

# Олимпиада СПбГУ по информатике 2022/23 учебного года

A	B	C	D	E	F	Sum
100	100	90	60	28	65	443

## Task A ()

```
#define _CRT_SECURE_NO_WARNINGS
#include <iostream>
#include <vector>
#include <algorithm>
#include <string>
#include <numeric>
#include <string.h>
#include <math.h>
#include <queue>
#include <deque>
#include <map>
#include <set>
#include <stdio.h>
#include <random>
#include <bitset>
using ll = long long;
using ld = long double;
#define all(x) x.begin(), x.end()
using namespace std;

signed main() {
    cin.tie(0) -> sync_with_stdio(0);
#ifdef LOCAL
    (void) freopen("D:\\Users\\ilyin\\source\\repos\\input.txt", "r", stdin);
#endif
    vector<int> v(6);
    for (int i = 0; i < 6; ++i) {
        cin >> v[i];
    }

    vector<int> v1(6);
    iota(all(v1), 1);

    do {
        int ok = 0;
        vector<int> cnt(7);
        for (int i = 0; i < 6; ++i) {
            int res = 0;
            for (int j = 0; j < v1[i]; ++j) {
                res += cnt[j];
            }
            if (res + 1 != v[i]) {
                ok = 1;
                break;
            }
            cnt[v1[i]] += 1;
        }
        if (!ok) {
            for (auto i : v1) {
                cout << i << ' ';
            }
            cout << endl;
            return 0;
        }
    } while (next_permutation(all(v1)));
}
```

```
}    return 0;
```

## Task B ()

```
#define _CRT_SECURE_NO_WARNINGS
#include <iostream>
#include <vector>
#include <algorithm>
#include <string>
#include <numeric>
#include <string.h>
#include <math.h>
#include <queue>
#include <deque>
#include <map>
#include <set>
#include <stdio.h>
#include <random>
#include <bitset>
#define int ll
using ll = long long;
using ld = long double;
#define all(x) x.begin(), x.end()
using namespace std;

signed main() {
    cin.tie(0) -> sync_with_stdio(0);
#ifdef LOCAL
    (void) freopen("D:\\Users\\ilyin\\source\\repos\\input.txt", "r", stdin);
#endif
    string s;
    cin >> s;
    if (s == "first") {
        int n;
        cin >> n;
        int su = 0;
        for (int i = 0; i < n; ++i) {
            int q;
            cin >> q;
            su += q * 100000;
        }
        cout << su;
    }
    else {
        int n;
        cin >> n;
        int su = 0;
        vector<int> v(n);
        for (int i = 0; i < n; ++i) {
            int q;
            cin >> q;
            v[i] = q;
            su += q;
        }
        string s = to_string(su);
        for (int i = 0; i < 5; ++i) {
            s[s.size() - i - 1] = '0';
        }
        int res = stoll(s);
        res = res / n;
        res = res / 100000;
        int sua = res;
        int ans = res;
        for (int i = 0; i < n; ++i) {
            ans += v[i] - 100000 * res;
        }
        cout << ans << endl;
    }

    return 0;
}
```

