Олимпиада СПбГУ по информатике 2022/23 учебного года

A	B	C	D	E	F	Sum
100	100	100	100	52	10	462

Task A ()

```cpp
#include <bits/stdc++.h>
///#include <ext/pb_ds/assoc_container.hpp>

#define ibase ios_base::sync_with_stdio(0);cin.tie(0);cout.tie(0);
#define fix6 fixed << setprecision(6)
#define all(x) (x).begin(), (x).end()
#define fi first
#define se second

/***/#pragma GCC optimize("Ofast,O3")/***/
//***/#pragma GCC target("sse,sse2,sse3,sse4,ssse3,popcnt,abm,mmx,tune=native")/***/

using namespace std;
///using namespace __gnu_pbds;
using ll = long long;
using ld = long double;
using pii = pair<int, int>;

signed main(){
    ibase;
    int n = 6;
    vector<int> d(n, -1);
    for(int i = 0; i < n; ++i){
        int x;
        cin >> x;
        --x;
        for(int j = n - 2; j >= x; --j){
            d[j + 1] = d[j];
        }
        d[x] = i;
    }
    vector<int> pos(n, 0);
    for(int i = 0; i < n; ++i){
        pos[d[i]] = i;
    }
    for(auto w : pos){
        cout << w + 1 << ' ';
    }
}
```

Task B ()

```
#include <bits/stdc++.h>
///#include <ext/pb_ds/assoc_container.hpp>

#define ibase ios_base::sync_with_stdio(0);cin.tie(0);cout.tie(0);
#define fix6 fixed << setprecision(6)
#define all(x) (x).begin(), (x).end()
#define fi first
#define se second

/***/#pragma GCC optimize("Ofast,O3")/***/
//***/#pragma GCC target("sse,sse2,sse3,sse4,ssse3,popcnt,abm,mmx,tune=native")/***/

using namespace std;
///using namespace __gnu_pbds;
using ll = long long;
using ld = long double;
using pii = pair<int, int>;

signed main(){
    ibase;
    string s;
    cin >> s;
    if(s == "first"){
        int n;
        cin >> n;
        vector<int> a(n);
        ll sum = 0;
        for(auto& x : a){
            cin >> x;
            sum += x;
        }
        cout << 101 * sum;
    }
    else{
        int n;
        cin >> n;
        vector<int> b(n);
        ll sum = 0;
        for(auto& x : b){
            cin >> x;
            sum += (x%101);
        }
        //cout << sum << ' ';
        cout << sum + b[0]/101;
    }
}
```

Task C ()

```
#include <bits/stdc++.h>
///#include <ext/pb_ds/assoc_container.hpp>

#define ibase ios_base::sync_with_stdio(0);cin.tie(0);cout.tie(0);
#define fix6 fixed << setprecision(6)
#define all(x) (x).begin(), (x).end()
#define fi first
#define se second

/***/#pragma GCC optimize("Ofast,O3")/***/
//***/#pragma GCC target("sse,sse2,sse3,sse4,ssse3,popcnt,abm,mmx,tune=native")/***/

using namespace std;
///using namespace __gnu_pbds;
using ll = long long;
using ld = long double;
using pii = pair<int, int>;

int const N = 35;
set<pii> st;

void ins(int a, int b){
    st.insert({min(a, b), max(a, b)});
    return;
}

void aboba4(int n, int m, vector < vector <int> > p){
//    for(int i = 0; i < n; ++i){
//        for(int j = 0; j < m; ++j){
//            cout << p[i][j] << ' ';
//        }
//        cout << endl;
//    }
//    cout << endl;
    for(int i = 0; i < n; ++i){
        for(int j = 0; j < m; ++j){
            for(int x = i; x < n; ++x){
                for(int y = j; y < m; ++y){
                    bool f = 1;
                    for(int _i = i; _i <= x; ++_i){
                        for(int _j = j; _j <= y; ++_j){
                            if(p[_i][_j] != 0){
                                f = 0;
                                break;
                            }
                        }
                        if(!f){ break;}
                    }
                    if(!f){ continue;}
                    for(int _i = i; _i <= x; ++_i){
                        if(!(j == 0 or p[_i][j - 1] != 0)){
                            f = 0; break;
                        }
                        if(!(y == m - 1 or p[_i][y + 1] != 0)){
                            f = 0; break;
                        }
                    }
                    for(int _j = j; _j <= y; ++_j){
                        if(!(i == 0 or p[i - 1][_j] != 0)){
                            f = 0; break;
                        }
                        if(!(x == n - 1 or p[x + 1][_j] != 0)){
