

**BỘ GIÁO DỤC VÀ ĐÀO TẠO
TRƯỜNG: ĐH GIAO THÔNG VẬN TẢI
KHOA: CNTT**



BÀI TẬP LỚN

Môn Học : Thuật toán và ứng dụng

Đề tài: Ngắm sao

Giáo viên hướng dẫn: Phạm Xuân Tích

Sinh viên thực hiện: Nguyễn Thị Ngũ _ 191200758

Lớp: Công nghệ thông tin 4

Khóa: 60

Hà Nội, năm 2021

Mục lục

1.	Nêu bài toán.....	3
2.	Nêu ý tưởng	3
3.	Mô phỏng tính toán bằng tay	4
4.	Code C/C++ hoặc python	5
5.	Đánh giá độ phức tạp.....	6

1. Nêu bài toán

MAH có sở thích ngắm sao. Hôm nay anh ấy thấy n ngôi sao trên bầu trời, ngôi sao thứ i có tọa độ (x_i, y_i) và độ sáng tối đa là c , và độ sáng ban đầu của các ngôi sao là $s_i (0 \leq s_i \leq c)$.

Ở thời điểm ban đầu ($t = 0$), ngôi sao thứ i có độ sáng s_i . Nếu ở thời điểm t một ngôi sao có độ sáng là x thì thời điểm $(t + 1)$ ngôi sao này có độ sáng là $x + 1$ khi $x + 1 \leq c$ và có độ sáng 0 trong trường hợp còn lại.

MAH muốn nhìn bầu trời q lần. Trong lần thứ i anh ấy sẽ nhìn vào thời điểm t_i và anh ấy sẽ nhìn một hình chữ nhật có các cạnh song song với trục tọa độ, góc trên bên trái là (x_{1i}, y_{1i}) và góc dưới bên phải là (x_{2i}, y_{2i}) . Mỗi lần nhìn, anh ấy muốn biết tổng độ sáng của tất cả các ngôi sao nằm trong hình chữ nhật (có thể nằm trên đường viền).

Input:

Dòng đầu gồm 3 số nguyên n, q, c ($1 \leq n, q \leq 10^5, 1 \leq c \leq 10$) biểu diễn số ngôi sao, số lần nhìn, và độ sáng tối đa của các ngôi sao.

N dòng tiếp theo biểu diễn các ngôi sao. Dòng thứ i chứa 3 số nguyên x_i, y_i, s_i ($1 \leq x_i, y_i \leq 100, 0 \leq s_i \leq c \leq 10$) biểu diễn tọa độ của ngôi sao thứ i và độ sáng ban đầu.

Q dòng tiếp theo mỗi dòng chứa 5 số nguyên $t_i, x_{1i}, y_{1i}, x_{2i}, y_{2i}$ ($0 \leq t_i \leq 109, 1 \leq x_{1i} < x_{2i} \leq 100, 1 \leq y_{1i} < y_{2i} \leq 100$) biểu diễn thời điểm nhìn và tọa độ của hình chữ nhật.

Output:

Mỗi dòng in tổng độ sáng trong mỗi lần nhìn.

2. Nêu ý tưởng

- Trường hợp test yếu: tìm các ngôi sao nằm trong hình chữ nhật. Áp dụng công thức tính độ sáng của ngôi sao có tọa độ x_i, y_i .

=> Do dùng 2 vòng lặp nên thời gian chạy sẽ lâu nên ta dùng quy hoạch động cho bài này.

