

Олимпиада СПбГУ по информатике 2022/23 учебного года

A	B	C	D	E	F	Sum
100	100	90	30	52	10	382

Task A ()

```
#include <iostream>
#include <vector>

using namespace std;

int main() {
    int x;
    vector<int> d1(6, -1), d2(6, -1);
    for (int i = 0; i < 6; i++) {
        cin >> x;
        x--;
        d2[x] = i;
        for (int j = x; j < 5; j++) {
            d2[j + 1] = d1[j];
        }
        for (int j = 0; j < x; j++) d2[j] = d1[j];
        d1 = d2;
        //for (int v : d2) cout << v + 1 << ' ';
        //cout << "\n";
    }
    vector<int> pos(6);
    for (int i = 0; i < 6; i++) pos[d1[i]] = i;
    for (int v : pos) cout << v + 1 << '␣';

    return 0;
}
```

Task B ()

```
#include <iostream>
#include <vector>
#include <string>

using namespace std;

int main() {
    string s;
    cin >> s;
    if (s[0] == 'f') {
        int n;
        cin >> n;

        vector<int> a(n);
        int sm = 0;
        for (int i = 0; i < n; i++) {
            cin >> a[i];
            sm += a[i];
        }
        cout << sm * 1000;
    }
    else {
        int n;
        cin >> n;
        vector<int> a(n);
        int sm = 0;
        for (int i = 0; i < n; i++) {
            cin >> a[i];
            sm += a[i] % 1000;
        }
        sm += a[0] / 1000;
        cout << sm;
    }

    return 0;
}
```

Task C ()

```
#include <iostream>
#include <vector>
#include <string>
#include <set>

using namespace std;

vector<pair<int, int>> ans;

struct st {
    int x0, y0, x1, y1;
};

vector<st> s;

vector<vector<int>> up(9, vector<int>(9)), down(9, vector<int>(9)), lft(9, vector<int>(9)), right
(9, vector<int>(9));

void add(st t) {
    for (int i = t.x0; i <= t.x1; i++) {
        up[i][t.y1] = 1;
        if (t.y0) up[i][t.y0 - 1] = 1;
    }
    for (int i = t.x0; i <= t.x1; i++) {
        down[i][t.y0] = 1;
        if (t.y1 + 1 < 9) down[i][t.y1 + 1] = 1;
    }

    for (int i = t.y0; i <= t.y1; i++) {
        lft[t.x0][i] = 1;
        if (t.x1 + 1 < 9) lft[t.x1 + 1][i] = 1;
    }
    for (int i = t.y0; i <= t.y1; i++) {
        right[t.x1][i] = 1;
        if (t.x0) right[t.x0 - 1][i] = 1;
    }
}

bool is_good(int x0, int y0, int x1, int y1) {
    for (int i = x0; i <= x1; i++) {
        for (int j = y0; j <= y1; j++) {
            if (i == x0) {
                if (!lft[i][j]) return 0;
            }
            if (i != x0 && !lft[i][j]) return 0;
            if (i == x1) {
                if (!right[i][j]) return 0;
            }
            if (i != x1 && !right[i][j]) return 0;
            if (j == y0) {
                if (!down[i][j]) return 0;
            }
            if (j != y0 && !down[i][j]) return 0;
            if (j == y1) {
                if (!up[i][j]) return 0;
            }
            if (j != y1 && !up[i][j]) return 0;
            //if (i != x0 && i != x1 && j != y0 && j != y1) {
            //    if (up[i][j] || down[i][j] || right[i][j] || lft[i][j]) return 0;
            //}
        }
    }
    return 1;
}

void check() {
    for (int i = 0; i < 9; i++) {
        for (int j = 0; j < 9; j++) {
            for (int il = i; il < 9; il++) {
                for (int jl = j; jl < 9; jl++) {
                    if (is_good(i, j, il, jl)) {
                        ans.push_back({ il - i + 1, jl - j + 1 });
                    }
                }
            }
        }
    }
}
```

```

                                //cout << i << ' ' << j << ' ' << i1 << ' ' << j1
                                << "x\n";
                                }
                            }
                        }
                    }
                }
            }

void ref() {
    for (int i = 0; i < 9; i++) {
        for (int j = 0; j < 9; j++) {
            up[i][j] = 0;
            down[i][j] = 0;
            lft[i][j] = 0;
            rght[i][j] = 0;
        }
    }
}

void big_add() {
    for (auto t : s) add(t);
}

void func(vector<pair<int, int>>& p) {
    s.clear();
    //vector<vector<int>> d(9, vector<int>(9));
    s.push_back({ 0, 0, p[0].first - 1, p[0].second - 1 });
    //add({ 0, 0, p[0].first - 1, p[0].second - 1 });
    if (p[1].first > p[0].first || p[1].second > p[0].second) return;

    s.push_back({ 0, 0, p[1].first - 1, p[1].second - 1 });
    //add({ 0, 0, p[1].first - 1, p[1].second - 1 });
    big_add();
    check();
    ref();

    if (p[2].first > p[0].first || p[2].second > p[0].second) return;

    if (p[2].first + p[1].first <= p[0].first && p[2].second <= p[0].second) {
        s.push_back({ p[1].first, 0, p[1].first + p[2].first - 1, p[2].second - 1 });
        //add({ p[1].first, 0, p[1].first + p[2].first - 1, p[2].second - 1 });
        big_add();
        //for (auto t : s) cout << t.x0 << ' ' << t.y0 << ' ' << t.x1 << ' ' << t.y1 << "\n";
        check();
        s.pop_back();
        ref();
    }

    if (p[2].second + p[1].second <= p[0].second && p[2].first <= p[0].first) {
        s.push_back({ 0, p[1].second, p[2].first - 1, p[1].second + p[2].second - 1 });
        //add({ p[1].first, 0, p[1].first + p[2].first - 1, p[2].second - 1 });
        big_add();
        //for (auto t : s) cout << t.x0 << ' ' << t.y0 << ' ' << t.x1 << ' ' << t.y1 << "\n";
        check();
        s.pop_back();
        ref();
    }

    if ((p[0].first - p[2].first >= p[1].first || p[0].second - p[2].second >= p[1].second)) {
        s.push_back({ p[0].first - p[2].first, p[0].second - p[2].second, p[0].first - 1,
            p[0].second - 1 });
        //add({ p[1].first, 0, p[1].first + p[2].first - 1, p[2].second - 1 });
        big_add();
        //for (auto t : s) cout << t.x0 << ' ' << t.y0 << ' ' << t.x1 << ' ' << t.y1 << "\n";
        check();
        s.pop_back();
        ref();
    }
}