                            f = 0; break;
                        }
                    }
                    if(f){
                        ins(x - i + 1, y - j + 1);
//                          if(x - i + 1 == 1 and y - j + 1 == 2){
//                              for(int i = 0; i < n; ++i){
//                                  for(int j = 0; j < m; ++j){
//                                      cout << p[i][j] << ' ';
//                                  }
```

```
//                                  cout << endl;
//                              }
//                              cout << endl;
//                          }
                        }
                    }
                }
            }
        }
    return;
}

void aboba3(int n, int m, vector < vector <int> > p){
    bool f1s = 0, f1u = 0, f2s = 0, f2u = 0;
    for(int i = 0; i < n; ++i){
        for(int j = 0; j < m; ++j){
            if(p[i][j] == 1){
                if((i + 1 == n or (i + 1 != n and p[i + 1][j] == 2)) or (i == 0 or (i != 0 and p[i
                    - 1][j] == 2))){
                    f1u = 1;
                }
                if((j + 1 == m or (j + 1 != m and p[i][j + 1] == 2)) or (j == 0 or (j != 0 and p[i
                    ][j - 1] == 2))){
                    f1s = 1;
                }
            }
            else if(p[i][j] == 2){
                if((i + 1 == n or (i + 1 != n and p[i + 1][j] == 1)) or (i == 0 or (i != 0 and p[i
                    - 1][j] == 1))){
                    f2u = 1;
                }
                if((j + 1 == m or (j + 1 != m and p[i][j + 1] == 1)) or (j == 0 or (j != 0 and p[i
                    ][j - 1] == 1))){
                    f2s = 1;
                }
            }
        }
    }
    if(f1s and f1u and f2s and f2u){
        aboba4(n, m, p);
    }
    return;
}

void _aboba(int n, int m, vector < vector <int> > p){
    bool f1s = 0, f1u = 0;
    for(int i = 0; i < n; ++i){
        for(int j = 0; j < m; ++j){
            if(p[i][j] == 1){
                if(i + 1 == n or i == 0){
                    f1u = 1;
                }
                if(j + 1 == m or j == 0){
                    f1s = 1;
                }
            }
        }
    }
    if(f1s and f1u){
        aboba4(n, m, p);
    }
}

void aboba2(int n, int m, int a, int b, vector < vector <int> > p){
    _aboba(n, m, p);
    for(int i = 0; i < n; ++i){
        for(int j = 0; j < m; ++j){
            if(i + a - 1 >= n or j + b - 1 >= m){ continue;}
            bool f = 1;
            auto _p = p;
            for(int _i = i; _i < i + a; ++_i){
                for(int _j = j; _j < j + b; ++_j){
                    if(p[_i][_j] == 1){
                        f = 0;
```

```
                        break;
                    }
                    p[_i][_j] = 2;
                }
                if(!f){ break;}
            }
            if(f){
                aboba3(n, m, p);
            }
            p = _p;
        }
    }
    return;
}

void aboba(int n, int m, int a1, int b1, int a2, int b2){
    for(int i = 0; i < n; ++i){
        for(int j = 0; j < m; ++j){
            if(i + a1 - 1 >= n or j + b1 - 1 >= m){ continue;}
            vector < vector <int> > p(n, vector <int>(m, 0));
            for(int _i = i; _i < i + a1; ++_i){
                for(int _j = j; _j < j + b1; ++_j){
                    p[_i][_j] = 1;
                }
            }
            aboba2(n, m, a2, b2, p);
            return;
        }
    }
    return;
}

signed main(){
    ibase;
    int n = 3;
    vector <int> a(n), b(n);
    for(int i = 0; i < n; ++i){
        cin >> a[i] >> b[i];
        ins(a[i], b[i]);
    }
    for(int mask = 0; mask < (1 << n); ++mask){
        for(int i = 0; i < n; ++i){
            if(mask & (1 << i)){
                swap(a[i], b[i]);
            }
        }
        aboba(a[0], b[0], a[1], b[1], a[2], b[2]);
        aboba(a[0], b[0], a[2], b[2], a[1], b[1]);
        aboba(a[1], b[1], a[0], b[0], a[2], b[2]);
        aboba(a[1], b[1], a[2], b[2], a[0], b[0]);
