Олимпиада СПбГУ по информатике 2021/22 учебного года

A	B	C	D	E	F	Sum
100	100	100	60	58	31	449

Task A ()

```
#define _CRT_SECURE_NO_WARNINGS

#include <iostream>
#include <algorithm>

using namespace std;

#define endl '\n'

const int Base = 2e5 + 99;

int n;
pair<long long, long long> num[Base];

int main()
{
#ifdef _DEBUG
        freopen("input.txt", "r", stdin);
#endif // _DEBUG

        ios_base::sync_with_stdio(0);
        cin.tie(0);

        cin >> n;
        for (int i = 0; i < n; i++)
        {
                cin >> num[i].second >> num[i].first;
        }
        sort(num, num + n);

        int zeroCnt = 0;
        long long sum = 0;
        int id = 0;
        while (sum % 10 == 0)
        {
                if (sum == 0)
                {
                        zeroCnt = num[id].first;
                }
                else if (sum % 10 == 0)
                {
                        sum /= 10;
                        zeroCnt++;
                }
                else
                {
                        break;
                }
                while (id < n && zeroCnt == num[id].first)
                {
                        sum += num[id].second;
                        id++;
                }
        }

        cout << zeroCnt << endl;
}
```

Task B ()

```
#define _CRT_SECURE_NO_WARNINGS

#include <iostream>
#include <algorithm>
#include <string>

using namespace std;

//#define endl '\n'

const int Base = 200;

int n;
int k[Base];

void wait(int id)
{
        int cnt = 0;
        while (true)
        {
                string res;
                cin >> res;
                if (res == "Burn" || res == "Fail")
                {
                        exit(0);
                }

                int d = res.size() - 3;
                d /= 2;

                cnt += d;
                if (cnt > id)
                {
                        return;
                }
                cout << "Wait" << endl;
        }
}

int main()
{
#ifdef _DEBUG
        //freopen("input.txt", "r", stdin);//
#endif // _DEBUG

        ios_base::sync_with_stdio(0);
        cin.tie(0);

        int sum = 0;
        cin >> n;
        for (int i = 0; i < n; i++)
        {
                cin >> k[i];
                sum += k[i];
        }
        cout << "Flip␣and␣wait" << endl;

        int ClockId = n - 1;
        while (true)
        {
                while (ClockId >= 0 && k[ClockId] == 0)
                {
                        ClockId--;
                }
                while (k[ClockId] > 0)
                {
                        wait(ClockId);
                        sum--;
                        k[ClockId]--;
                        if (sum == 0)
                        {
                                cout << "Stop" << endl;
                return 0;
            }
            else
            {
                cout << "Flip␣and␣wait" << endl;
            }
        }
    }
}
```

Task C ()

```
#define _CRT_SECURE_NO_WARNINGS

#include <iostream>
#include <algorithm>
#include <string>

using namespace std;

#define endl '\n'

const int Base = 200;

int n;

int main()
{
#ifdef _DEBUG
        freopen("input.txt", "r", stdin);
#endif // _DEBUG

        ios_base::sync_with_stdio(0);
        cin.tie(0);

        cin >> n;

        string s;
        for (int j = 0; j < n; j++)
        {
                cin >> s;
                int m = s.size();
                int type = 1;
                for (int i = 0; i < m; i++)
                {
                        if (s[i] == '?')
                        {
                                type = 2;
                                break;
                        }
                }

                if (type == 1)
                {
                        int cnt1 = 0, cnt0 = 0;
                        for (int i = 0; i < m; i++)
                        {
                                if (s[i] == '0')
                                {
                                        cnt0++;
                                }
                                else
                                {
                                        cnt1++;
                                }
                        }
                        int div = 0;
                        if (cnt1 > cnt0)
                        {
                                div = 1;
                        }
                        else if (cnt1 == cnt0)
                        {
                                div = 1 - (s[0] - '0');
                        }

                        bool fl = false;
                        for (int i = 0; i < m; i++)
                        {
                                if (!fl && s[i] == ('0' + div))
                                {
                                        fl = true;
                                        div = 1 - div;
                                }
                                else if(s[i] != ('0' + div))
                                {
                                        s[i] = '?';
                                }
                        }
                }
                else
                {
                        int div = 2;
                        for (int i = 0; i < m; i++)
                        {
                                if (s[i] != '?')
                                {
                                        div = 1 - (s[i] - '0');
                                        break;
                                }
                        }

                        bool fl = false;
                        for (int i = 0; i < m; i++)
                        {
                                if (!fl && s[i] != '?')
                                {
                                        div = 1 - div;
                                        fl = true;
                                }
                                else if(s[i] == '?')
                                {
                                        s[i] = '0' + div;
                                }
                        }
                }

                cout << s << endl;
        }
}
```

Task D ()

```
#define _CRT_SECURE_NO_WARNINGS

#include <iostream>
#include <algorithm>
#include <string>

using namespace std;

#define endl '\n'

const int Base = 150;
const long long Mod = 998244353;

int n;
long long F[2];

int main()
{
#ifdef _DEBUG
        freopen("input.txt", "r", stdin);
#endif // _DEBUG

        ios_base::sync_with_stdio(0);
        cin.tie(0);

        cin >> n;
        F[0] = 1;
        F[1] = 2;

        int last = 1;
        for (long long i = 2; i <= n; i++)
        {
                F[1 - last] = ((F[1 - last] * ((i * i) % Mod)) % Mod + F[last]) % Mod;
                last = 1 - last;
                if (F[last] == 0)
                {
                        cout << i << endl;
                }
        }

        cout << 0 << '␣' << F[last] << endl;

        return 0;
}
```

Task E ()

```
#define _CRT_SECURE_NO_WARNINGS

#include <iostream>
#include <algorithm>
#include <string>

using namespace std;

#define endl '\n'

const int Base = 3e5 + 99;
const long long Mod = 998244353;

int n;
int v[Base];
int q;

int main()
{
#ifdef _DEBUG
        freopen("input.txt", "r", stdin);
#endif // _DEBUG

        ios_base::sync_with_stdio(0);
        cin.tie(0);

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

        cin >> q;
        while (q--)
        {
                int l, r, cnt;
                cin >> l >> r >> cnt;
                long long ans = 0;
                for (int i = l; i < r; i++)
                {
                        ans += (cnt + v[i] - 1) / v[i];
                }
                cout << ans << endl;
        }

        return 0;
}
```

Task F ()

```
#define _CRT_SECURE_NO_WARNINGS

#include <iostream>
#include <algorithm>
#include <string>

using namespace std;

#define endl '\n'

const int Base = 1e5;
const long long Mod = 998244353;

int n;
long long L, R;
pair<long long, long long> segment[Base];
int ans[Base];

int main()
{
#ifdef _DEBUG
        freopen("input.txt", "r", stdin);
#endif // _DEBUG

        ios_base::sync_with_stdio(0);
        cin.tie(0);

        cin >> n;
        for (int i = 0; i < n; i++)
        {
                cin >> segment[i].first >> segment[i].second;
        }
        ans[n] = 1;
        for (int st = n - 1; st >= 0; st--)
        {
                L = 0;
                R = 0;
                ans[st] = 1;
                for (int i = st; i < n; i++)
                {
                        L += segment[i].first;
                        R += segment[i].second;
                        if (L <= 0 && 0 <= R)
                        {
                                ans[st] = max(ans[st], 1 + ans[i + 1]);
                        }
                }
        }

        int maxAns = 0;
        for (int i = 0; i < n; i++)
        {
                maxAns = max(maxAns, ans[i]);
        }

        cout << maxAns << endl;

        return 0;
}
```