Задача «ИССЛЕДОВАНИЕ»
(Компилятор GNU c++ 11 4.9)

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

using namespace std;

#define vec vector
#define ALL(x) (x).begin(), (x).end()

typedef pair< int, int > pii;
typedef long long ll;

int const inf = 1000 * 1000 * 1000;
ll const inf64 = 1ll * inf * inf;

int n;
int a[5];

ll ff(int x) { // f'(x + 0.5)
    ll res = 0;
    for(int i = n;i >= 1;i--) {
        res = res * (2 * x + 1) + a[i];
    }
    return res;
}

void print_value(int x) {
    if(x > 0) cout << "+";
    cout << x;
}

void print_result(int l, int r, int t) {
    print_value(l);
    cout << " ";
    print_value(r);
    cout << " ";
    if(t == 1) {
        cout << "up\n";
    }else {
        cout << "down\n";
    }
}

int get_type(int x) {
    int t = ff(x);
    if(t > 0) t = 1;
    else if(t < 0) t = -1;
    return t;
}

int main() {

//    freopen("input.txt", "r", stdin);

    cin >> n;
    for(int i = n;i >= 0;i--) cin >> a[i];
```

```
    a[1] *= 8;
    a[2] *= 8;
    a[3] *= 6;
    a[4] *= 4;

    for(int i = -100;i < 100;i++) {
        int j = i;
        int t = get_type(i);
        while(j < 100 && get_type(j) == t) j++;
        print_result(i, j, t);
        i = j - 1;
    }

    return 0;
}
```

Задача «БИРЖА-2017»
(Компилятор GNU c++ 11 4.9)

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

using namespace std;

typedef long long ll;

inline int get_cnt(int x) {
    int res = 0;
    while(x) res++, x /= 10;
    return res;
}

int main() {

    int pw[7];

    pw[0] = 1;
    for(int i = 1;i < 7;i++) {
        pw[i] = pw[i - 1] * 10;
    }

    int mx1 = -1;
    int mx2 = -1;
    int n, x;

    scanf("%d", &n);

      assert(n >= 2 && n <= 500);

    for(int i = 0;i < n;i++) {
        scanf("%d", &x);
      assert(x >= 1 && x <= 100000);
        if(x > mx1) {
            mx2 = mx1;
            mx1 = x;
        }else if(x > mx2) {
            mx2 = x;
        }
    }

    assert(mx1 >= 1 && mx1 <= 100000);
    assert(mx2 >= 1 && mx2 <= 100000);

    assert(get_cnt(mx1) >= 0 && get_cnt(mx1) < 7);
    assert(get_cnt(mx2) >= 0 && get_cnt(mx2) < 7);

    ll u = mx1 + 1ll * pw[get_cnt(mx1)] * mx2;
    ll v = mx2 + 1ll * pw[get_cnt(mx2)] * mx1;

    printf("%lld\n", max(u, v));

    return 0;
}
```

Задача «КВАТЕР-БОЛ»
(Компилятор GNU c++ 11 4.9)

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

using namespace std;

typedef pair< int, int > pii;

int const inf = 1000 * 1000 * 1000;

struct pt {
    int x, y;
    pt() : x(0), y(0) { }
    pt(int x, int y) : x(x), y(y) { }
};

pt a[4];

bool check(pt p) {
    int type = 0;
    for(int cur, j, i = 0;i < 4;i++) {
        j = (i + 1) % 4;
        cur = (a[j].x - a[i].x) * (p.y - a[i].y) -
              (p.x - a[i].x) * (a[j].y - a[i].y);
        if(cur > 0) cur = 1;
        else if(cur < 0) cur = -1;
        if(cur == 0) return 0;
        if(i == 0) type = cur;
        else if(type != cur) return 0;
    }
    return 1;
}

pii get() {
    for(int i = 0;i < 4;i++) {
        cin >> a[i].x >> a[i].y;
    }
    int area = 0;
    for(int j, i = 0;i < 4;i++) {
        j = (i + 1) % 4;
        area += a[i].x * a[j].y - a[j].x * a[i].y;
    }
    if(area < 0) area = -area;
    int cnt = 0;
    for(int i = 0;i < 4;i++) {
        pt p;
        cin >> p.x >> p.y;
        cnt += check(p);
    }
    return make_pair(-cnt, area);
}