        aboba(a[2], b[2], a[0], b[0], a[1], b[1]);
        aboba(a[2], b[2], a[1], b[1], a[0], b[0]);
        for(int i = 0; i < n; ++i){
            if(mask & (1 << i)){
                swap(a[i], b[i]);
            }
        }
    }
    for(auto w : st){
        cout << w.fi << ' ' << w.se << '\n';
    }
}
/**
8 8
2 3
2 3

5 3
2 3
1 1
**/
```

Task D ()

```
#include <bits/stdc++.h>
///#include <ext/pb_ds/assoc_container.hpp>

#define ibase ios_base::sync_with_stdio(0);cin.tie(0);cout.tie(0);
#define fix6 fixed << setprecision(6)
#define all(x) (x).begin(), (x).end()
#define fi first
#define se second

/***/#pragma GCC optimize("Ofast,O3")/***/
//***/#pragma GCC target("sse,sse2,sse3,sse4,ssse3,popcnt,abm,mmx,tune=native")/***/

using namespace std;
///using namespace __gnu_pbds;
using ll = long long;
using ld = long double;
using pii = pair<int, int>;

struct S{
    int a0, a1, mask;
};

struct _S{
    int a0, a1, a2, mask;
};

int const C = 51;
vector<S> dp[C][C][4];
bool is[C][C][4];
bool used[C][C][4];

vector<_S> _dp[C][C][C][8];
bool _is[C][C][C][8];
bool _used[C][C][C][8];

void dfs(S v, vector<int>& a){
    bool f = 0;
    used[v.a0][v.a1][v.mask] = 1;
    for(auto u : dp[v.a0][v.a1][v.mask]){
        if(!used[u.a0][u.a1][u.mask]){
            dfs(u, a);
        }
        if(!is[u.a0][u.a1][u.mask]){
            f = 1;
        }
    }
    is[v.a0][v.a1][v.mask] = f;
    //cout << v.a0 << ' ' << v.a1 << "  " << v.mask << "  " << f << endl;
    return;
}

void cal(int n, vector<int> a){
    for(int a0 = 0; a0 < C; ++a0){
        for(int a1 = 0; a1 < C; ++a1){
            for(int mask = 0; mask < (1 << n); ++mask){
                for(int x = 1; x <= a0; ++x){
                    dp[a0][a1][mask].push_back({a0 - x, a1, mask});
                }
                for(int x = 1; x <= a1; ++x){
                    dp[a0][a1][mask].push_back({a0, a1 - x, mask});
                }
                if(mask & (1 << 0)){
                    dp[a0][a1][mask].push_back({a[0], a1, mask - (1 << 0)});
                }
                if(mask & (1 << 1)){
                    dp[a0][a1][mask].push_back({a0, a[1], mask - (1 << 1)});
                }
            }
        }
    }

    is[0][0][0] = 0;
```

```
    dfs({a[0], a[1], 3}, a);
}

void _dfs(_S v, vector<int>& a){
    bool f = 0;
    int a0 = v.a0, a1 = v.a1, a2 = v.a2, mask = v.mask;
    _used[v.a0][v.a1][v.a2][v.mask] = 1;

    for(int x = 1; x <= a0; ++x){
        if(!_used[a0 - x][a1][a2][mask]){
            _dfs({a0 - x, a1, a2, mask}, a);
        }
        if(!_is[a0 - x][a1][a2][mask]){
            f = 1;
        }
    }
    for(int x = 1; x <= a1; ++x){
        if(!_used[a0][a1 - x][a2][mask]){
            _dfs({a0, a1 - x, a2, mask}, a);
        }
        if(!_is[a0][a1 - x][a2][mask]){
            f = 1;
        }
    }
    for(int x = 1; x <= a2; ++x){
        if(!_used[a0][a1][a2 - x][mask]){
            _dfs({a0, a1, a2 - x, mask}, a);
        }
        if(!_is[a0][a1][a2 - x][mask]){
            f = 1;
        }
    }
    if(mask & (1 << 0)){
        if(!_used[a[0]][a1][a2][mask - (1 << 0)]){
            _dfs({a[0], a1, a2, mask - (1 << 0)}, a);
        }
        if(!_is[a[0]][a1][a2][mask - (1 << 0)]){
            f = 1;
        }
    }
    if(mask & (1 << 1)){
        if(!_used[a0][a[1]][a2][mask - (1 << 1)]){
            _dfs({a0, a[1], a2, mask - (1 << 1)}, a);
        }
        if(!_is[a0][a[1]][a2][mask - (1 << 1)]){
            f = 1;
        }
    }
    if(mask & (1 << 2)){
        if(!_used[a0][a1][a[2]][mask - (1 << 2)]){
            _dfs({a0, a1, a[2], mask - (1 << 2)}, a);
        }
        if(!_is[a0][a1][a[2]][mask - (1 << 2)]){
            f = 1;
        }
    }

//    for(auto u : _dp[v.a0][v.a1][v.a2][v.mask]){
//        if(!_used[u.a0][u.a1][u.a2][u.mask]){
//            _dfs(u, a);
//        }
//        if(!_is[u.a0][u.a1][u.a2][u.mask]){
//            f = 1;
//        }
//    }
    _is[v.a0][v.a1][v.a2][v.mask] = f;
    return;
}

void _cal(int n, vector<int> a){
