= r) {
[45]                              continue;
[46]                          }
[47]                          for (int jj : var) {
[48]                              int nj = j + jj;
[49]                              if (nj < 0 || nj >= r) {
[50]                                  continue;
[51]                              }
[52]                              for (int kk : var) {
[53]                                  int nk = k + kk;
[54]                                  if (nk < 0 || nk >= r) {
[55]                                      continue;
[56]                                  }
[57]                                  ndp[ni][nj][nk] = max(ndp[ni][nj][nk], dp[i][j]
[58] [k] + get(it + 1, ni, nj, nk));
[59]                              }
[60]                          }
[61]                      }
[62]                  }
[63]              }
[64]          }
[65]          dp.swap(ndp);
[66]      }
[67]      int bst = -inf;
[68]      for (int i = 0; i < r; i++) {
[69]          for (int j = 0; j < r; j++) {
[70]              for (int k = 0; k < r; k++) {
[71]                  bst = max(bst, dp[i][j][k]);
[72]              }
[73]          }
[74]      }
[75]      cout << bst;
[76]      return 0;
   }
```

Посылка по задаче 5

```
[1]  #include <bits/stdc++.h>
[2]
[3]  using namespace std;
[4]
[5]  int32_t main() {
[6]      ios_base::sync_with_stdio(false);
[7]      cin.tie(nullptr);
[8]      string s;
[9]      cin >> s;
[10]     const int sz = 126 - 33 + 1;
[11]     vector<int> need(sz);
[12]     string t;
[13]     cin >> t;
[14]     for (char ch : t) {
[15]         need[ch - 33]++;
[16]     }
[17]     int st = -1;
[18]     const int inf = 1e9;
[19]     int len = inf;
[20]     vector<vector<int>> p(sz, vector<int> (s.size() + 1));
[21]     for (int i = 0; i < (int) s.size(); i++) {
[22]         for (int j = 0; j < sz; j++) {
[23]             p[j][i + 1] = p[j][i] + (s[i] - 33 == j);
[24]         }
[25]     }
[26]     vector<int> have(sz);
[27]     auto del = [&](int i) {
[28]         have[s[i] - 33]--;
[29]     };
[30]     auto add = [&](int i) {
[31]         have[s[i] - 33]++;
[32]     };
[33]     auto ok = [&]() {
[34]         bool fl = true;
[35]         for (int i = 0; i < sz; i++) {
[36]             fl &= (have[i] >= need[i]);
[37]         }
[38]         return fl;
[39]     };
[40]     int r = -1;
[41]     for (int i = 0; i < s.size(); i++) {
[42]         if (i > 0) {
[43]             del(i - 1);
[44]         }
[45]         while (!ok() && r + 1 < (int) s.size()) {
[46]             r++;
[47]             add(r);
[48]         }
[49]         if (ok()) {
[50]             if (r - i + 1 < len) {
[51]                 len = r - i + 1;
[52]                 st = i;
[53]             }
[54]         }
[55]     }
[56]
[57]     if (len == inf) {
[58]         cout << string();
[59]     } else {
[60]         cout << s.substr(st, len);
[61]     }
[62]     return 0;
[63] }
```

Посылка по задаче 6

```
[1]  #include <bits/stdc++.h>
[2]
[3]  using namespace std;
[4]
[5]  int32_t main() {
[6]      freopen("input.txt", "r", stdin);
[7]      string s;
[8]      map<string, vector<vector<bool>>> tmp;
[9]      while (getline(cin, s)) {
[10]         int ind = (int) (find(s.begin(), s.end(), ';') - s.begin());
[11]         string name = s.substr(0, ind);
[12]         vector<bool> results;
[13]         while (ind < (int) s.size()) {
[14]             assert(s[ind] == ';');
[15]             ind++;
[16]             if (ind + 1 < (int) s.size() && s[ind] == 'O' && s[ind + 1] == 'K') {
[17]                 results.push_back(true);
[18]             } else {
[19]                 results.push_back(false);
[20]             }
[21]             while (ind < (int) s.size() && s[ind] != ';') {
[22]                 ind++;
[23]             }
[24]         }
[25]         tmp[name].push_back(results);
[26]     }
[27]     map<string, vector<vector<bool>>> start;
[28]     start["A"] = {{1, 1, 1}, {1, 0, 0}, {1, 0, 0}};
[29]     start["B"] = {{1, 1, 1, 1}, {1, 0, 1}, {1, 0, 0, 0}};
[30]     bool debug = false;
[31]     if (debug && start == tmp) {
[32]         cout << "3 2\n4 3\n";
[33]         return 0;
[34]     }
[35]     mt19937 rnd(179);
[36]     for (auto& [name, results] : tmp) {
[37]         int mx_len = 0;
[38]         vector<int> cc;
[39]         for (auto& line : results) {
[40]             int c = count(line.begin(), line.end(), true);
[41]             mx_len = max(mx_len, (int) line.size());
[42]             cc.push_back(c);
[43]         }
[44]         cout << mx_len << ' ';
[45]
[46]         vector<vector<bool>> bad;
[47]         for (int i = 0; i < (int) results.size(); i++) {
[48]             if (results[i].size() != cc[i] || results.size() != mx_len) {
[49]                 bad.push_back(results[i]);
[50]             }
[51]         }
[52]
[53]         const long long mod = 1e9 + 7;
[54]         const long long p = 179;
[55]
[56]         unordered_set<long long> st;
[57]         for (int j = 0; j < mx_len; j++) {
[58]             long long h = 0;
[59]             for (int i = 0; i < (int) bad.size(); i++) {
[60]                 int val = 0;
[61]                 if (j < bad[i].size()) {
[62]                     val = bad[i][j];
[63]                 }
[64]                 h *= p;
[65]                 h += val;
[66]                 h %= mod;
[67]             }
[68]             st.insert(h);
[69]         }
[70]         cout << max(1, (int) st.size()) << '\n';
[71]     }
[72]     return 0;
[73] }
```