

# Олимпиада СПбГУ по информатике 2022/23 учебного года

A	B	C	D	E	F	Sum
100	100	80	30	28	65	403

## Task A ()

```
#include <iostream>
#include <vector>
#include <string>

using namespace std;

int main()
{
    string s;
    int x;
    cin >> x;
    s = "0";
    for(int i = 0; i < 5; ++i){
        cin >> x;
        s = s.substr(0, x-1) + (char)(i+1 + '0') + s.substr(x-1, s.size()-x+1);
        //cout << s << " ";
    }
    vector<int> ans(6, 0);
    for(int i = 0; i < 6; ++i){
        for(int j = 0; j < 6; ++j){
            if(s[i] == (char)('0' + j)){
                ans[j] = i;
            }
        }
    }
    for(int i = 0; i < 6; ++i){
        cout << ans[i]+1 << " ";
    }
    return 0;
}
```

## Task B ()

```
#include <iostream>
#include <vector>
#include <string>

using namespace std;

int main()
{
    string op;
    cin >> op;
    int n;
    cin >> n;
    vector<int> a(n);
    for(int i = 0; i < n; ++i){
        cin >> a[i];
    }
    if(op == "first"){
        long long int sum = 0;
        for(int i = 0; i < n; ++i){
            sum += a[i];
        }
        cout << 1000*sum;
        return 0;
    }else{
        long long int sum = 0;
        for(int i = 0; i < n; ++i){
            sum += a[i]%1000;
        }
        cout << sum + a[0]/1000;
        return 0;
    }
    return 0;
}
```

## Task C ()

```
#include <iostream>
#include <vector>
#include <string>
#include <set>

using namespace std;

int main()
{
    vector<pair<int, int>> a(3);
    for(int i = 0; i < 3; ++i){
        cin >> a[i].first >> a[i].second;
    }
    vector<vector<int>> cur;
    cur = {{0, 1, 2}, {0, 2, 1}, {1, 0, 2}, {1, 2, 0}, {2, 1, 0}, {2, 0, 1}};
    set<pair<int, int>> ans;
    for(int i = 0; i < 3; ++i){
        ans.insert({min(a[i].first, a[i].second), max(a[i].first, a[i].second)});
    }
    for(int i = 0; i < 6; ++i){
        vector<int> oreder = cur[i];
        for(int j = 0; j < 8; ++j){
            int a1, a2, b1, b2, c1, c2;
            if(j < 4){
                a1 = a[oreder[0]].first;
                a2 = a[oreder[0]].second;
            } else {
                a2 = a[oreder[0]].first;
                a1 = a[oreder[0]].second;
            }
            if(j < 2 || j == 4 || j == 5){
                b1 = a[oreder[1]].first;
                b2 = a[oreder[1]].second;
            } else {
                b2 = a[oreder[1]].first;
                b1 = a[oreder[1]].second;
            }
            if(j%2 == 0){
                c1 = a[oreder[2]].first;
                c2 = a[oreder[2]].second;
            } else {
                c2 = a[oreder[2]].first;
                c1 = a[oreder[2]].second;
            }
            if(a1 >= b1 && a2 >= b2){
                if(a1 == b1){
                    if(a2 != b2){
                        ans.insert({min(a1, a2-b2), max(a1, a2-b2)});
                    }
                }
                if(a2 == b2){
                    if(a1 != b1){
                        ans.insert({min(a2, a1-b1), max(a2, a1-b1)});
                    }
                }
            }
            if(c1 <= a1 && c1 > a1 - b1 && a2 - b2 >= c2){
                if(c1 == a1){
                    ans.insert({min(a1, a2-b2-c2), max(a1, a2-b2-c2)});
                }
                if(a2 - b2 == c2){
                    ans.insert({min(a1-c1, a2-b2), max(a1-c1, a2-b2)});
                }
                ans.insert({min(a1-b1, b2), max(a1-b1, b2)});
            }
            if(c1 == b1 && c2 == a2-b2){
                ans.insert({min(a1-b1, a2), max(a1-b1, a2)});
            }
            if(c1 == a1-b1 && c2 == b2){
                ans.insert({min(a2-b2, a1), max(a2-b2, a1)});
            }
        }
    }
}
```

```

        if(c1 == a1-b1 && c2 >= a2-b2 && c2 <= a2){
            ans.insert({min(b1, a2-b2), max(b1, a2-b2)});
            ans.insert({min(a2-c2, a1-b1), max(a2-c2, a1-b1)});
        }
    }
}
set<pair<int,int> >::iterator it = ans.begin();
while(it != ans.end()){
    if((*it).first != 0){
        cout << (*it).first << "⌋" << (*it).second << "\n";
    }
    it ++;
}
return 0;
}

```

## Task D ()

```
#include <iostream>
#include <vector>
#include <string>
#include <set>

using namespace std;

int main()
{
    int n;
    cin >> n;
    if(n == 1){
        int x;
        cin >> x;
        cout << 1 << "┘" << x << endl;
        int a, b;
        cin >> a >> b;
        cout << 1 << "┘" << x << endl;
    }
    if(n == 2){
        int x, y;
        cin >> x >> y;
        cout << -1 << "┘" << -1 << endl;
    }
    return 0;
}
```

## Task E ()

```
#include <iostream>
#include <vector>
#include <string>
#include <set>
#include <algorithm>

using namespace std;

int main()
{
    int t;
    cin >> t;
    string op;
    cin >> op;
    vector<vector<long long int>> cnk(20, vector<long long int>(20, 1));
    for(int i = 1; i < 20; ++i){
        for(int j = 1; j < 20; ++j){
            cnk[i][j] = cnk[i-1][j] + cnk[i][j-1];
            //cout << cnk[i][j] << " ";
        }
        //cout << "\n";
    }
    for(int w = 0; w < t; ++w){
        if(op[0] == 't'){
            vector<vector<int>> ans(10, vector<int>(10, 0));
            vector<int> ot;
            int n;
            cin >> n;
            int a = 10;
            int b = 10;
            while(b != 0){
                while(cnk[a][b-1] < n){
                    n -= cnk[a][b-1];
                    a--;
                }
                ot.push_back(10 - a);
                b--;
            }
            for(int i = 0; i < 10; ++i){
                for(int j = 0; j < ot[i]; ++j){
                    ans[i][j] = 1;
                }
            }
            for(int i = 0; i < 10; ++i){
                for(int j = 0; j < 10; ++j){
                    cout << ans[i][j];
                }
                cout << "\n";
            }
        }
        else{
            vector<int> ot;
            for(int i = 0; i < 10; ++i){
                int c = 0;
                for(int j = 0; j < 10; ++j){
                    char x;
                    cin >> x;
                    if(x == '1'){
                        c++;
                    }
                }
                ot.push_back(c);
            }
            //cout << "&&&&";
            sort(ot.begin(), ot.end());
            long long int ans = 0;
            long long int p = 10;
            long long int b = 10;
            while(b != 0){
                long long int a = 10-ot[10-b];
                while(a != p){
                    ans += cnk[p][b-1];
                    p--;
                }
            }
        }
    }
}
```

```
        }  
        b --;  
    }  
    cout << ans+1 << "\n";  
}  
}  
return 0;  
}
```

## Task F ()