Олимпиада СПбГУ по информатике 2020/21 учебного года

A	B	C	D	E	F	Sum
100	100	100	100	55	25	480

Task A ()

```
#pragma GCC optimize("Ofast,no-stack-protector")
#pragma GCC optimize("unroll-loops")
#pragma GCC optimize("fast-math")

#include <iostream>
#include <cmath>
#include <vector>
#include <set>
#include <algorithm>
#include <map>
#include <iomanip>
#include <queue>

using namespace std;

#define ll long long
#define ld long double
#define S second
#define F first
#define all(x) x.begin(), x.end()


ll a, b, c, d, e, f, st;

vector <vector <ll> > arr;




int main()
{
    ios_base::sync_with_stdio(0);
    cin.tie(0);
    cout.tie(0);
    cin >> a;
    if (a <= 10)
    {
       cout << a % 10;
        return 0;
    }
    else
    {
        a -= 10;
        --a;
        string s = "234567890";
        cout << s[a % (ll)s.length()];
    }
}
```

Task B ()

```
#pragma GCC optimize("Ofast,no-stack-protector")
#pragma GCC optimize("unroll-loops")
#pragma GCC optimize("fast-math")

#include <iostream>
#include <cmath>
#include <vector>
#include <set>
#include <algorithm>
#include <map>
#include <iomanip>
#include <queue>

using namespace std;

#define ll long long
#define ld long double
#define S second
#define F first
#define all(x) x.begin(), x.end()


ll a, b, c, d, e, f, st;

vector <vector <ll> > arr;




int main()
{
    ios_base::sync_with_stdio(0);
    cin.tie(0);
    cout.tie(0);
    cin >> a >> b;
    string s;
    cin >> s;
    ll cur_cnt = 0;
    ll cnt_b = 0;
    vector <ll> vis(26);
    ll ans = 0;
    for (ll i = 0; i < a; ++i)
    {
        if (vis[s[i] - 'a'])
        {
           ++cur_cnt;
            if (cur_cnt == b)
            {
               ans++;
               cur_cnt = 0;
               for (ll j = 0; j < 26; ++j)
                     vis[j] = 0;
               cnt_b = 0;
            }
            continue;
        }
        else
        {
            if (cnt_b < 3)
            {
                ++cnt_b;
                vis[s[i] - 'a'] = 1;
                ++cur_cnt;
                if (cur_cnt == b)
                {
                   ans++;
                   cur_cnt = 0;
                   for (ll j = 0; j < 26; ++j)
                         vis[j] = 0;
                   cnt_b = 0;
                }
            }
            else
            {
                ++ans;
                cnt_b = 1;
                cur_cnt = 1;
                for (ll j = 0; j < 26; ++j)
                    vis[j] = 0;
                vis[s[i] - 'a'] = 1;
                if (cur_cnt == b)
                {
                   ans++;
                   cur_cnt = 0;
                   for (ll j = 0; j < 26; ++j)
                        vis[j] = 0;
                   cnt_b = 0;
                }
            }
        }
    }
    if (cnt_b != 0)
        ++ans;
    cout << ans;
}
```

Task C ()

```
#pragma GCC optimize("Ofast,no-stack-protector")
#pragma GCC optimize("unroll-loops")
#pragma GCC optimize("fast-math")

#include <iostream>
#include <cmath>
#include <vector>
#include <set>
#include <algorithm>
#include <map>
#include <iomanip>
#include <queue>

using namespace std;

#define ll int
#define ld long double
#define S second
#define F first
#define all(x) x.begin(), x.end()


ll a, b, c, d, e, f, st;

vector <vector <ll> > arr;
vector <char> ans;




int main()
{
    ios_base::sync_with_stdio(0);
    cin.tie(0);
    cout.tie(0);
    cin >> a >> b >> c;
    ans.resize(a);
    vector <pair <pair <ll, ll>, ll> > pp(a);
    for (ll i = 0; i < a; ++i)
        cin >> pp[i].F.F, pp[i].S = i;
    for (ll i = 0; i < a; ++i)
        cin >> pp[i].F.S;
    //vector <vector <ll> > dp(a + 1, vector <ll> (b + 1, -1));
    vector <ll> dp(b + 1, -1);
    vector <ll> dp1(b + 1, -1);
    vector <vector <short int> > pr(a + 1, vector <short int> (b + 1, -1));
    dp[0] = 0;
    for (ll i = 1; i <= a; ++i)
    {
        dp1 = dp;
        pr[i] = pr[i - 1];
        for (ll j = b; j >= 0; --j)
        {
            if (j - pp[i - 1].F.F < 0)
                break;
            if (dp[j - pp[i - 1].F.F] == -1)
                continue;
            if (dp[j - pp[i - 1].F.F] + pp[i - 1].F.S > dp1[j])
            {
                dp1[j] = max(dp[j - pp[i - 1].F.F] + pp[i - 1].F.S, dp1[j]);
                pr[i][j] = i - 1;
            }

        }
        dp = dp1;
    }
    ll sum = 0;
    for (auto v : pp)
        sum += v.F.S;
    ll pos = -1;
    for (ll i = 0; i <= b; ++i)
    {
        if (dp[i] == -1)
            continue;
        if (sum - dp[i] <= c)
        {
            pos = i;
            break;
        }
    }
    if (pos == -1)
    {
        cout << -1;
        return 0;
    }
    vector <ll> tt;
    for (ll i = a; i > 0; --i)
    {
        if (i == pr[i][pos] + 1)
        {
            tt.push_back(pr[i][pos]);
            pos = pos - pp[pr[i][pos]].F.F;
        }
    }
    ans.assign(a, 'y');
    for (ll i = 0; i < tt.size(); ++i)
        ans[tt[i]] = 'x';
    for (auto v : ans)
        cout << v;
}
```