## Task C ()

```
#define _CRT_SECURE_NO_WARNINGS
#include <iostream>
#include <vector>
#include <algorithm>
#include <string>
#include <numeric>
#include <string.h>
#include <math.h>
#include <queue>
#include <deque>
#include <map>
#include <set>
#include <stdio.h>
#include <random>
#include <bitset>
#define int ll
using ll = long long;
using ld = long double;
#define all(x) x.begin(), x.end()
using namespace std;

signed main() {
    cin.tie(0) -> sync_with_stdio(0);
#ifdef LOCAL
    (void) freopen("D:\\Users\\ilyin\\source\\repos\\input.txt", "r", stdin);
#endif
    int a1, b1, a2, b2, a3, b3;
    cin >> a1 >> b1 >> a2 >> b2 >> a3 >> b3;
    if (b1 < a1) {
        swap(b1, a1);
    }
    if (b2 < a2) {
        swap(b2, a2);
    }
    if (b3 < a3) {
        swap(b3, a3);
    }
    vector<pair<int, int>> v = { {a1, b1}, {a2, b2}, {a3, b3} };
    set<pair<int, int>> ans = { {a1, b1}, {a2, b2}, {a3, b3} };

    for (int i = 0; i < v.size(); ++i) {
        pair<int, int> box1 = v[i];
        for (int k1 = 0; k1 < 2; ++k1) {
            if (k1 == 1) {
                swap(box1.first, box1.second);
            }
            for (int j = 0; j < v.size(); ++j) {
                if (j == i)
                    continue;

                pair<int, int> box2 = v[j];
                for (int k = 0; k < 2; ++k) {
                    if (k == 1) {
                        swap(box2.first, box2.second);
                    }
                    if (box1.second < box2.second || box1.first < box2.first)
                        continue;
                }
                if (box1.first == box2.first && box1.second != box2.second)
                    ans.insert({ min(box1.first, box1.second - box2.second), max(box1.first, box1.second - box2.second) });
            }
            if (box1.second == box2.second && box1.first != box2.first)
                ans.insert({ min(box1.second, box1.first - box2.first), max(box1.second, box1.first - box2.first) });
        }
    }
}
```

```

for (int i1 = 0; i1 < v.size(); ++i1) {
    if (i1 == i || i1 == j)
        continue;
    for (int k2 = 0; k2 < 2; ++k2) {
        pair<int, int> box3 = v[i1];
        if (k2 == 1) {
            swap(box3.first, box3.second);
        }
        if (box1.second < box3.second || box1.
            first < box3.first)
            continue;
        if (box2.first == box1.first && box2.
            second != box1.second) {
            ans.insert({ min(box1.first, box1.
                second - box2.second - box3.
                second),
                max(box1.first, box1.
                    second - box2.second -
                    box3.second) });
        }
        if (box3.first == box1.first && box3.
            second != box1.second) {
            ans.insert({ min(box1.first, box1.
                second - box2.second - box3.
                second),
                max(box1.first, box1.
                    second - box2.second -
                    box3.second) });
        }
        if (box3.second == box2.second && box3.
            first + box2.first == box1.first) {
            ans.insert({ min(box1.second -
                box2.second, box1.first), max(
                box1.second - box2.second,
                box1.first) });
        }
        if (box3.first == box2.first && box3.
            second + box2.second == box1.second) {
            ans.insert({ min(box1.first - box2.
                .first, box1.second),
                max(box1.first - box2.
                    first, box1.second) });
        }
        if (box3.first == box1.first - box2.first)
        {
            for (int iter = box3.second; iter
                <= box3.second; ++iter) {
                if (iter + box2.second >=
                    box1.second && iter <
                    box1.second) {
                    ans.insert({ min(
                        box1.first -
                        box2.first,
                        box1.second -
                        iter)
                        , max(box1.
                            .first
                            -
                            box2.
                            first,
                            box1.
                            second
                            -
                            iter)
                    });
                }
            }
        }
        if (box3.second == box1.second - box2.
            second) {
            for (int iter = box3.first; iter
                <= box3.first; ++iter) {

```

```

        if (iter + box2.first >=
            box1.first && iter <
            box1.first) {
            ans.insert({ min(
                box1.second -
                box2.second,
                box1.first -
                iter)
                , max(box1
                    .second
                    -
                    box2.
                    second
                    , box1
                    .first
                    -
                    iter)
                });
        }
    }
    if (box3.first >= box1.first - box2.first
        && box2.second + box3.second <= box1.
        second) {
        ans.insert({ min(box2.second, box1
            .first - box2.first),
            max(box2.second, box1.
                first - box2.first) });
        ;
    }
    if (box3.second >= box1.second - box2.
        second && box3.first + box2.first <=
        box1.first) {
        ans.insert({ min(box2.first, box1.
            second - box2.second),
            max(box2.first, box1.
                second - box2.second)
            });
    }
}
/*if (ans.find({ 0, 2 }) != ans.end()) {
    cout << i << ' ' << k1 << ' ' << j << ' '
        << k << ' ' << il << endl;
}*/
}
}
}
}
}
//cout << ans.size() << endl;
for (auto i : ans) {
    if (i.first > 0)
        cout << i.first << ' ' << i.second << endl;
}
return 0;
}

```

## Task D ()

```
#define _CRT_SECURE_NO_WARNINGS
#include <iostream>
#include <vector>
#include <algorithm>
#include <string>
#include <numeric>
#include <string.h>
#include <math.h>
#include <queue>
#include <deque>
#include <map>
#include <set>
#include <stdio.h>
#include <random>
#include <bitset>
using ll = long long;
using ld = long double;
#define all(x) x.begin(), x.end()
using namespace std;

signed main() {
    // cin.tie(0) -> sync_with_stdio(0);
    // #ifdef LOCAL
    // (void) freopen("D:\\Users\\ilyin\\source\\repos\\input.txt", "r", stdin);
    // #endif
    int n;
    cin >> n;
    vector<int> v(n);
    for (int i = 0; i < n; ++i) {
        cin >> v[i];
    }

    if (n == 1) {
        cout << 1 << ' ' << v[0] << endl;

        int a, b;
        cin >> a >> b;

        cout << 1 << ' ' << v[0] << endl;
    }
    else {
        // cout << -1 << ' ' << -1 << endl;
        // return 0;
        if (v[0] == v[1]) {
            cout << -1 << ' ' << -1 << endl;
            return 0;
        }
        if (v[1] > v[0]) {
            cout << 2 << ' ' << v[1] - v[0] << endl;
            int a, b;
            cin >> a >> b;
            while (a != -1) {
                cout << ((a % 2) + 1) << ' ' << b << endl;
                cin >> a >> b;
            }
        }
        else {
            cout << 1 << ' ' << v[0] - v[1] << endl;
            int a, b;
            cin >> a >> b;
            while (a != -1) {
                cout << ((a % 2) + 1) << ' ' << b << endl;
                cin >> a >> b;
            }
        }
    }
    return 0;
}
```

