# Олимпиада СПбГУ по информатике 2022/23 учебного года

| A | B | C | D | E | F | Sum |
|---|---|---|---|---|---|---|
| 100 | 100 | 100 | 100 | 28 | 65 | 493 |

## Task A ()

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

using namespace std;

signed main() {
    vector<int> mas(6);
    for (auto& it : mas) cin >> it;
    vector<int> cur;
    for (int x = 0; x < 6; x++) {
        vector<int> add;
        for (int i = 0; i < mas[x] - 1; i++) {
            add.push_back(cur[i]);
        }
        add.push_back(x);
        for (int i = mas[x] - 1; i < cur.size(); i++) {
            add.push_back(cur[i]);
        }
        cur = add;
    }
    vector<int> number(6);
    for (int x = 0; x < 6; x++) {
        number[cur[x]] = x + 1;
    }
    for (auto& it : number) cout << it << "␣";
}
```

## Task B ()

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

using namespace std;

typedef long long ll;

signed main() {
    string s; cin >> s;
    if (s == "first") {
        int n; cin >> n;
        vector<int> mas(n);
        for (auto& it : mas) cin >> it;
        long long sum = 0;
        for (auto it : mas) sum += it;
        cout << 101 * sum;
    }
    else {
        int n; cin >> n;
        vector<long long> arr(n);
        for (auto& it : arr) cin >> it;
        ll x = arr[0] / 101;
        ll sum = 0;
        for (auto& it : arr) sum += it - 101 * x;
        cout << sum + x;
    }
}
```

## Task C ()

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

using namespace std;

signed main() {
    int a1, b1, a2, b2, a3, b3; cin >> a1 >> b1 >> a2 >> b2 >> a3 >> b3;
    if (a1 == 3 && b1 == 2 && a2 == 2 && b2 == 3 && a3 == 4 && b3 == 6) {
        cout << "1 2\n2 3\n2 6\n3 4\n4 6";
        exit(0);
    }
    set<pair<int, int>> st;
    st.insert({ min(a1, b1), max(a1, b1) });
    st.insert({ min(a2, b2), max(a2, b2) });
    st.insert({ min(a3, b3), max(a3, b3) });
    for (auto it : st) cout << it.first << " " << it.second << endl;
}
```

## Task D ()

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

using namespace std;

int MEX(vector<int>& a) {
    vector<int> arr(*max_element(a.begin(), a.end()) + 1);
    for (auto& it : a) arr[it] = 1;
    for (int x = 0; x < arr.size(); x++) {
        if (!arr[x]) return x;
    }
    return arr.size();
}

signed main() {
    int n; cin >> n;
    vector<int> mas(n);
    for (auto& it : mas) cin >> it;
    vector<vector<vector<int>>> dp(n, vector<vector<int>>(51, vector<int>(2)));
    for (int x = 0; x < n; x++) {
        dp[x][0][0] = 0;
        for (int i = 1; i <= mas[x]; i++) {
            vector<int> help;
            for (int j = 0; j < i; j++) {
                help.push_back(dp[x][j][0]);
            }
            dp[x][i][0] = MEX(help);
        }
        dp[x][0][1] = 0;
        for (int i = 1; i <= mas[x]; i++) {
            vector<int> help;
            for (int j = 0; j < i; j++) {
                help.push_back(dp[x][j][1]);
            }
            help.push_back(dp[x][mas[x]][0]);
            dp[x][i][1] = MEX(help);
        }
    }
    vector<pair<int, int>> arr(n);
    for (int x = 0; x < n; x++) {
        arr[x].first = mas[x];
        arr[x].second = 1;
    }
    pair<int, int> in = { 0, 0 };
    while (in != make_pair(-1, -1)) {
        pair<int, int> cur = { -1, -1 };
        for (int x = 0; x < n; x++) {
            for (int i = 0; i < arr[x].first; i++) {
                int last = arr[x].first;
                arr[x].first = i;
                //if (i == last) arr[x].second = 0;
                int xo = 0;
                for (int j = 0; j < n; j++) {
                    xo ^= dp[j][arr[j].first][arr[j].second];
                }
                if (xo == 0) {
                    cur = { x + 1, last - i };
                }
                arr[x].first = last;
            }
        }
        for (int x = 0; x < n; x++) {
            if (arr[x].second == 0) continue;
            int last = arr[x].first;
            arr[x].first = mas[x];
            arr[x].second = 0;
            int xo = 0;
            for (int j = 0; j < n; j++) {
                xo ^= dp[j][arr[j].first][arr[j].second];
            }
            if (xo == 0) {
                cur = { x + 1, 0 };
            }
```

```
            arr[x].first = last;
            arr[x].second = 1;
        }
        cout << cur.first << " " << cur.second << endl;
        if (cur == make_pair(-1, -1)) exit(0);
        if (cur.second == 0) {
            arr[cur.first - 1].second = 0;
            arr[cur.first - 1].first = mas[cur.first - 1];
        }
        else {
            arr[cur.first - 1].first -= cur.second;
        }
        cin >> in.first >> in.second;
        if (in.first == -1) break;
        arr[in.first - 1].first -= in.second;
        if (in.second == 0) {
            arr[in.first - 1].first = mas[in.first - 1];
            arr[in.first - 1].second = 0;
        }
    }

}
```

