

Олимпиада СПбГУ по информатике 2020/21 учебного года

A	B	C	D	E	F	Sum
100	100	100	100	55	25	480

Task A ()

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

#pragma warning (disable : 4996)

using namespace std;

int main() {
#ifdef _DEBUG
    freopen("input.txt", "r", stdin);
#endif
    ios_base::sync_with_stdio(false);
    cin.tie(0);
    cout.tie(0);
    int k;
    cin >> k;
    cout << (k + (k - 2) / 9) % 10;
    return 0;
}
```

Task B ()

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

#pragma warning (disable : 4996)

using namespace std;

int main() {
#ifdef _DEBUG
    freopen("input.txt", "r", stdin);
#endif
    ios_base::sync_with_stdio(false);
    cin.tie(0);
    cout.tie(0);
    int n, k;
    cin >> n >> k;
    string s;
    cin >> s;
    vector<int> dp(n);
    multiset<int> get_min;
    get_min.insert(0);
    multiset<char> c;
    set<int> d;
    int en = n;
    for (int i = n - 1; i >= 0; i--) {
        c.insert(s[i]);
        d.insert(s[i]);
        while ((en - i) > k || d.size() > 3) {
            if (en == n)
                get_min.erase(get_min.find(0));
            else {
                get_min.erase(get_min.find(dp[en]));
                c.erase(c.find(s[en]));
                if (c.find(s[en]) == c.end())
                    d.erase(s[en]);
            }
            en--;
        }
        dp[i] = *(get_min.begin()) + 1;
        if (en - i < k) {
            if (en == n - 1)
                dp[i] = min(dp[i], 1);
            else if (en < n - 1)
                dp[i] = min(dp[i], dp[en + 1] + 1);
        }
        get_min.insert(dp[i]);
    }
    cout << dp[0];
    return 0;
}
```

Task C ()

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

#pragma warning (disable : 4996)

using namespace std;

int dp[502][250010];

int main() {
#ifdef _DEBUG
    freopen("input.txt", "r", stdin);
#endif
    ios_base::sync_with_stdio(false);
    cin.tie(0);
    cout.tie(0);
    int n;
    cin >> n;
    int x, y;
    cin >> x >> y;
    vector<int> v(n + 1);
    vector<int> w(n + 1);
    int sumw = 0;
    for (int i = 1; i <= n; i++)
        cin >> v[i];
    for (int i = 1; i <= n; i++)
        cin >> w[i], sumw += w[i];
    for (int i = 1; i <= n; i++) {
        for (int j = 0; j <= x; j++) {
            dp[i][j] = max(dp[i - 1][j], dp[i][j]);
            if (j + v[i] <= x)
                dp[i][j + v[i]] = max(dp[i][j + v[i]], dp[i - 1][j] + w[i]);
        }
    }
    pair<int, int> st = { -1, -1 };
    for (int i = x; i >= 0; i--) {
        if (sumw - dp[n][i] <= y) {
            st = { n, i };
            break;
        }
    }
    if (st.first == -1) {
        cout << -1;
        return 0;
    }
    vector<char> ans(n);
    while (st.first > 0) {
        int i = st.first, j = st.second;
        if (j - v[i] >= 0 && dp[i][j] == dp[i - 1][j - v[i]] + w[i]) {
            ans[i - 1] = 'x';
            st = { i - 1, j - v[i] };
        }
        else {
            ans[i - 1] = 'y';
            st.first--;
        }
    }
    for (char c : ans)
        cout << c;
    return 0;
}
```

Task D ()

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

#pragma warning (disable : 4996)

using namespace std;

bool type(char a, char b) {
    return abs(a - b) <= 2;
}

int main() {
#ifdef _DEBUG
    freopen("input.txt", "r", stdin);
#endif
    ios_base::sync_with_stdio(false);
    cin.tie(0);
    cout.tie(0);
    int n;
    cin >> n;
    string s;
    cin >> s;
    vector<char> st;
    int res = 0;
    for (int i = 0; i < 2 * n; i++) {
        if (st.size() && type(st.back(), s[i]))
            st.pop_back();
        else {
            st.push_back(s[i]);
        }
    }
    cout << st.size() / 2;
    return 0;
}
```

Task E ()