```

```

        if (p[1].first + p[2].first <= p[0].first) {
            s.push_back({ p[1].first , p[0].second - p[2].second , p[1].first + p[2].first - 1 ,
                p[0].second - 1 });
            //add({ p[1].first , 0 , p[1].first + p[2].first - 1 , p[2].second - 1 });
            big_add();
            //for (auto t : s) cout << t.x0 << ' ' << t.y0 << ' ' << t.x1 << ' ' << t.y1 << "\n";
            check();
            s.pop_back();
            ref();
        }

        if (p[1].second + p[2].second <= p[0].second) {
            s.push_back({ p[0].first - p[2].first , p[1].second , p[0].first - 1 , p[1].second +
                p[2].second - 1 });
            //add({ p[1].first , 0 , p[1].first + p[2].first - 1 , p[2].second - 1 });
            big_add();
            //for (auto t : s) cout << t.x0 << ' ' << t.y0 << ' ' << t.x1 << ' ' << t.y1 << "\n";
            check();
            s.pop_back();
            ref();
        }
    }

    void gen(vector<pair<int , int>>& p) {
        vector<pair<int , int>> ps(3);
        for (int i = 0; i < 3; i++) {
            for (int j = 0; j < 3; j++) {
                if (i == j) continue;
                for (int k = 0; k < 3; k++) {
                    if (k == i || k == j) continue;
                    ps[0] = p[i];
                    ps[1] = p[j];
                    ps[2] = p[k];
                    func(ps);
                    swap(ps[1].first , ps[1].second);
                    func(ps);
                    swap(ps[2].first , ps[2].second);
                    func(ps);
                    swap(ps[1].first , ps[1].second);
                    func(ps);
                    swap(ps[2].first , ps[2].second);
                    func(ps);
                }
            }
        }

    }

    int main() {
        int a, b;
        vector<pair<int , int>> p(3);
        vector<vector<int>> d(9, vector<int>(9));
        for (int i = 0; i < 3; i++) {
            cin >> a >> b;
            p[i] = { a, b };
        }
        gen(p);
        //func(p);

        set <pair<int , int>> st;
        ans.push_back(p[0]);
        ans.push_back(p[1]);
        ans.push_back(p[2]);
        for (auto f : ans) st.insert({ min(f.first , f.second), max(f.first , f.second) });
        for (auto f : st) cout << f.first << ' ' << f.second << "\n";

        return 0;
    }

