

MCD interactivo

Autor del problema: Félix Moreno Peñarubia (Miembro del comité científico)

Resumen del enunciado

Debes adivinar dos enteros positivos x , y haciendo preguntas donde se te da:
 $mcd(|x - a|, |y - b|)$

Explicación de la solución

- Consideramos la representación binaria de x , y . Podemos extraer el último bit preguntando (0, 0), (1, 0), (0, 1), (1, 1).
- Podemos ir extrayendo los bits siguientes con la misma estrategia en 4 o 3 preguntas por bit.
- Para hacerlo con 2 preguntas por bit, tenemos que tener en cuenta la potencia de 2 del mcd y tener en cuenta que el siguiente resultado no puede ser (0, 0).

Solución en C++

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

using namespace std;

typedef long long ll;
typedef pair<ll, ll> ii;
typedef vector<ii> vii;

ll ask(ll a, ll b) {
    cout << "? " << a << " " << b << endl;
    ll res;
    cin >> res;
    if(res == 0 || res == -1) exit(0);
    return res;
}

ll hp2(ll x) {
    ll r = 0;
    while(x % 2 == 0) {
        x /= 2;
        r++;
    }
    return r;
}

mt19937 gen(0);

ll random(ll x) {
    if(x < 1) return 0;
    return uniform_int_distribution<ll>(1, x)(gen);
}
```

```

}

int main() {
    ll i = hp2(ask(0, 0));
    ll x = 0;
    ll y = 0;
    int q = 0;
    int forbidden = 0;
    while(q < 121) {
        ll p2 = (1LL << i);
        vii v;
        ll rmax = ll(5e17)/p2;
        if(forbidden != 0) v.emplace_back(2*p2*random(rmax), 2*p2*random(rmax));
        if(forbidden != 1) v.emplace_back(p2 + 2*p2*random(rmax), 2*p2*random(rmax));
        if(forbidden != 2) v.emplace_back(2*p2*random(rmax), p2+2*p2*random(rmax));
        if(forbidden != 3) v.emplace_back(p2 + 2*p2*random(rmax), p2 + 2*p2*random(rmax));

        shuffle(v.begin(), v.end(), gen);

        for(ii p : v) {
            ll e = hp2(ask(x+p.first, y+p.second));
            q++;
            if(e > i) {
                for(ll j=i; j < e; ++j) {
                    x += p.first&(1LL << j);
                    y += p.second&(1LL << j);
                }
                i = e;
                forbidden = 0;
                if(p.first&(1LL << e)) forbidden += 1;
                if(p.second&(1LL << e)) forbidden += 2;
                break;
            }
        }

        ask(x, y);
    }
}

```

Autor de la solución: Félix Moreno Peñarrubia (Miembro del comité científico)