## Task E ()

```
#define _CRT_SECURE_NO_WARNINGS
#include <iostream>
#include <vector>
#include <algorithm>
#include <string>
#include <numeric>
#include <string.h>
#include <math.h>
#include <queue>
#include <deque>
#include <map>
#include <set>
#include <stdio.h>
#include <random>
#include <bitset>
using ll = long long;
using ld = long double;
#define all(x) x.begin(), x.end()
using namespace std;
vector<int> v;
map<vector<int>, int> ans;
map<int, vector<int>> ans1;
void rec(int idx) {
    if (idx == 10) {
        ans1[ans1.size()] = v;
        ans[v] = ans1.size();
        return;
    }
    for (int i = 0; i <= 10; ++i) {
        if (idx == 0 || i >= v.back()) {
            v.push_back(i);
            rec(idx + 1);
            v.pop_back();
        }
    }
}

signed main() {
    cin.tie(0) -> sync_with_stdio(0);
#ifdef LOCAL
    (void) freopen("D:\\Users\\ilyin\\source\\repos\\input.txt", "r", stdin);
#endif
    rec(0);
    int t;
    cin >> t;
    string s;
    cin >> s;
    while (t--) {
        if (s == "transmit") {
            int n;
            cin >> n;
            vector<vector<int>> v1(10, vector<int>(10, 0));
            for (int i = 0; i < 10; ++i) {
                for (int j = 0; j < ans1[n][i]; ++j) {
                    v1[i][j] = 1;
                }
            }
            for (int i = 0; i < 10; ++i) {
                for (int j = 0; j < 10; ++j) {
                    cout << v1[i][j];
                }
                cout << endl;
            }
            cout << endl;
        }
        else {
            vector<int> v1;
            for (int i = 0; i < 10; ++i) {
                int cnt = 0;
                string q;
                cin >> q;
                for (int j = 0; j < 10; ++j) {
```



```

        cnt += q[j] - '0';
    }
    v1.push_back(cnt);
}
sort(all(v1));
cout << ans[v1] << endl;
}

return 0;
}

```

## Task F ()

```
#define _CRT_SECURE_NO_WARNINGS
#include <iostream>
#include <vector>
#include <algorithm>
#include <string>
#include <numeric>
#include <string.h>
#include <math.h>
#include <queue>
#include <deque>
#include <map>
#include <set>
#include <stdio.h>
#include <random>
#include <bitset>
using ll = long long;
using ld = long double;
#define all(x) x.begin(), x.end()
using namespace std;

signed main() {
    cin.tie(0) -> sync_with_stdio(0);
#ifdef LOCAL
    (void) freopen("D:\\Users\\ilyin\\source\\repos\\input.txt", "r", stdin);
#endif

    string s1;
    cin >> s1;
    string s2;
    cin >> s2;
    string v1 = "";
    int i = 0;
    string v2 = "";
    while (i < s1.size()) {
        if (s1[i] != '(') {
            v1 += s1[i];
        }
        i += 1;
    }
    int j = 0;
    while (j < s2.size()) {
        if (s2[j] != '(') {
            v2 += s2[j];
        }
        j += 1;
    }
    string res = "";
    while (s1.size() < s2.size()) {
        s1 = '0' + s1;
    }
    while (s2.size() < s1.size()) {
        s2 = '0' + s2;
    }
    int su = 0;
    for (int i = s2.size() - 1; i >= 0; --i) {
        int q = s2[i] - '0';
        int q1 = s1[i] - '0';
        int qq = q + q1 + su;
        int q2 = qq % 10;
        res += to_string(q2);
        su = qq / 10;
    }

    if (su != 0) {
        res += to_string(su);
    }
    reverse(all(res));
    cout << res << endl;
    return 0;
}
```