int main() {

//    freopen("input.txt", "r", stdin);
```

```
    pair< pii, int > res = make_pair( make_pair(inf, inf), inf
);

    for(int i = 0;i < 5;i++) {
        res = min(res, make_pair( get(), i + 1 ) );
    }

    cout << res.second << "\n";

    return 0;
}
```

Задача «ПОХОД»
(Компилятор GNU c++ 11 4.9)

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

using namespace std;

#define vec vector
#define ALL(x) (x).begin(), (x).end()

typedef pair< int, int > pii;
typedef long long ll;

int const inf = 1000 * 1000 * 1000;
ll const inf64 = 1ll * inf * inf;

int const N = 35;

int n;
int h[N][N];
bool used[N][N];
bool used2[N][N];

bool check(int i, int j) {
    for(int di = -1;di <= 1;di++) {
        for(int dj = -1;dj <= 1;dj++) {
            if(h[i + di][j + dj] > h[i][j]) {
                return 0;
            }
        }
    }
    return 1;
}

int ok;
int cnt;
int l, r, u, d;

void dfs(int i, int j) {
    if(used[i][j]) return;
    used[i][j] = 1;
    ok &= check(i, j);
    cnt++;
    l = min(l, j);
    r = max(r, j);
    u = min(u, i);
    d = max(d, i);
    for(int di = -1;di <= 1;di++) {
        for(int dj = -1;dj <= 1;dj++) {
            if(abs(di) == abs(dj)) continue;
            int toi = i + di;
            int toj = j + dj;
            if(toi < 1 || toi > n || toj < 1 || toj > n)
continue;
            if(h[toi][toj] != h[i][j]) continue;
            dfs(toi, toj);
        }
```

```
    }
}

void dfs2(int i, int j) {
    if(used2[i][j]) return;
    used2[i][j] = 1;
    ok &= check(i, j);
    cnt++;
    for(int di = -1;di <= 1;di++) {
        for(int dj = -1;dj <= 1;dj++) {
            if(di == 0 || dj == 0) continue;
            int toi = i + di;
            int toj = j + dj;
            if(toi < 1 || toi > n || toj < 1 || toj > n)
continue;
            if(h[toi][toj] != h[i][j]) continue;
            dfs2(toi, toj);
        }
    }
}

int main() {

//    freopen("input.txt", "r", stdin);

    for(int i = 0;i < N;i++) {
        for(int j = 0;j < N;j++) {
            h[i][j] = inf;
        }
    }

    scanf("%d", &n);

    for(int i = 1;i <= n;i++) {
        for(int j = 1;j <= n;j++) {
            scanf("%d", &h[i][j]);
        }
    }

    int res;
    int res1, res2, res3;

    res = 0;
    for(int i = 1;i <= n;i++) {
        for(int j = 1;j <= n;j++) {
            if(used[i][j]) continue;
            ok = 1;
            cnt = 0;
            l = +inf;
            r = -inf;
            d = -inf;
            u = +inf;
            dfs(i, j);
            if(ok && cnt >= 3 && (l != r && u != d) ) {
                res++;
            }
        }
```

```
}

res1 = res;

res = 0;
for(int i = 2;i < n;i++) {
    for(int j = 2;j < n;j++) {
        ok = 1;
        for(int di = -1;di <= 1;di++) {
            for(int dj = -1;dj <= 1;dj++) {
                if(di == 0 && dj == 0) continue;
                int toi = i + di;
                int toj = j + dj;
                if(h[toi][toj] >= h[i][j]) {
                    ok = 0;
                }
            }
        }
        res += ok;
    }
}

res2 = res;

for(int i = 0;i < N;i++) {
    for(int j = 0;j < N;j++) {
        used[i][j] = 0;
    }
}

res = 0;
for(int i = 2;i < n;i++) {
    for(int j = 2;j < n;j++) {
        if(used[i][j] || used2[i][j]) continue;
        ok = 1;
        cnt = 0;
        l = +inf;
        r = -inf;
        u = +inf;
        d = -inf;
        dfs(i, j);
        if(cnt >= 2 && (l == r || u == d) && ok) {
            res++;
            continue;
        }
        if(cnt >= 2) continue;
        ok = 1;
        cnt = 0;
        dfs2(i, j);
        if(cnt >= 2 && ok) {
            res++;
        }
    }
}

res3 = res;
```

```
    printf("%d %d %d\n", res1, res2, res3);

    return 0;
}
```