```

Task D ()

```
#include <iostream>
#include <vector>
#include <string>
#include <set>

using namespace std;

#define N 51

vector<int> dp0(N);
vector<vector<int>>> dp1(N, vector<int> (N));

vector<int> a, fl, prv;
int n;

bool calc() {
    int xr = 0;
    for (int i = 0; i < n; i++) {
        if (fl[i]) xr ^= dp1[i][a[i]];
        else xr ^= dp0[a[i]];
    }
    return xr != 0;
}

bool check(int i, int x) {
    bool ret;
    if (x == 0) {
        if (!fl[i]) return 0;
        fl[i] = 0;
        int tmp = a[i];
        a[i] = prv[i];
        ret = !calc();
        fl[i] = 1;
        a[i] = tmp;
    }
    else {
        if (a[i] < x) return 0;
        a[i] -= x;
        ret = !calc();
        a[i] += x;
    }
    return ret;
}

bool make_step() {
    for (int i = 0; i < n; i++) {
        for (int x = 0; x <= a[i]; x++) {
            if (check(i, x)) {
                cout << i + 1 << ' ' << x << endl;
                if (x == 0) {
                    fl[i] = 0;
                    a[i] = prv[i];
                }
                else a[i] -= x;
                return 1;
            }
        }
    }
    return 0;
}

int main() {
    cin >> n;
    a.resize(n);
    fl.resize(n, 1);
    for (int i = 0; i < n; i++) cin >> a[i];
    prv = a;
    for (int i = 1; i < N; i++) {
        vector<int> mex(N);
        for (int j = 0; j < i; j++) mex[dp0[j]] = 1;
        int mx = 0;
        for (int j = 0; j < N; j++) if (!mex[j]) {
```

```

        mx = j;
        break;
    }
    dp0[i] = mx;
}

for (int k = 0; k < n; k++) {
    for (int i = 0; i < N; i++) {
        vector<int> mex(N + 10);
        for (int j = 0; j < i; j++) mex[dp1[k][j]] = 1;
        mex[dp0[a[k]]] = 1;
        int mx = 0;
        for (int j = 0; j < N + 10; j++) if (!mex[j]) {
            mx = j;
            break;
        }
        dp1[k][i] = mx;
    }
}

if (calc()) {
    int i = 0, x = 0;
    while (i != -1 || x != -1) {
        if (!make_step()) return 0;
        cin >> i >> x;
        i--;
        if (i == -1 || x == -1) break;
        if (x == 0) {
            fl[i] = 0;
            a[i] = prv[i];
        }
        else a[i] = x;
    }
}
else {
    cout << -1 << '␣' << -1;
}

return 0;
}

