Олимпиада СПбГУ по информатике 2021/22 учебного года

A	B	C	D	E	F	Sum
100	100	100	80	58	0	438

Task A ()

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

using namespace std;
map <int , int> nmb;
vector <pair <int , int>> nmp;
int main()
{
    int n, ans = 0;
    cin >> n;
    for (int i = 0; i < n; ++i)
    {
        int a, b, k = 0;
        cin >> a >> b;
        while (a)
        {
            nmb[k + b] += a % 10;
            a /= 10;
            ++k;
        }
    }
    for (auto it = nmb.begin(); it != nmb.end(); ++it)
        nmp.push_back(*it);
    /*for (int i = 0; i < nmp.size(); ++i)
        cout << nmp[i].first << " " << nmp[i].second << "\n";*/
    for (int i = 0; i < nmp.size(); ++i)
    {
        ans = nmp[i].first;
        if (!nmp[i].second)
            continue;
        if (nmp[i].second % 10)
            break;
        int k = 1;
        while (nmp[i + 1].first != nmp[i].first + k)
        {
            nmp[i].second /= 10;
            if (nmp[i].second % 10)
                break;
            ++k;
        }
        if (nmp[i + 1].first != nmp[i].first + k)
        {
            ans = nmp[i].first + k;
            break;
        }
        else
            nmp[i + 1].second += nmp[i].second / 10;
    }
    cout << ans;
    return 0;
}
```

Task B ()

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

using namespace std;
int a[101];
int main()
{
    int n;
    cin >> n;
    for (int i = 0; i < n; ++i)
        cin >> a[i];
    for (int i = n - 1; i >= 0; --i)
    {
        while (a[i])
        {
            cout << "Flip and wait" << endl;
            int cw = 0;
            while (cw <= i)
            {
                int ke = 0;
                string s;
                cin >> s;
                for (int i = 0; i < s.size(); ++i)
                    if (s[i] == 'e')
                        ++ke;
                ke /= 2;
                cw += ke;
                if (cw <= i)
                    cout << "Wait" << endl;
            }
            --a[i];
        }
    }
    cout << "Stop";
    return 0;
}
```

Task C ()

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

using namespace std;
const int N = 1e5;
string s[N];
int main()
{
    int t;
    cin >> t;
    bool ques = false;
    for (int i = 0; i < t; ++i)
    {
        cin >> s[i];
        for (int j = 0; j < s[i].size(); ++j)
            if (s[i][j] == '?')
                ques = true;
    }
    if (ques)
    {
        for (int i = 0; i < t; ++i)
        {
            int nq = 0;
            char sv = '?';
            bool ah = false;
            for (int j = 0; j < s[i].size(); ++j)
                if (s[i][j] != '?')
                {
                    if (s[i].size() % 2 == 0 && j >= s[i].size() / 2)
                        ah = true;
                    ++nq;
                    sv = s[i][j];
                }
            if (nq > s[i].size() / 2 || (s[i].size() % 2 == 0 && !ah && nq == s[i].size() / 2))
                ;
            else
                if (sv == '0')
                    sv = '1';
                else
                    sv = '0';
            for (int j = 0; j < s[i].size(); ++j)
                if (s[i][j] == '?')
                    s[i][j] = sv;
            cout << s[i] << "\n";
        }
        return 0;
    }
    for (int i = 0; i < t; ++i)
    {
        int nq[2] = { 0, 0 };
        char ch;
        for (int j = 0; j < s[i].size(); ++j)
            ++nq[s[i][j] - '0'];
        if (nq[0] == s[i].size() || nq[1] == s[i].size())
        {
            for (int j = (s[i].size() + 1) / 2; j < s[i].size(); ++j)
                s[i][j] = '?';
        }
        else
        {
            if (nq[0] == nq[1])
            {
                if (s[i][0] == '0')
                    ch = '0';
                else
                    ch = '1';
            }
            else if (nq[0] < nq[1])
                ch = '1';
            else
                ch = '0';
            for (int j = 0; j < s[i].size(); ++j)
                if (s[i][j] == ch)
```

```
                    s[i][j] = '?';
            }
            cout << s[i] << "\n";
        }
        return 0;
}
```

Task D ()

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

using namespace std;
const long long MD = 998244353;
pair <int, int> nft[12];
long long fact(long long n)
{
    int ind = (n * 10) / MD;
    if (nft[ind + 1].first == n)
        return nft[ind + 1].second;
    long long rft = nft[ind].second;
    for (int i = nft[ind].first + 1; i <= n; ++i)
    {
        rft *= i;
        rft %= MD;
    }
    return rft;
}
int main()
{
    nft[0] = { 0, 1 };
    nft[1] = { 99824435,  656466042 };
    nft[2] = { 199648870, 388653856 };
    nft[3] = { 299473305, 43736361 };
    nft[4] = { 399297740, 619654534 };
    nft[5] = { 499122175, 173167436 };
    nft[6] = { 598946610, 5066314 };
    nft[7] = { 698771045, 199050241 };
    nft[8] = { 798595480, 91231272 };
    nft[9] = { 898419915, 237011908 };
    nft[10] = { 998244350, 499122176 };
    nft[11] = { 998244353, 0 };
    long long n, ans = 1;
    cin >> n;
    ++n;
    cout << n / MD + n / (MD * MD) << "␣";
    for (int i = 1; i <= n / MD; ++i)
        ans *= fact(MD - 1);
    ans *= fact(n % MD);
    n /= MD;
    for (int i = 1; i <= n / MD; ++i)
        ans *= fact(MD - 1);
    ans *= fact(n % MD);
    cout << ans;
}
```

Task E ()

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

using namespace std;
const int MD = 998244353, N = 3e5;
int v[N];
int main()
{
    int n, m;
    cin >> n;
    for (int i = 0; i < n; ++i)
        cin >> v[i];
    cin >> m;
    for (int i = 0; i < m; ++i)
    {
        int a, b, d;
        cin >> a >> b >> d;
        long long ans = 0;
        for (int i = a; i < b; ++i)
            ans += (d + v[i] - 1) / v[i];
        cout << ans << "\n";
    }
}
```

Task F ()