Task D ()

```
#pragma GCC optimize("Ofast,no-stack-protector")
#pragma GCC optimize("unroll-loops")
#pragma GCC optimize("fast-math")

#include <iostream>
#include <cmath>
#include <vector>
#include <set>
#include <algorithm>
#include <map>
#include <iomanip>
#include <queue>

using namespace std;

#define ll long long
#define ld long double
#define S second
#define F first
#define all(x) x.begin(), x.end()


ll a, b, c, d, e, f, st;

vector <vector <ll> > arr;

ll k = 124113;


int main()
{
    ios_base::sync_with_stdio(0);
    cin.tie(0);
    cout.tie(0);
    cin >> a;
    a *= 2;
    string s;
    cin >> s;
    ll cnt = 0;
    ll cnt1 = 0;
    for (ll i = 0; i < s.length(); ++i)
    {
        if (s[i] == '(' || s[i] == ')')
        {
            if (i % 2)
                ++cnt;
            else
                ++cnt1;
        }
    }
    cout << abs(cnt - cnt1);
}
```

Task E ()

```
#pragma GCC optimize("Ofast,no-stack-protector")
#pragma GCC optimize("unroll-loops")
#pragma GCC optimize("fast-math")

#include <iostream>
#include <cmath>
#include <vector>
#include <set>
#include <algorithm>
#include <map>
#include <iomanip>
#include <queue>

using namespace std;

#define ll long long
#define ld long double
#define S second
#define F first
#define all(x) x.begin(), x.end()


ll a, b, c, d, e, f, st;

vector <vector <ll> > arr;

ll k = 124113;
ll k1 = (30 * rand() + rand() * 40 + rand() * 40  + rand() * 18  + rand() * 1) % (ll)1e5;
ll k2 = (30 * rand() + rand() * 40 + rand() * 40  + rand() * 18  + rand() * 1) % (ll)1e5;
ll k3 = (30 * rand() + rand() * 40 + rand() * 40  + rand() * 18  + rand() * 1) % (ll)1e5;
ll k4 = (30 * rand() + rand() * 40 + rand() * 40  + rand() * 18  + rand() * 1) % (ll)1e5;

vector <ll> gg = {10, 3, 8, 5, 3, 7, 7, 7, 8, 2, 8, 5, 7, 5, 9, 9, 10, 9, 2, 2, 3, 8, 4, 10, 2, 4,
     3, 2, 8, 5, 10, 5, 3, 7, 3, 2, 5, 7, 9, 8, 5, 5, 6, 7, 1, 8, 10, 5, 3, 10, 8, 6, 7, 1, 10, 7,
     10, 8, 9, 6, 5, 1, 5, 3, 9, 10, 9, 5, 2, 8, 9, 10, 6, 10, 8, 9, 4, 8, 5, 6, 8, 6, 10, 6, 6,
    2, 6, 9, 7, 8, 8, 1, 1, 9, 1, 2, 6, 2, 9, 5, 2, 10, 4, 7, 10, 10, 4, 3, 9, 1, 10, 4, 1, 7, 1,
    5, 10, 3, 2, 2};

map <vector <ll>, ll> map1;


int main()
{
    ios_base::sync_with_stdio(0);
    cin.tie(0);
    cout.tie(0);
    string s;
    cin >> s;
    ll cnt = 0;
    for (ll i = 1; i <= 10; ++i)
    {
        for (ll j = i + 1; j <= 10; ++j)
        {
            for (ll k = j + 1; k <= 10; ++k)
            {
                vector <ll> cur = {i, j, k};
                map1[cur] = cnt;
                ++cnt;
            }
        }
    }
    if (s == "add")
    {
        ll t;
        cin >> t;
        while(t--)
        {
            cin >> a >> b;
            if (a == 1e6)
            {
                for (ll i = 0; i < b; ++i)
                    cin >> c;
                cout << k << endl;
                continue;
            }
            if (a == 10)
            {
                vector <ll> pp(b);
                for (ll i = 0; i < b; ++i)
                {
                    cin >> pp[i];
                }
                sort(all(pp));
                cout << gg[map1[pp]] << endl;
                continue;
            }
            if (a == 1e5)
            {
                bool fl = true;
                bool fl1 = true;
                bool fl2 = true;
                bool fl3 = true;
                for (ll i = 0; i < b; ++i)
                {
                    cin >> c;
                    if (c == k1)
                        fl = false;
                    if (c == k2)
                        fl1 = false;
                    if (c == k3)
                        fl2 = false;
                    if (c == k4)
                        fl3 = false;
                }
