Олимпиада СПбГУ по информатике 2021/22 учебного года

A	B	C	D	E	F	Sum
100	100	100	80	58	31	469

Task A ()

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

using namespace std;

int main() {
    int n, i, j;
    long long a, b;

    cin >> n;

    map<long long, long long> zeros;

    for (i = 0; i < n; i++) {
        cin >> a >> b;

        while (a % 10 == 0) {
            b += 1;
            a /= 10;
        }

        zeros[b] += a;
    }

    auto it = zeros.begin();

    while (it->second % 10 == 0) {
        zeros[it->first + 1] += it->second / 10;
        zeros.erase(it);
        it = zeros.begin();
    }

    cout << it->first << '\n';
}
```

Task B ()

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

using namespace std;

int main() {
    int n, i, j;

    cin >> n;

    vector<int> t(n);

    for (i = 0; i < n; i++) {
        cin >> t[i];
    }

    for (i = n - 1; i >= 0; i--) {
        string str;
        while (t[i] != 0) {
            cout << "Flip and wait" << endl;
            cin >> str;
            int beeps = count(str.begin(), str.end(), 'e') / 2;
            while (beeps <= i) {
                cout << "Wait" << endl;
                cin >> str;
                beeps += count(str.begin(), str.end(), 'e') / 2;
            }
            t[i]--;
        }
    }

    cout << "Stop" << endl;
}
```

Task C ()

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

using namespace std;

void solve() {
    string s;

    cin >> s;

    int one_count = 0;
    int zero_count = 0;
    int question_count = 0;
    int i, n = s.size();

    for (char ch : s) {
        one_count += (ch == '1');
        zero_count += (ch == '0');
        question_count += (ch == '?');
    }

    if (question_count) {
        if (s == "?0") {
            s = "10";
        } else if (s == "?1") {
            s = "01";
        } else if (one_count && zero_count) {
            for (i = 1; i < n; i++) {
                s[i] = s[1];
            }
        } else if ((one_count == 1 || zero_count == 1) && s[0] != '?') {
            for (i = 1; i < n; i++) {
                s[i] = s[0];
            }
        } else if (one_count) {
            for (i = 0; i < n; i++) {
                if (s[i] == '?') {
                    s[i] = '0';
                }
            }
        } else {
            for (i = 0; i < n; i++) {
                if (s[i] == '?') {
                    s[i] = '1';
                }
            }
        }
    } else if (s == "10") {
        s = "?0";
    } else if (s == "01") {
        s = "?1";
    } else if (!one_count || !zero_count) {
        for (i = 1; i < n; i++) {
            s[i] = '?';
        }
    } else if (one_count == 1 && s[0] == '1') {
        for (i = 2; i < n; i++) {
            s[i] = '?';
        }
    } else if (zero_count == 1 && s[0] == '0') {
        for (i = 2; i < n; i++) {
            s[i] = '?';
        }
    } else if (one_count > zero_count) {
        for (i = 0; i < n; i++) {
            if (s[i] == '1') {
                s[i] = '?';
            }
        }
    } else {
        for (i = 0; i < n; i++) {
            if (s[i] == '0') {
                s[i] = '?';
```

```
            }
        }
    }

    cout << s << '\n';
}

int main() {
    int t;

    cin >> t;

    while (t--) {
        solve();
    }
}
```

Task D ()

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

using namespace std;

constexpr long long mod = 998244353;
constexpr long long part = 3e8;

int main() {
    long long n, i;

    cin >> n;
    n += 1;
    long long zeros = n / mod;
    n %= mod;
    vector<long long> precomputed = {
            zeros % 2 ? mod - 1 : 1,
            zeros % 2 ? 236544645 : 761699708,
            zeros % 2 ? 486622545 : 511621808,
            zeros % 2 ? 122364049 : 875880304,
    };
    long long position = min((long long) precomputed.size() - 1, n / part);
    long long result = precomputed[position];

    for (i = position * part + 1; i <= n; i++) {
        result = (result * i % mod);
    }

    cout << zeros << ' ' << result << '\n';
}
```

Task E ()

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

using namespace std;

int main() {
    int n, i, j;
    int q, a, b, d;

    cin >> n;

    vector<int> v(n);

    for (i = 0; i < n; i++) {
        cin >> v[i];
    }

    cin >> q;

    for (i = 0; i < q; i++) {
        cin >> a >> b >> d;
        long long result = 0;

        for (j = a; j < b; j++) {
            result += d / v[j] + (d % v[j] ? 1 : 0);
        }

        cout << result << '\n';
    }
}
```

Task F ()

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

using namespace std;

int main() {
    int n, i, j;

    cin >> n;

    vector<int> l(n);
    vector<int> r(n);

    for (i = 0; i < n; i++) {
        cin >> l[i] >> r[i];
    }

    vector<long long> best(n + 1, 0);

    for (i = n - 1; i >= 0; i--) {
        long long left_sum = 0;
        long long right_sum = 0;
        for (j = i; j < n; j++) {
            left_sum += l[j];
            right_sum += r[j];
            if (left_sum <= 0 && right_sum >= 0) {
                best[i] = max(best[i], best[j + 1] + 1);
            }
        }
    }

    cout << *max_element(best.begin(), best.end()) + 1 << '\n';
}
```