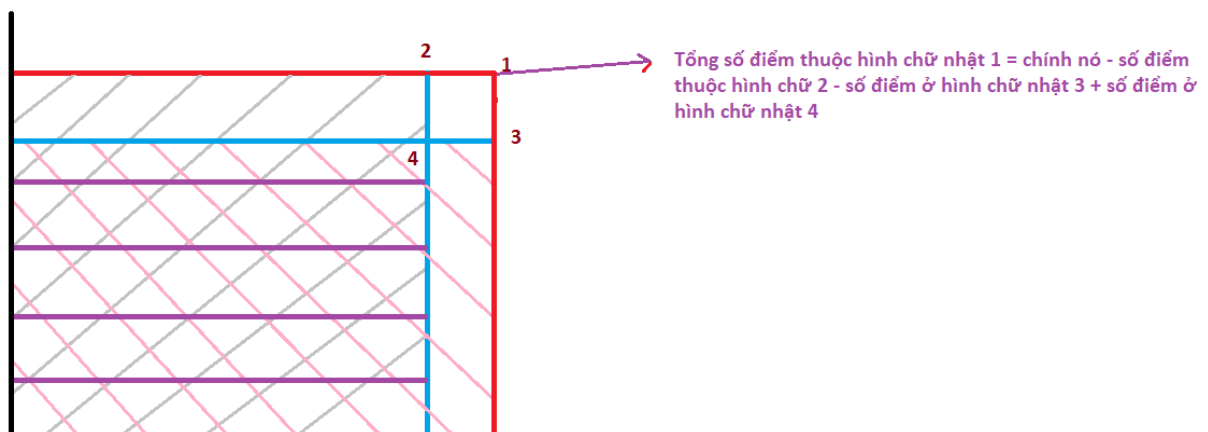
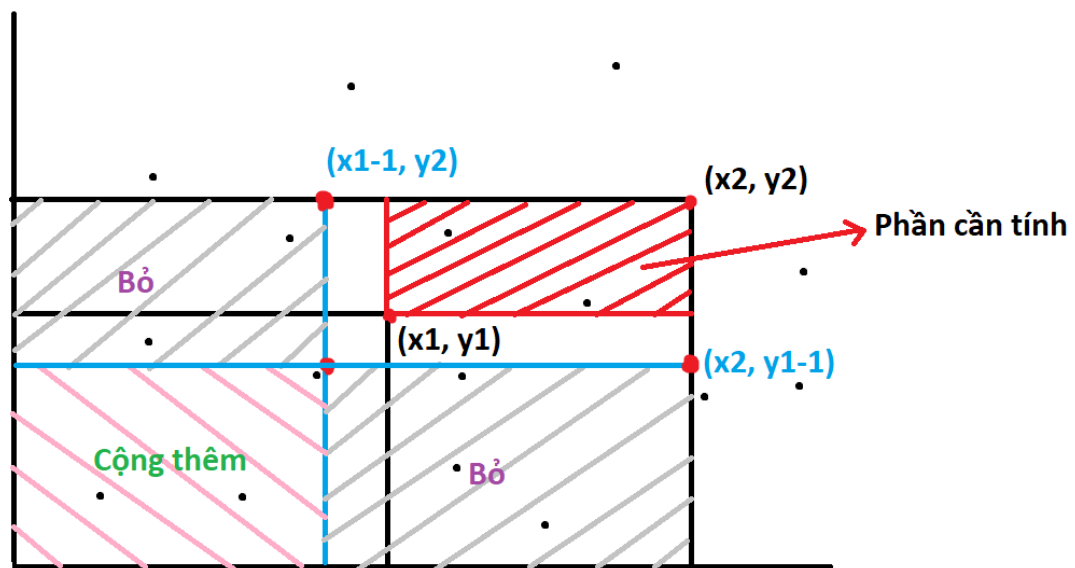
Ý tưởng:

- Nhập tọa độ và độ sáng ban đầu của các ngôi sao. Đếm số lượng ngôi sao ở tọa độ x_i, y_i và có độ sáng s_i .

- Gọi $P[t_i][x_i][y_i]$ là số ngôi sao có độ sáng ban đầu là t thuộc hình chữ nhật có 2 đỉnh $(0,0)$ và (x_i, y_i) . Ta tìm tất cả $P[t_i][x_i][y_i]$ với $1 \leq t_i \leq c, 1 \leq x_i < x_i \leq 100$

- Chạy bộ test: Với mỗi hình chữ nhật, số ngôi sao ở hình chữ nhật có 2 đỉnh (x_{1i}, y_{1i}) và $(x_{2i}, y_{2i}) =$ tổng của $(P[t_i][x_{2i}][y_{2i}] - P[t_i][x_i - 1][y_i] - P[t_i][x_i][y_i - 1] + P[t_i][x_i - 1][y_i - 1])$ nhân với độ sáng tại thời điểm test.