                if (fl && fl1 && fl2)
                    cout << k1 << endl;
                else if ((!fl1 || !fl) && fl2)
                    cout << k3 << endl;
                else if (!fl2)
                {
                    if (!fl3)
                        cout << 1 / 0;
                    cout << k4 << endl;
                }
            }
        }
        return 0;
    }
    ll t;
    cin >> t;
    while(t--)
    {
        cin >> a >> b;
        if (a == 1e6)
        {
            for (ll i = 0; i < b + 1; ++i)
            {
                cin >> c;
                if (c != k)
                    cout << c << "␣";
            }
            cout << endl;
            continue;
        }
        if (a == 10)
        {
            vector <ll> pp(b + 1);
            for (ll i = 0; i < b + 1; ++i)
            {
                cin >> pp[i];
            }
            sort(all(pp));
            bool fl = false;
            for (ll i = 1; i <= 24; ++i)
            {
                ll k = pp.back();
```

```
                pp.pop_back();
                vector <ll> pp1 = pp;
                sort(all(pp1));
                if (gg[map1[pp1]] == k)
                {
                    for (ll j = 0; j < 3; ++j)
                        cout << pp[j] << "␣";
                    cout << endl;
                    fl = true;
                    break;
                }
                pp.push_back(k);
                next_permutation(all(pp));
            }
            if (!fl)
                cout << 1 / 0;
            continue;
        }
        if (a == 1e5)
        {
            bool fl = false;
            bool fl1 = false;
            bool fl2 = false;
            bool fl3 = false;
            vector <ll> pp(b + 1);
            for (ll i = 0; i < b + 1; ++i)
            {
                cin >> pp[i];
                c = pp[i];
                if (c == k1)
                    fl = true;
                if (c == k2)
                    fl1 = true;
                if (c == k3)
                    fl2 = true;
                if (c == k4)
                    fl3 = true;
            }
            for (ll i = 0; i < b + 1; ++i)
            {
                if (fl && !fl2 && !fl1)
                {
                    if (pp[i] != k1)
                        cout << pp[i] << "␣";
                }
                else if (fl2 && !fl3)
                {
                    if (pp[i] != k3)
                        cout << pp[i] << "␣";
                }
                else
                {
                    if (pp[i] != k4)
                        cout << pp[i] << "␣";
                }
            }
            cout << endl;
        }
    }
}
```

Task F ()

```
#pragma GCC optimize("Ofast,no-stack-protector")
#pragma GCC optimize("unroll-loops")
#pragma GCC optimize("fast-math")

#include <iostream>
#include <cmath>
#include <vector>
#include <set>
#include <algorithm>
#include <map>
#include <iomanip>
#include <queue>

using namespace std;

#define ll long long
#define ld long double
#define S second
#define F first
#define all(x) x.begin(), x.end()


ll a, b, c, d, e, f, st;

vector <vector <ll> > arr;
vector <char> ans;

void solve1()
{
    vector <pair <ll, ll> > pp = {{0, 0}, {0, 1}, {1, 1}, {1, 0}};
    vector <pair <ll, ll> > ans = {{-1, 0}, {0, -1}, {0, 1}, {1, 0}, {1, 1}, {-1, -1}, {-1, 1},
        {1, -1}};
    cout << 4 << endl;
    for (auto v : pp)
        cout << v.F << "␣" << v.S << endl;
    for (ll i = 0; i < a; ++i)
    {
        cout << ans[i].F << "␣" << ans[i].S << endl;
    }
}


int main()
{
    ios_base::sync_with_stdio(0);
    cin.tie(0);
    cout.tie(0);
    cin >> a;
    if (a <= 8)
    {
        solve1();
        return 0;
    }
}
```