

# 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 <bits/stdc++.h>

using namespace std;

typedef unsigned long long ull;
typedef long long ll;

const int INF = 1e9 + 1;
const ll INFF = 1e18 + 1;
const int SZ = 1e3;

int main() {
    //ios_base::sync_with_stdio(0); cin.tie(0); cout.tie(0);
    int k, x; cin >> k;
    x = (k - 11) / 9 + 1;
    if (k <= 10) cout << k % 10;
    else cout << (k + x) % 10;
}
```

## Task B ()

```
#include <bits/stdc++.h>

using namespace std;

typedef unsigned long long ull;
typedef long long ll;

const int INF = 1e9 + 1;
const ll INFF = 1e18 + 1;
const int SZ = 1e3;

int main() {
    //ios_base::sync_with_stdio(0); cin.tie(0); cout.tie(0);
    int n, k, cur = 0, ans = 1;
    string t;
    cin >> n >> k >> t;
    set<char> s;
    for (int i = 0; i < n; i++) {
        s.insert(t[i]);
        if (s.size() > 3 || cur + k <= i) {
            s.clear();
            cur = i;
            ans++;
            s.insert(t[i]);
        }
    }
    cout << ans;
}
```

## Task C ()

```
#include <bits/stdc++.h>

using namespace std;

typedef unsigned long long ull;
typedef long long ll;

int main() {
    ios_base::sync_with_stdio(0); cin.tie(0); cout.tie(0);
    int n, x, y; cin >> n >> x >> y;
    vector<pair<int, int>> a(n + 1);
    vector<char> ans(n + 1, 'y');
    ll sum = 0;
    for (int i = 1; i <= n; i++) {
        cin >> a[i].first;
    }
    for (int i = 1; i <= n; i++) {
        cin >> a[i].second;
        sum += a[i].second;
    }
    vector<vector<int>> dp(n + 1, vector<int> (x + 1));
    for (int j = 1; j <= x; j++) {
        dp[0][j] = 0;
    }
    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++) {
            dp[i][j] = dp[i - 1][j];
            if (j - a[i].first >= 0) {
                dp[i][j] = max(dp[i][j], dp[i - 1][j - a[i].first] + a[i].second);
            }
        }
    }
    if (sum - dp[n][x] > y) {
        cout << "-1";
        return 0;
    }
    int i = n, j = x;
    while (i > 0 && j > 0) {
        if (dp[i][j] == dp[i - 1][j]) {
            ans[i] = 'y';
        }
        else {
            j -= a[i].first;
            ans[i] = 'x';
        }
        i--;
    }
    for (int k = 1; k <= n; k++) {
        cout << ans[k];
    }
}
```

## Task D ()

```
#include <bits/stdc++.h>

using namespace std;

typedef unsigned long long ull;
typedef long long ll;

const int INF = 1e9 + 1;
const ll INFF = 1e18 + 1;
const int SZ = 1e3;

int main() {
    //ios_base::sync_with_stdio(0); cin.tie(0); cout.tie(0);
    int n, cnt = 1;
    string s;
    cin >> n >> s;
    vector<char> v, t;
    for (int i = 0; i < n * 2; i++) {
        if (s[i] == ')') s[i] = '(';
        if (s[i] == '[') s[i] = ']';
    }
    // for (int i = 0; i < n * 2; i++) {
    //     cout << s[i] << '\n';
    // }
    for (int i = 0; i < n * 2 - 1; i++) {
        if (s[i + 1] != s[i]) {
            if (cnt % 2 == 1) v.push_back(s[i]);
            cnt = 1;
        }
        else {
            cnt++;
        }
    }
    if (cnt % 2 == 1) v.push_back(s[2 * n - 1]);
    stack<char> st;
    for (int i = 0; i < v.size(); i++) {
        if (!st.empty() && st.top() == v[i]) {
            st.pop();
        }
        else {
            st.push(v[i]);
        }
    }
    cout << st.size() / 2;
}
```

## Task E ()

```
#include <bits/stdc++.h>

using namespace std;

typedef unsigned long long ull;
typedef long long ll;

vector<int> p1(3), p2(4);

map<vector<int>, vector<int>> mp_add, mp_clear;

vector<vector<int>> v;
vector<bool> used;

int n, k;

const int key = 698143;

void gen1(int cur, int last) {
    if (cur == 3) {
        for (int i = 0; i < v.size(); i++) {
            if (!used[i]) {
                bool flag = true;
                for (int k = 0; k < 3; k++) {
                    if (find(v[i].begin(), v[i].end(), p1[k]) == v[i].end()) {
                        flag = false;
                        break;
                    }
                }
                if (flag) {
                    used[i] = true;
                    mp_add[p1] = v[i];
                    mp_clear[v[i]] = p1;
                    return;
                }
            }
        }
        for (int i = last + 1; i <= 10; i++) {
            p1[cur] = i;
            gen1(cur + 1, i);
        }
    }
}

void gen2(int cur, int last) {
    if (cur == 4) {
        v.push_back(p2);
        return;
    }
    for (int i = last + 1; i <= 10; i++) {
        p2[cur] = i;
        gen2(cur + 1, i);
    }
}

int main() {
    ios_base::sync_with_stdio(0); cin.tie(0); cout.tie(0);
    gen2(0, 0);
    used.resize(v.size(), false);
    gen1(0, 0);
    for (auto it : mp_add) {
        for (auto &i : it.first) cout << i << ' ';
        cout << "/ ";
        for (auto &i : it.second) cout << i << ' ';
        cout << '\n';
    }
    string s; cin >> s;
    if (s == "add") {
        int t; cin >> t;
        while (t--) {
            cin >> n >> k;
            vector<bool> was(11, false);
```