    _is[0][0][0][0] = 0;

    _dfs({a[0], a[1], a[2], 7}, a);
```

```
}

signed main(){
    ibase;
    int n;
    cin >> n;
    vector <int> a(n);
    for(auto& w : a){
        cin >> w;
    }
    auto _a = a;
    if(n == 1){
        cout << 1 << ' ' << a[0] << endl;
        int id, x;
        cin >> id >> x;
        if(id == -1){
            return 0;
        }
        else{
            cout << 1 << ' ' << a[0] << endl;
        }
        cin >> id >> x;
        return 0;
    }
    else if(n == 2){
        cal(n, a);
        if(is[a[0]][a[1]][3] == 0){
            cout << -1 << ' ' << -1 << endl;
            return 0;
        }
        //vector <bool> is(n, 0);
        int mask = 3;
        while(1){
            for(auto u : dp[a[0]][a[1]][mask]){
                if(is[u.a0][u.a1][u.mask] == 0){
                    if(u.mask != mask){
                        if(((1 << 0) & mask ) and !((1 << 0) & u.mask)){
                            cout << 1 << ' ' << 0 << endl;
                            a[0] = _a[0];
                        }
                        else{
                            cout << 2 << ' ' << 0 << endl;
                            a[1] = _a[1];
                        }
                        mask = u.mask;
                    }
                    else if(a[0] != u.a0){
                        cout << 1 << ' ' << a[0] - u.a0 << endl;
                        a[0] = u.a0;
                    }
                    else{
                        cout << 2 << ' ' << a[1] - u.a1 << endl;
                        a[1] = u.a1;
                    }
                    break;
                }
            }

            int id, x;
            cin >> id >> x;
            if(id == -1){
                break;
            }
            --id;
            if(x == 0){
                if(id == 0){
                    a[0] = _a[0];
                    mask -= (1 << 0);
                }
                else{
                    a[1] = _a[1];
                    mask -= (1 << 1);
                }
            }
```

```
            else{
                a[id] -= x;
            }
        }
    }
    else if(n == 3){
        _cal(n, a);
        if(_is[a[0]][a[1]][a[2]][7] == 0){
            cout << -1 << ' ' << -1 << endl;
            return 0;
        }
        int mask = 7;
        while(1){
            while(1){
                int a0 = a[0], a1 = a[1], a2 = a[2];
                bool f = 0;
                if(mask & (1 << 0)){
                    if(!_is[_a[0]][a1][a2][mask - (1 << 0)]){
                        cout << 1 << ' ' << 0 << endl;
                        a[0] = _a[0];
                        mask -= (1 << 0);
                        f = 1;
                        break;
                    }
                }
                if(f){ break;}
                if(mask & (1 << 1)){
                    if(!_is[a0][_a[1]][a2][mask - (1 << 1)]){
                        cout << 2 << ' ' << 0 << endl;
                        a[1] = _a[1];
                        mask -= (1 << 1);
                        f = 1;
                        break;
                    }
                }
                if(f){ break;}
                if(mask & (1 << 2)){
                    if(!_is[a0][a1][_a[2]][mask - (1 << 2)]){
                        cout << 3 << ' ' << 0 << endl;
                        a[2] = _a[2];
                        mask -= (1 << 2);
                        f = 1;
                        break;
                    }
                }
                if(f){ break;}
                for(int x = a0; x >= 1; --x){
                    if(!_is[a0 - x][a1][a2][mask]){
                        cout << 1 << ' ' << x << endl;
                        a[0] -= x;
                        f = 1;
                        break;
                    }
                }
                if(f){ break;}
                for(int x = a1; x >= 1; --x){
                    if(!_is[a0][a1 - x][a2][mask]){
                        cout << 2 << ' ' << x << endl;
                        a[1] -= x;