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

#pragma warning (disable : 4996)

using namespace std;

const int ans = 743851;

int dp[11][11][11];
set<vector<int>> s;

void add() {
    int n, k;
    cin >> n >> k;
    vector<int> v(k);
    for (int& i : v)
        cin >> i;
    sort(v.begin(), v.end());
    if (n == 10) {
        cout << dp[v[0]][v[1]][v[2]] << "\n";
        return;
    }
    cout << ans << "\n";
}

void clear() {
    int n, k;
    cin >> n >> k;
    k++;
    vector<int> v(k);
    for (int& i : v)
        cin >> i;
    if (n == 10) {
        sort(v.begin(), v.end());
        if (dp[v[0]][v[1]][v[2]] == v[3]) {
            cout << v[0] << "_" << v[1] << "_" << v[2] << endl;
            return;
        }
        while (next_permutation(v.begin(), v.end())) {
            if (dp[v[0]][v[1]][v[2]] == v[3]) {
                cout << v[0] << "_" << v[1] << "_" << v[2] << endl;
                return;
            }
        }
    }
    for (int i : v)
        if (i != ans)
            cout << i << "_";
    cout << endl;
    return;
}

int main() {
#ifdef _DEBUG
    freopen("input.txt", "r", stdin);
#endif
    ios_base::sync_with_stdio(false);
    cin.tie(0);
    cout.tie(0);
    for (int i = 1; i <= 10; i++) {
        for (int j = i + 1; j <= 10; j++) {
            for (int k = j + 1; k <= 10; k++) {
                for (int p = 1; p <= 10; p++) {
                    if (p != i && p != j && p != k) {
                        vector<int> tmp = { i, j, p, k };
                        sort(tmp.begin(), tmp.end());
                        if (s.find(tmp) == s.end()) {
                            s.insert(tmp);
                            dp[i][j][k] = p;
                        }
                    }
                }
            }
        }
    }
}
```

```

        break;
    }
}
}
}
}
string comm;
cin >> comm;
if (comm == "add") {
    int t;
    cin >> t;
    while (t--) {
        add();
    }
}
else {
    int t;
    cin >> t;
    while (t--) {
        clear();
    }
}
return 0;
}

```

Task F ()

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

#pragma warning (disable : 4996)

using namespace std;

void add() {
    int n, k;
    cin >> n >> k;
    vector<int> v(k);
    for (int& i : v)
        cin >> i;
    sort(v.begin(), v.end());
    for (int i = 0; i < k; i++) {
        if (i == 0) {
            if (v[i] - 1 > 2) {
                cout << 2 << endl;
                return;
            }
        }
        else {
            if (v[i] - v[i - 1] > 2) {
                cout << v[i - 1] + 1 << endl;
                return;
            }
        }
    }
    if (n - v.back() > 2)
        cout << v[k - 1] + 1 << endl;
}

void clear() {
    int n, k;
    cin >> n >> k;
    k++;
    vector<int> v(k);
    for (int& i : v)
        cin >> i;
    sort(v.begin(), v.end());
    int ans = -1;
    if (v[1] - 1 > 2)
        ans = v[0];
    else {
        for (int i = 2; i < k; i++) {
            if (v[i] - v[i - 2] > 2) {
                ans = v[i - 1];
                break;
            }
        }
    }
    if (ans == -1) {
        if (n - v[k - 2] > 2)
            ans = v[k - 1];
    }
    for (int i : v)
        if (i != ans)
            cout << i << "_";
    cout << endl;
    return;
}

int main() {
#ifdef _DEBUG
    freopen("input.txt", "r", stdin);
#endif
    ios_base::sync_with_stdio(false);
    cin.tie(0);
    cout.tie(0);
```

```

int n;
cin >> n;
cout << 4 << "\n";
cout << "0_0\n0_1\n1_1\n1_0\n";
vector<pair<int, int>> ans(8);
ans[0] = { 1, 1 };
ans[1] = { 1, 0 };
ans[2] = { 1, -1 };
ans[3] = { 0, -1 };
ans[4] = { 0, 1 };
ans[5] = { -1, -1 };
ans[6] = { -1, 0 };
ans[7] = { -1, 1 };
for (int i = 0; i < n; i++)
    cout << ans[i].first << "_" << ans[i].second << "\n";
return 0;
}

```