

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;

using ll = long long;

void f1(void)
{
    int n;
    cin >> n;
    vector<pair<int, ll>> v(n);
    for(int i = 0; i < n; i++)
    {
        ll a;
        int b;
        cin >> a >> b;
        while(a > 0 && a % 10 == 0)
        {
            a /= 10;
            b++;
        }
        v[i] = make_pair(b, a);
    }
    sort(v.begin(), v.end());
    ll A = v[0].second;
    ll ans = v[0].first;
    for(int i = 1; i < n; i++)
    {
        if(v[i].first != v[i - 1].first)
        {
            ll t = A;
            int cnt = 0;
            while(t % 10 == 0)
            {
                t /= 10;
                cnt++;
            }
            if(cnt < v[i].first - v[i - 1].first)
            {
                cout << ans + cnt << endl;
                return;
            }
            else
            {
                for(int j = 0; j < v[i].first - v[i - 1].first; j++)
                {
                    A /= 10;
                    ans++;
                }
            }
        }
        A += v[i].second;
    }
    while(A % 10 == 0)
    {
        A /= 10;
```

```
        ans++;  
    }  
    cout << ans << endl;  
}  
  
int main()  
{  
    fl();  
    return 0;  
}
```

Task B ()

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

using namespace std;

int main()
{
    int n;
    cin >> n;
    vector<int> k(n);
    for (int i = 0; i < n; i++)
        cin >> k[i];
    while (k[k.size() - 1] == 0)
        k.pop_back();
    int cur = k.size() - 1;
    int act = 0;
    cout << "Flip_and_wait" << endl;
    while (1)
    {
        string s;
        cin >> s;
        int cnt = count(s.begin(), s.end(), 'e');
        if (cnt == 0 || cnt % 2 == 1)
            return 0;
        cnt >>= 1;
        act += cnt;
        if (act >= cur + 1)
        {
            act = 0;
            k[cur]--;
            while (cur >= 0 && k[cur] == 0)
                cur--;
            if (cur == -1)
            {
                cout << "Stop" << endl;
                return 0;
            }
            cout << "Flip_and_wait" << endl;
            continue;
        }
        else
            cout << "Wait" << endl;
    }
    return 0;
}
```

Task C ()

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

using namespace std;

