

Олимпиада СПбГУ по информатике 2022/23 учебного года

A	B	C	D	E	F	Sum
100	100	100	10	52	10	372

Task A ()

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

using namespace std;

typedef long long ll;

int main() {
    ios_base::sync_with_stdio(false);
    cin.tie(0);
    cout.tie(0);

    vector<int> a;
    for (int i = 0; i < 6; ++i) {
        int x;
        cin >> x;
        --x;
        vector<int> newa;
        for (int j = 0; j < x; ++j) {
            newa.push_back(a[j]);
        }
        newa.push_back(i);
        for (int j = x; j < a.size(); ++j) {
            newa.push_back(a[j]);
        }
        swap(a, newa);
    }
    vector<int> ans(6);
    for (int i = 0; i < 6; ++i) {
        ans[a[i]] = i + 1;
    }
    for (auto x : ans)
        cout << x << " ";
}
```

Task B ()

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

using namespace std;

typedef long long ll;

const ll D = 1e9;

int main() {
    ios_base::sync_with_stdio(false);
    cin.tie(0);
    cout.tie(0);

    string type;
    cin >> type;
    int n;
    cin >> n;
    vector<ll> a(n);
    for (int i = 0; i < n; ++i) {
        cin >> a[i];
    }
    if (type == "first") {
        ll sum = 0;
        for (auto x : a)
            sum += x;
        cout << sum * D << "\n";
    } else {
        ll f = 0, sum = 0;
        for (int i = 0; i <= 10000; ++i) {
            if (a[0] - D * i < D) {
                f = i;
                break;
            }
        }
        for (int i = 0; i < n; ++i) {
            sum += a[i] - f * D;
        }
        cout << sum + f << "\n";
    }
}
```

Task C ()

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

using namespace std;

typedef long long ll;

void show_f(vector<vector<int>>& f) {
    for (int i = 0; i < f.size(); ++i) {
        cout << "┌───";
        for (int j = 0; j < f[0].size(); ++j) {
            cout << f[i][j] << "┐";
        }
        cout << "\n";
    }
    cout << "\n";
}

set<pair<int, int>> s;

bool check_empty(vector<vector<int>>& f, int x1, int y1, int x2, int y2) {
    int col = f[x1][y1];
    for (int i = x1; i <= x2; ++i) {
        for (int j = y1; j <= y2; ++j) {
            if (f[i][j] != col) {
                return false;
            }
        }
    }
    return true;
}

int check_sides(vector<vector<int>>& f, int x1, int y1, int x2, int y2) {
    int n = f.size(), m = f[0].size();
    int col = f[x1][y1];
    bool full = true;
    bool touchx = false, touchy = false;
    for (int i = x1; i <= x2; ++i) {
        if (!(y1 - 1 < 0 || f[i][y1 - 1] != col) || !(y2 + 1 >= m || f[i][y2 + 1] != col)) {
            full = false;
        }
        if ((y1 - 1 < 0 || f[i][y1 - 1] != col) || (y2 + 1 >= m || f[i][y2 + 1] != col)) {
            touchx = true;
        }
    }
    for (int j = y1; j <= y2; ++j) {
        if (!(x1 - 1 < 0 || f[x1 - 1][j] != col) || !(x2 + 1 >= n || f[x2 + 1][j] != col)) {
            full = false;
        }
        if ((x1 - 1 < 0 || f[x1 - 1][j] != col) || (x2 + 1 >= n || f[x2 + 1][j] != col)) {
            touchy = true;
        }
    }
    if (full)
        return 2;
    if (touchx && touchy)
        return 1;
    return 0;
}

void check(vector<vector<int>>& f) {
    for (int x = 1; x <= f.size(); ++x) {
        for (int y = 1; y <= f[0].size(); ++y) {
            if (s.count({ x, y }) != 0 || s.count({ y, x }) != 0)
                continue;
            bool xy = false;
            for (int si = 0; si + x <= f.size(); ++si) {
                for (int sj = 0; sj + y <= f[0].size(); ++sj) {
                    bool is = check_empty(f, si, sj, si + x - 1, sj + y - 1);
```

```

        is = (is && check_sides(f, si, sj, si + x - 1, sj + y - 1)
              == 2);
        if (is) {
            xy = true;
            break;
        }
    }
    if (xy)
        break;
}
if (xy) {
    s.insert({ x, y });
}
}
}

bool can_put(vector<vector<int>>& f, int x1, int y1, int x2, int y2) {
    if (x2 >= f.size() || y2 >= f[0].size())
        return false;
    bool ee = check_empty(f, x1, y1, x2, y2);
    bool ss = check_sides(f, x1, y1, x2, y2);
    return ee && ss >= 1;
}

void put(vector<vector<int>>& f, int x1, int y1, int x2, int y2, int col) {
    for (int i = x1; i <= x2; ++i) {
        for (int j = y1; j <= y2; ++j) {
            f[i][j] = col;
        }
    }
}