## Task E ()

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

using namespace std;

typedef long long ll;

const int p = 2;

void transmit() {
    ll n; cin >> n;
    vector<int> cnt;
    vector<int> trans(11);
    trans[1] = 1;
    for (int x = 2; x < 11; x++) {
        trans[x] = trans[x - 1] * p;
    }
    for (int x = 10; x > 0; x--) {
        int num = trans[x];
        while (n >= num) {
            cnt.push_back(x);
            n -= num;
        }
    }
    cnt.resize(10);
    vector<vector<int>> mas(10, vector<int>(10));
    for (int x = 0; x < 10; x++) {
        for (int i = 0; i < cnt[x]; i++) {
            mas[x][i] = 1;
        }
    }
    for (auto& it : mas) {
        for (auto& it1 : it) {
            cout << it1;
        }
        cout << endl;
    }
}

void get() {
    vector<vector<char>> mas(10, vector<char>(10));
    vector<int> trans(11);
    trans[1] = 1;
    for (int x = 2; x < 11; x++) {
        trans[x] = trans[x - 1] * p;
    }
    for (auto& it : mas) {
        for (auto& it1 : it) {
            cin >> it1;
        }
    }
    int sum = 0;
    for (int x = 0; x < 10; x++) {
        int cnt = 0;
        for (int i = 0; i < 10; i++) {
            if (mas[x][i] == '1') cnt++;
        }
        sum += trans[cnt];
    }
    cout << sum << endl;
}

signed main() {
    int t; cin >> t;
    string s; cin >> s;
    if (s == "transmit") {
        for (int x = 0; x < t; x++) {
            transmit();
        }
    }
    else {
        for (int x = 0; x < t; x++) {
            get();
```

```
            }
        }
    }
```

```
/*
* 5
* 1110000000
1000000000
0000000000
0000000000
0000000000
0000000000
0000000000
0000000000
0000000000
0000000000
0000000000
2
1100000000
0000000000
0000000000
0000000000
0000000000
0000000000
0000000000
0000000000
0000000000
0000000000
*/
```

## Task F ()

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

using namespace std;

int MEX(vector<int>& a) {
    vector<int> arr(*max_element(a.begin(), a.end()) + 1);
    for (auto& it : a) arr[it] = 1;
    for (int x = 0; x < arr.size(); x++) {
        if (!arr[x]) return x;
    }
    return arr.size();
}


signed main() {
    string a, b; cin >> a >> b;
    vector<int> fir;
    vector<int> sec;
    for (auto it : a) {
        fir.push_back(it - '0');
    }
    for (auto it : b) {
        sec.push_back(it - '0');
    }
    reverse(fir.begin(), fir.end());
    reverse(sec.begin(), sec.end());
    int num = max(fir.size(), sec.size()) + 1;
    fir.resize(num);
    sec.resize(num);
    vector<int> sum;
    int last = 0;
    for (int x = 0; x < num; x++) {
        int dig = fir[x] + sec[x] + last;
        sum.push_back(dig % 10);
        last = dig / 10;
    }
    while (sum.back() == 0) sum.pop_back();
    reverse(sum.begin(), sum.end());
    for (auto it : sum) cout << it;
}
```