int main()
{
    int n;
    cin >> n;
    vector<string> v(n);
    for(int i = 0; i < n; i++)
        cin >> v[i];
    string s = v[0];
    int c = count(s.begin(), s.end(), '?');
    bool md = (c != 0);
    for(int i = 0; i < n; i++)
    {
        if(!md)
        {
            int t = 0;
            if(v[i].size() % 2 != 0)
                t++;
            while(t < v[i].size())
            {
                if(v[i].substr(t, 2) == "00")
                    v[i][t] = '?';
                else if(v[i].substr(t, 2) == "01")
                    v[i][t] = '?';
                else if(v[i].substr(t, 2) == "10")
                    v[i][t + 1] = '?';
                else
                    v[i][t] = v[i][t + 1] = '?';
                t += 2;
            }
        }
        else
        {
            int t = 0;
            if(v[i].size() % 2 != 0)
                t++;
            while(t < v[i].size())
            {
                if(v[i].substr(t, 2) == "?0")
                    v[i][t] = '0';
                else if(v[i].substr(t, 2) == "?1")
                    v[i][t] = '0';
                else if(v[i].substr(t, 2) == "1?")
                    v[i][t + 1] = '0';
                else
                    v[i][t] = v[i][t + 1] = '1';
                t += 2;
            }
        }
    }
    for(int i = 0; i < n; i++)
        cout << v[i] << endl;
    return 0;
}
```

Task D ()

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

using namespace std;

using ll = long long;

#define MOD 998244353

#define FACT 998244352

vector<ll> fmem(100);

void init(void)
{
    fmem[0] = 1;
    fmem[1] = 295201906;
    fmem[2] = 160030060;
    fmem[3] = 957629942;
    fmem[4] = 545208507;
    fmem[5] = 213689172;
    fmem[6] = 760025067;
    fmem[7] = 939830261;
    fmem[8] = 506268060;
    fmem[9] = 39806322;
    fmem[10] = 808258749;
    fmem[11] = 440133909;
    fmem[12] = 686156489;
    fmem[13] = 741797144;
    fmem[14] = 390377694;
    fmem[15] = 12629586;
    fmem[16] = 544711799;
    fmem[17] = 104121967;
    fmem[18] = 495867250;
    fmem[19] = 421290700;
    fmem[20] = 117153405;
    fmem[21] = 57084755;
    fmem[22] = 202713771;
    fmem[23] = 675932866;
    fmem[24] = 79781699;
    fmem[25] = 956276337;
    fmem[26] = 652678397;
    fmem[27] = 35212756;
    fmem[28] = 655645460;
    fmem[29] = 468129309;
    fmem[30] = 761699708;
    fmem[31] = 533047427;
    fmem[32] = 287671032;
    fmem[33] = 206068022;
    fmem[34] = 50865043;
    fmem[35] = 144980423;
    fmem[36] = 111276893;
    fmem[37] = 259415897;
    fmem[38] = 444094191;
    fmem[39] = 593907889;
    fmem[40] = 573994984;
    fmem[41] = 892454686;
    fmem[42] = 566073550;
    fmem[43] = 128761001;
    fmem[44] = 888483202;
    fmem[45] = 251718753;
    fmem[46] = 548033568;
    fmem[47] = 428105027;
    fmem[48] = 742756734;
    fmem[49] = 546182474;
    fmem[50] = 62402409;
    fmem[51] = 102052166;
    fmem[52] = 826426395;
    fmem[53] = 159186619;
    fmem[54] = 926316039;
    fmem[55] = 176055335;
    fmem[56] = 51568171;
    fmem[57] = 414163604;
```

```

fmem[58] = 604947226;
fmem[59] = 681666415;
fmem[60] = 511621808;
fmem[61] = 924112080;
fmem[62] = 265769800;
fmem[63] = 955559118;
fmem[64] = 763148293;
fmem[65] = 472709375;
fmem[66] = 19536133;
fmem[67] = 860830935;
fmem[68] = 290471030;
fmem[69] = 851685235;
fmem[70] = 242726978;
fmem[71] = 169855231;
fmem[72] = 612759169;
fmem[73] = 599797734;
fmem[74] = 961628039;
fmem[75] = 953297493;
fmem[76] = 62806842;
fmem[77] = 37844313;
fmem[78] = 909741023;
fmem[79] = 689361523;
fmem[80] = 887890124;
fmem[81] = 380694152;
fmem[82] = 669317759;
fmem[83] = 367270918;
fmem[84] = 806951470;
fmem[85] = 843736533;
fmem[86] = 377403437;
fmem[87] = 945260111;
fmem[88] = 786127243;
fmem[89] = 80918046;
fmem[90] = 875880304;
fmem[91] = 364983542;
fmem[92] = 623250998;
fmem[93] = 598764068;
fmem[94] = 804930040;
fmem[95] = 24257676;
fmem[96] = 214821357;
fmem[97] = 791011898;
fmem[98] = 954947696;
fmem[99] = 183092975;
}

void f2(void)
{
    ifstream fin("input.txt");
    ll n;
    fin >> n;
    ll fact = 1;
    for (ll i = 1; i <= n + 1; i++)
    {
        if (i % MOD == 0)
            continue;
        fact = (fact * i) % MOD;
    }
    cout << fact << endl;
}

void f1(void)
{
    //ifstream fin("input.txt");
    ll n;
    cin >> n;
    ll zer = (n + 1) / MOD;
    if (n + 1 >= 11(MOD) * 11(MOD))
        zer++;
    cout << zer << "\n";
    if (zer * MOD == n + 1)
        zer--;
    if (n + 1 >= 11(MOD) * 11(MOD))
        zer--;
    ll rem = n + 1 - zer * MOD;
    init();
}

```

```

int b = rem / 10000000;
ll ans = fmem[b];
for(int i = 1 + (b * 10000000); i <= rem; i++)
    ans = (ans * i) % MOD;
if(zer % 2 == 0)
    cout << ans << endl;
else
    cout << MOD - ans << endl;
}

int main()
{
    f1();
    //456690899
    return 0;
}

```

Task E ()

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

using namespace std;

using ll = long long;

ll my_div(int a, int b)
{
    int t = a / b;
    if(b * t < a)
        t++;
    return t;
}

int main()
{
    int n, q;
    cin >> n;
    vector<int> v(n);
    for(int i = 0; i < n; i++)
        cin >> v[i];
    cin >> q;
    while(q--)
    {
        int a, b, d;
        cin >> a >> b >> d;
        ll ans = 0;
        for(int i = a; i < b; i++)
            ans += my_div(d, v[i]);
        cout << ans << endl;
    }
    return 0;
}
```


Task F ()