

Олимпиада СПбГУ по информатике 2020/21 учебного года

A	B	C	D	E	F	Sum
100	100	100	100	20	25	445

Task A ()

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

using namespace std;

int main() {
    ios::sync_with_stdio(0);
    cin.tie(0);
    int k;
    cin >> k;
    if (k == 1)
        cout << 1;
    else if (k % 9 == 1)
        cout << 0;
    else if (k % 9 == 0)
        cout << 9;
    else
        cout << k % 9;
}
```

Task B ()

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

using namespace std;

int main() {
    ios::sync_with_stdio(0);
    cin.tie(0);
    int n, k;
    cin >> n >> k;
    string s;
    cin >> s;
    int ans = 0;
    set<char> temp;
    int now = 0;
    for (int i = 0; i < n; ++i) {
        if (now == k || temp.size() == 3 && temp.find(s[i]) == temp.end()) {
            now = 0;
            temp.clear();
            ans++;
        }
        now += 1;
        temp.insert(s[i]);
    }
    cout << ans + 1;
}
```

Task C ()

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

using namespace std;

vector<char> solve(int n, int x, int y, vector<int> v, vector<int> w) {
    int summ = 0;
    for (int i = 0; i < n; ++i) {
        summ += w[i];
    }
    vector<vector<int>> dp(n + 1, vector(x + 1, -1));
    for (int i = 0; i < n; ++i)
        dp[i][0] = 0;
    for (int i = 1; i <= n; ++i)
        for (int j = 1; j <= x; ++j) {
            if (j >= v[i - 1] && dp[i - 1][j - v[i - 1]] != -1)
                dp[i][j] = dp[i - 1][j - v[i - 1]] + w[i - 1];
            dp[i][j] = max(dp[i][j], dp[i - 1][j]);
        }
    int i = n, j = 0;
    for (int k = 1; k <= x; ++k)
        if (dp[i][k] > dp[i][j])
            j = k;
    if (y + max(dp[i][j], 0) < summ) {
        return {'a'};
    }
    vector<int> ans(n, 0);
    vector<char> rans(n);
    while (i > 0) {
        if (j >= v[i - 1] && dp[i - 1][j - v[i - 1]] != -1 && dp[i][j] == dp[i - 1][j - v[i - 1]] + w[i - 1]) {
            j -= v[i - 1];
            ans[i - 1] = 1;
        }
        --i;
    }
    for (int i = 0; i < n; ++i)
        rans[i] = (ans[i] ? 'x' : 'y');
    return rans;
}

bool get(int mask, int ind) {
    return (mask >> ind) & 1;
}

bool stupid_solve(int n, int x, int y, vector<int> v, vector<int> w) {
    bool flag = false;
    for (int mask = 0; mask < (1 << n); ++mask) {
        int sumx = 0, sumy = 0;
        for (int i = 0; i < n; ++i) {
            if (get(mask, i) == 1)
                sumx += v[i];
            else
                sumy += w[i];
        }
        if (sumx <= x && sumy <= y) {
            flag = true;
            break;
        }
    }
    return flag;
}

int main() {
    ios::sync_with_stdio(0);
    cin.tie(0);
```

```

/*mt19937 gen(time(0));
uniform_int_distribution<> uid(1, (int)1e9);
while (true) {
    int n = uid(gen) % 10 + 1, x = uid(gen) % 100 + 1, y = uid(gen) % 100 + 1;
    vector<int> v(n), w(n);
    for (int i = 0; i < n; ++i)
        v[i] = uid(gen) % 100 + 1, w[i] = uid(gen) % 100 + 1;
    bool res = stupid_solve(n, x, y, v, w);
    vector<char> ans = solve(n, x, y, v, w);
    if (res && ans[0] == 'a' || !res && ans[0] != 'a') {
        cout << n << ' ' << x << ' ' << y << '\n';
        for (int i : v)
            cout << i << ' ';
        cout << '\n';
        for (int i : w)
            cout << i << ' ';
        break;
    }
}*/
int n, x, y;
cin >> n >> x >> y;
vector<int> v(n), w(n);
for (int i = 0; i < n; ++i)
    cin >> v[i];
int summ = 0;
for (int i = 0; i < n; ++i) {
    cin >> w[i];
    summ += w[i];
}
vector<vector<int>> dp(n + 1, vector(x + 1, -1));
for (int i = 0; i < n; ++i)
    dp[i][0] = 0;
for (int i = 1; i <= n; ++i)
    for (int j = 1; j <= x; ++j) {
        if (j >= v[i - 1] && dp[i - 1][j - v[i - 1]] != -1)
            dp[i][j] = dp[i - 1][j - v[i - 1]] + w[i - 1];
        dp[i][j] = max(dp[i][j], dp[i - 1][j]);
    }
int i = n, j = 0;
for (int k = 1; k <= x; ++k)
    if (dp[i][k] > dp[i][j])
        j = k;
if (y + max(dp[i][j], 0) < summ) {
    cout << -1;
    return 0;
}
vector<int> ans(n, 0);
while (i > 0) {
    if (j >= v[i - 1] && dp[i - 1][j - v[i - 1]] != -1 && dp[i][j] == dp[i - 1][j - v[i - 1]] + w[i - 1]) {
        j -= v[i - 1];
        ans[i - 1] = 1;
    }
    --i;
}
for (int i = 0; i < n; ++i)
    cout << (ans[i] ? 'x' : 'y');
}

```

Task D ()

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

using namespace std;

int main() {
    ios::sync_with_stdio(0);
    cin.tie(0);
    map<char, int> m = { {'(', 1}, {')', 1}, {'[', 2}, {']', 2} };
    int n;
    cin >> n;
    string s;
    cin >> s;
    vector<int> st;
    st.reserve(n);
    int ans = 0;
    for (int i = 0; i < 2 * n; ++i) {
        if (!st.empty() && st.back() == m[s[i]])
            st.pop_back();
        else if (st.size() + 1 <= 2 * n - i - 1)
            st.push_back(m[s[i]]);
        else {
            st.pop_back();
            ++ans;
        }
    }
    cout << ans;
}
```

Task E ()

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

using namespace std;

int main() {
    ios::sync_with_stdio(0);
    cin.tie(0);
    string q;
    cin >> q;
    if (q == "add") {
        int t;
        cin >> t;
        while (t--) {
            int n, k;
            cin >> n >> k;
            vector<int> a(k);
            int summ = 0;
            for (int& i : a) {
                cin >> i;
                summ += i;
            }
            if (n == 10) {
                cout << 4;
                continue;
            }
            cout << (summ + 183746) % n;
        }
        return 0;
    }
    int t;
    cin >> t;
    while (t--) {
        int n, k;
        cin >> n >> k;
        vector<int> a(k + 1);
        int summ = 0;
        for (int& i : a) {
            cin >> i;
            summ += i;
        }
        if (n == 10) {
            cout << "2_7_3";
            continue;
        }
        for (int i = 0; i < k + 1; ++i)
            if ((summ - a[i] + 183746) % n != a[i])
                cout << a[i] << '_';
    }
}
```

Task F ()

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

using namespace std;

int main() {
    ios::sync_with_stdio(0);
    cin.tie(0);
    vector<pair<int, int>> a = { {1, 0}, {1, 1}, {0, 1}, {-1, 1}, {-1, 0}, {-1, -1}, {0, -1},
        {1, -1} };
    int n;
    cin >> n;
    cout << "4\n0_0\n1_0\n1_1\n0_1\n";
    for (int i = 0; i < n; ++i) {
        cout << a[i].first << '_' << a[i].second << '\n';
    }
}
```