```

vector<int> temp(k);
for (int i = 0; i < k; i++) {
    cin >> temp[i];
    if (n == 10) was[temp[i]] = true;
}
if (n == 10 && k == 3) {
    sort(temp.begin(), temp.end());
    for (auto &i : mp_add[temp]) {
        if (!was[i]) {
            cout << i << '\n';
            break;
        }
    }
}
else if (n == 1000000) {
    cout << key << '\n';
}
else {
    sort(temp.begin(), temp.end());
    if (temp[0] > 3) {
        cout << "1\n";
    }
    else {
        cout << n << '\n';
    }
}
}
else {
    int t; cin >> t;
    while (t--) {
        cin >> n >> k;
        vector<int> temp(k + 1);
        for (auto &i : temp) cin >> i;
        if (n == 10 && k == 3) {
            sort(temp.begin(), temp.end());
            for (auto &i : mp_clear[temp]) cout << i << '\u';
            cout << '\n';
        }
        else if (n == 1000000){
            for (auto it : temp) {
                if (it != key) {
                    cout << it << '\u';
                }
            }
            cout << '\n';
        }
        else {
            sort(temp.begin(), temp.end());
            if (temp[1] > 3) {
                for (int i = 1; i < temp.size(); i++) {
                    cout << temp[i] << '\u';
                }
            }
            else {
                for (int i = 0; i < temp.size() - 1; i++) {
                    cout << temp[i] << '\u';
                }
            }
            cout << '\n';
        }
    }
}
}
}

```

## Task F ()

```
#include <bits/stdc++.h>

using namespace std;

typedef unsigned long long ull;
typedef long long ll;

const int INF = 1e9 + 1;
const ll INFF = 1e18 + 1;
const int SZ = 1e3;

int main() {
    //ios_base::sync_with_stdio(0); cin.tie(0); cout.tie(0);
    int n; cin >> n;
    if (n == 1) {
        cout << "4\n";
        cout << "0_0\n";
        cout << "1_0\n";
        cout << "1_1\n";
        cout << "0_1\n";

        cout << "0_1\n";
    }
    else if (n == 2) {
        cout << "4\n";
        cout << "0_0\n";
        cout << "1_0\n";
        cout << "1_1\n";
        cout << "0_1\n";

        cout << "0_1\n";
        cout << "0_-1\n";
    }
    else if (n == 3) {
        cout << "4\n";
        cout << "0_0\n";
        cout << "1_0\n";
        cout << "1_1\n";
        cout << "0_1\n";

        cout << "0_1\n";
        cout << "0_-1\n";
        cout << "1_0\n";
    }
    else if (n == 4) {
        cout << "4\n";
        cout << "0_0\n";
        cout << "1_0\n";
        cout << "1_1\n";
        cout << "0_1\n";

        cout << "0_1\n";
        cout << "0_-1\n";
        cout << "1_0\n";
        cout << "-1_0\n";
    }
    else if (n == 5) {
        cout << "4\n";
        cout << "0_0\n";
        cout << "1_0\n";
        cout << "1_1\n";
        cout << "0_1\n";

        cout << "0_1\n";
        cout << "0_-1\n";
        cout << "1_0\n";
        cout << "-1_0\n";
        cout << "1_1\n";
    }
    else if (n == 6) {
        cout << "4\n";
        cout << "0_0\n";
    }
}
```

```

        cout << "1_0\n";
        cout << "1_1\n";
        cout << "0_1\n";

        cout << "0_1\n";
        cout << "0_-1\n";
        cout << "1_0\n";
        cout << "-1_0\n";
        cout << "1_1\n";
        cout << "1_-1\n";
    }
    else if (n == 7) {
        cout << "4\n";
        cout << "0_0\n";
        cout << "1_0\n";
        cout << "1_1\n";
        cout << "0_1\n";

        cout << "0_1\n";
        cout << "0_-1\n";
        cout << "1_0\n";
        cout << "-1_0\n";
        cout << "1_1\n";
        cout << "1_-1\n";
        cout << "-1_1\n";
    }
    else if (n == 8) {
        cout << "4\n";
        cout << "0_0\n";
        cout << "1_0\n";
        cout << "1_1\n";
        cout << "0_1\n";

        cout << "0_1\n";
        cout << "0_-1\n";
        cout << "1_0\n";
        cout << "-1_0\n";
        cout << "1_1\n";
        cout << "1_-1\n";
        cout << "-1_1\n";
        cout << "-1_-1\n";
    }
}
}

```