```

Task E ()

```
#include <iostream>
#include <vector>
#include <string>
#include <set>
#include <map>
#include <algorithm>

using namespace std;

#define ll long long

ll cnk(int n, int k) {
    ll a = 1, b = 1, c = 1;
    for (int i = 2; i <= n; i++) a *= i;
    for (int i = 2; i <= k; i++) b *= i;
    for (int i = 2; i <= n - k; i++) c *= i;
    return a / (b * c);
}

#define N 1 << 100

vector<int> mask_less(vector<int> a) {
    vector<int> b(10);

    for (int i = 0; i < 9; i++) {
        b[i] = a[i] > a[i + 1];
    }
    b[9] = 1;
    return b;
}

vector<int> to_str(int mask) {
    vector<int> ret;
    for (int i = 9; i >= 0; i--) ret.push_back((mask >> i) & 1);
    return ret;
}

vector<vector<int>> dp;

void int_to_vct(int n, vector<int>& a) {
    int x = 0, sm;
    for (int i = 0; i < 10; i++) {
        sm = 0;
        for (int j = x; j < 11; j++) {
            if (sm + dp[i][j] > n) {
                n -= sm;
                a[i] = j;
                x = j;
                break;
            }
            if (sm + dp[i][j] == n) {
                n = 0;
                a[i] = j;
                x = j;
                break;
            }
            sm += dp[i][j];
        }
    }
}

int vct_to_int(vector<int>& a) {
    int ret = 0;
    for (int i = 0; i < 10; i++) {
        for (int j = 0; j < a[i]; j++) ret += dp[i][j];
    }
    return ret;
}

int main() {
```



```

dp.resize(10, vector<int>(11));
for (int j = 0; j < 11; j++) {
    dp[9][j] = 1;
}
for (int i = 8; i >= 0; i--) {
    for (int j = 0; j < 11; j++) {
        int sm = 0;
        for (int k = j; k < 11; k++) sm += dp[i + 1][k];
        dp[i][j] = sm;
        //cout << dp[i][j] << ' ' << 10 - i << ' ' << j << "\n";
    }
}

/*int n;
cin >> n;
vector<int> a(10);
int_to_vct(n, a);
cout << vct_to_int(a) + 1;

*/

```

```

string s;
vector<int> b(10);
int cnt;
int t;
cin >> t;
cin >> s;

if (s[0] == 'r') {
    while (t--) {
        for (int i = 0; i < 10; i++) {
            cin >> s;
            cnt = 0;
            for (int j = 0; j < 10; j++) cnt += s[j] == '1';
            b[i] = cnt;
        }
        sort(b.begin(), b.end());
        cout << vct_to_int(b) + 1 << "\n";
    }
}
else {
    int n;
    vector<int> a(10);
    while (t--) {
        cin >> n;
        int_to_vct(n, a);
        //for (int v : a) cout << v << ' ';
        for (int i = 0; i < 10; i++) {
            int j = 0;
            for (j = 0; j < a[i]; j++) cout << 1;
            for (; j < 10; j++) cout << 0;
            cout << "\n";
        }
        cout << "\n";
    }
}

return 0;
}

```

Task F ()

```
#include <iostream>
#include <vector>
#include <string>
#include <set>
#include <algorithm>

using namespace std;

#define ll long long

void decode(string& s, vector<int> &a) {
    int x = -1;
    int n = s.size();
    string t = "-1";
    for (int i = 0; i < n; i++) {
        if (s[i] == '(') {
            x = 0;
        }
        else if (s[i] == '|') {
            t = "";
        }
        else if (s[i] == ')') {
            for (int j = 0; j < x; j++) {
                for (int f = 0; f < t.size(); f++) {
                    a.push_back(((int)t[f] - '0'));
                }
            }
            x = -1;
            t = "-1";
        }
        else {
            if (x == -1) {
                a.push_back(((int)s[i] - '0'));
            }
            else {
                if (t[0] == '-') {
                    x *= 10;
                    x += ((int)s[i] - '0');
                }
                else t += s[i];
            }
        }
    }
    reverse(a.begin(), a.end());
}

void sm(vector<int>& a, vector<int>& b, vector<int>& c) {
    int n = max(a.size(), b.size());
    for (int i = (int)a.size(); i < n; i++) {
        a.push_back(0);
    }
    for (int i = (int)b.size(); i < n; i++) {
        b.push_back(0);
    }
    int x = 0;
    for (int i = 0; i < n; i++) {
        int s = a[i] + b[i] + x;
        c.push_back(s % 10);
        x = s / 10;
    }
    if (x) c.push_back(x);
}

int main() {
    string s, t;
    cin >> s >> t;
    vector<int> a, b, c;
    decode(s, a);
    //for (int v : a) cout << v;
    //cout << "\n";
    decode(t, b);
```

```
    //for (int v : b) cout << v;  
    //cout << "\n";  
    sm(a, b, c);  
    reverse(c.begin(), c.end());  
    for (int v : c) cout << v;  
  
    return 0;  
}
```