                        f = 1;
                        break;
                    }
                }
                if(f){ break;}
                for(int x = a2; x >= 1; --x){
                    if(!_is[a0][a1][a2 - x][mask]){
                        cout << 3 << ' ' << x << endl;
                        a[2] -= x;
                        f = 1;
                        break;
                    }
                }
                if(f){ break;}
                cout << -1 << ' ' << -1;
```

```
            return 0;
        }

        int id, x;
        cin >> id >> x;
        if(id == -1){
            break;
        }
        --id;
        if(x == 0){
            if(id == 0){
                a[0] = _a[0];
                mask -= (1 << 0);
            }
            else if (id == 1){
                a[1] = _a[1];
                mask -= (1 << 1);
            }
            else{
                a[2] = _a[2];
                mask -= (1 << 2);
            }
        }
        else{
            a[id] -= x;
        }
    }
  }
}
/**
8 8
2 3
2 3

5 3
2 3
1 1
**/
```

Task E ()

```
#include <bits/stdc++.h>
///#include <ext/pb_ds/assoc_container.hpp>

#define ibase ios_base::sync_with_stdio(0);cin.tie(0);cout.tie(0);
#define fix6 fixed << setprecision(6)
#define all(x) (x).begin(), (x).end()
#define fi first
#define se second

/***/#pragma GCC optimize("Ofast,O3")/***/
//***/#pragma GCC target("sse,sse2,sse3,sse4,ssse3,popcnt,abm,mmx,tune=native")/***/

using namespace std;
///using namespace __gnu_pbds;
using ll = long long;
using ld = long double;
using pii = pair<int, int>;

int const N = 10;

vector < vector <int> > al;

void gen(int i, vector <int>& a){
    if(i == N){
        al.push_back(a);
        return;
    }
    for(int x = (i == 0 ? 0 : a[i - 1]); x <= 10; ++x){
        a[i] = x;
        gen(i + 1, a);
    }
    return;
}

void gg(){
    vector <int> a(N);
    gen(0, a);
    return;
}

vector < vector <char> > get(int n){
//    vector <int> a(N);
//    for(int i = 0; i < N; ++i){
//        a[i] = min(10, n);
//        n -= a[i];
//    }
    vector <int> a = al[n];
    vector < vector <char> > v(N, vector <char>(N, '1'));
    for(int j = 0; j < N; ++j){
        for(int i = 0; i < a[j]; ++i){
            v[i][j] = '0';
        }
    }
    return v;
}

int deget(vector < vector <char> > v){
    vector < vector <char> > rv(N, vector <char>(N));
    vector <bool> is(N, 0);
    for(int id = N - 1; id >= 0; --id){
        int mn = 11, j = -1;
        for(int i = 0; i < N; ++i){
            if(is[i]){ continue;}
            int cnt = 0;
            for(auto w : v[i]){
                cnt += (w == '0');
            }
            if(cnt < mn){
                mn = cnt;
                j = i;
            }
        }
```

```
        is[j] = 1;
        rv[id] = v[j];
    }

    vector <int> a(N);
    for(int j = 0; j < N; ++j){
        int cnt = 0;
        for(int i = 0; i < N; ++i){
            cnt += (rv[i][j] == '0');
        }
        a[j] = cnt;
    }

    sort(all(a));

//    int ans = 0;
//    for(auto w : a){
//        ans += w;
//    }
    for(int i = 0; i < (int)al.size(); ++i){
        if(a == al[i]){
            return i;
        }
    }

//    return ans;
}

signed main(){
    ibase;
    gg();
    int t;
    cin >> t;
    string s;
    cin >> s;
    if(s == "transmit"){
        while(t --> 0){
            int n;
            cin >> n;
            auto v = get(n);
            for(auto w : v){
                for(auto c : w){
                    cout << c;
                }
                cout << endl;
            }
            cout << endl;
        }
    }
    else{
        while(t --> 0){
            vector < vector <char> > v(N, vector <char> (N));
            for(int i = 0; i < N; ++i){
                for(int j = 0; j < N; ++j){
                    cin >> v[i][j];
                }
            }
            cout << deget(v) << endl;
        }
    }
}
/**
8 8
2 3
2 3

5 3
2 3
1 1
**/
```

Task F ()

```
a = int(input())
b = int(input())
print(a + b)
```