void rec(vector<bool>& used, vector<vector<int>>& f, vector<pair<int, int>>& a, int col) {
    for (int j = 0; j < 3; ++j) {
        if (used[j]) continue;
        used[j] = true;
        for (int i1 = 0; i1 < f.size(); ++i1) {
            for (int j1 = 0; j1 < f[0].size(); ++j1) {
                if (can_put(f, i1, j1, i1 + a[j].first - 1, j1 + a[j].second - 1))
                {
                    vector<vector<int>> f2 = f;
                    put(f2, i1, j1, i1 + a[j].first - 1, j1 + a[j].second - 1,
                        col);
                    check(f2);
                    //show_f(f2);
                    rec(used, f2, a, col + 1);
                }
                if (can_put(f, i1, j1, i1 + a[j].second - 1, j1 + a[j].first - 1))
                {
                    vector<vector<int>> f2 = f;
                    put(f2, i1, j1, i1 + a[j].second - 1, j1 + a[j].first - 1,
                        col);
                    //show_f(f2);
                    check(f2);
                    rec(used, f2, a, col + 1);
                }
            }
        }
        used[j] = false;
    }
}

int main() {
    ios_base::sync_with_stdio(false);
    cin.tie(0);
    cout.tie(0);

    vector<pair<int, int>> a(3);
    for (int i = 0; i < 3; ++i) {
        cin >> a[i].first >> a[i].second;
    }
}

```

```

vector<bool> used(3, false);
for (int i = 0; i < 3; ++i) {
    used[i] = true;
    vector<vector<int>> f(a[i].first, vector<int>(a[i].second));
    check(f);
    rec(used, f, a, 1);
    used[i] = false;
}
vector<pair<int, int>> ans;
for (auto x : s) {
    if (x.first < x.second)
        ans.push_back(x);
    else
        ans.push_back({ x.second, x.first });
}
sort(ans.begin(), ans.end());
for (auto x : ans) {
    cout << x.first << " " << x.second << "\n";
}
}

```

Task D ()

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

using namespace std;

map<vector<int>, map<vector<bool>, pair<int, int>>>> m;
vector<int> b;

bool rec(vector<int>& a, vector<bool>& used) {
    auto [f1, f2] = m[a][used];
    if (f1 == -1)
        return false;
    if (f1 != 0)
        return true;
    for (int i = 0; i < a.size(); ++i) {
        if (!used[i]) {
            int ai = a[i];
            a[i] = b[i];
            used[i] = true;
            bool res = rec(a, used);
            a[i] = ai;
            used[i] = false;
            if (res == false) {
                m[a][used] = { i + 1, 0 };
                return true;
            }
        }
        for (int j = 1; j <= a[i]; ++j) {
            a[i] -= j;
            bool res = rec(a, used);
            a[i] += j;
            if (res == false) {
                m[a][used] = { i + 1, j };
                return true;
            }
        }
    }
    m[a][used] = { -1, -1 };
    return false;
}

int main() {
    ios_base::sync_with_stdio(false);
    cin.tie(0);
    cout.tie(0);

    int n;
    cin >> n;
    vector<int> a(n);
    for (int i = 0; i < n; ++i) {
        cin >> a[i];
    }
    b = a;
    if (n == 1) {
        cout << 1 << "\n" << a[0] << endl;
        int x, y;
        cin >> x >> y;
        if (x != -1) {
            cout << 1 << "\n" << a[0] << endl;
        }
        return 0;
    }
    vector<bool> used(n, false);
    rec(a, used);
    a = b;
    while (true) {
        pair<int, int> f = m[a][used];
        if (f.first <= 0) {
            cout << -1 << "\n" << -1 << endl;
        }
    }
}
```

```

        return 0;
    }
    cout << f.first << "┘" << f.second << endl;
    if (f.second == 0) {
        a[f.first - 1] = b[f.first - 1];
        used[f.first - 1] = true;
    } else {
        a[f.first - 1] -= f.second;
    }
    int x, y;
    cin >> x >> y;
    if (x == -1)
        return 0;
    if (y == 0) {
        a[x - 1] = b[x - 1];
        used[x - 1] = true;
    } else
        a[x - 1] -= y;
    }
}

```

Task E ()

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

using namespace std;

void solve1() {
    int n;
    cin >> n;
    vector<vector<int>>> a(10, vector<int>(10, 0));
    for (int i = 0; i < 10; ++i) {
        for (int j = 0; j < 10; ++j) {
            if (n > 0) {
                a[i][j] = 1;
                --n;
            }
        }
    }
    for (int i = 0; i < 10; ++i) {
        for (int j = 0; j < 10; ++j) {
            cout << a[i][j];
        }
        cout << "\n";
    }
    cout << "\n";
}

void solve2() {
    vector<vector<int>>> a(10, vector<int>(10, 0));
    int cnt = 0;
    for (int i = 0; i < 10; ++i) {
        string s;
        cin >> s;
        for (int j = 0; j < 10; ++j) {
            a[i][j] = s[j] - '0';
            cnt += a[i][j];
        }
    }
    cout << cnt << "\n";
}

int main() {
    ios_base::sync_with_stdio(false);
    cin.tie(0);
    cout.tie(0);

    int t;
    cin >> t;
    string type;
    cin >> type;
    while (t--) {
        if (type == "transmit")
            solve1();
        else
            solve2();
    }
}
```


Task F ()

```
a = int(input())  
b = int(input())  
print(a + b)
```