3. Mô phỏng tính toán bằng tay



VD:

```
2 1 5
40 6 4
60 48 4
17 4 1 47 38
```

- $P[4][40][6] = 1$; // thời điểm ban đầu $t = 4$ tọa độ $(40, 6)$ có 1 điểm

- $P[4][60][48] = 1$; // thời điểm ban đầu $t = 4$ tọa độ $(60, 48)$ có 1 điểm

\Rightarrow Quy hoạch động giá trị của $P[1][1][1]$ đến $P[5][100][100]$:

- $t = 1$:

+ $P[1][1][1] = P[1][1][1] - P[1][0][1] - P[1][1][0] + P[1][0][0]$; // tọa độ $(1, 1)$

+ $P[1][1][2] = P[1][1][2] - P[1][0][2] - P[1][1][1] + P[1][0][1]$; // tọa độ $(1, 2)$

..... tương tự

- $t = 4$:

$+P[4][1][1] = P[4][1][1] - P[4][0][1] - P[4][1][0] + P[4][0][0]; \quad // \text{ tọa độ } (1, 1)$

 $+P[4][40][6] = P[4][40][6] - P[4][39][6] - P[4][40][5] + P[4][39][5]; \quad // \text{ tọa độ } (1, 2)$
 $= 1 - 0 - 0 + 0 = 1$
 tương tự
 - t = 5:
 $+P[5][1][1] = P[5][1][1] - P[5][0][1] - P[5][1][0] + P[5][0][0]; \quad // \text{ tọa độ } (1, 1)$
 $+P[5][1][2] = P[5][1][2] - P[5][0][2] - P[5][1][1] + P[5][0][1]; \quad // \text{ tọa độ } (1, 2)$

 * Các test: hình chữ nhật (4,1), (47, 38) => Chỉ có điểm (6,4) nằm trong hình chữ nhật
 - t = 1: $P[1][][] = 0$
 - t = 2: $P[2][][] = 0$
 - t = 3: $P[3][][] = 0$
 - t = 4: $P[4][][] = 1$
 $\Rightarrow KQ = 0*(1+4\%(5+1))\%(4+1) + 0*(2+4\%(5+1))\%(4+1) + 0*(3+4\%(5+1))\%(4+1) + 1*(4+17\%(5+1))\%(4+1) = 0 + 0 + 0 + 3 = 3$

4. Code C/C++ hoặc python

```

#include<bits/stdc++.h>

#define ll long long

#define x first
#define y second

using namespace std;

int main()
{
    ios_base::sync_with_stdio(false);

    cin.tie(0); cout.tie(0);

    int n, q, c, t, x1, y1, x2, y2, d;

    cin>>n>>q>>c;

    int p[15][102][102];

    int xi, yi, si;

    for(int i=0;i<n;i++){

```

```

        cin>>xi>>yi>>si;

        p[si][xi][yi]++;

    }

    for(int t=1; t<=c; t++) for(int i=1; i<=100; i++) for(int j=1; j<=100; j++){ //dem so
luong diem co do sang ban dau la t tai hcn co dinh (0,0) va (i,j)

        p[t][i][j] += p[t][i-1][j] + p[t][i][j-1] - p[t][i-1][j-1];

    }

    while(q--){

        d=0;

        cin>>t>>x1>>y1>>x2>>y2;

        for(int i=1; i<=c; i++){

            d+= (p[i][x2][y2] - p[i][x1-1][y2] - p[i][x2][y1-1] + p[i][x1-1][y1-1] ) *
((i+t%(c+1))%(c+1));

        }

        cout<<d<<"\n";

    }

}

```

5. Đánh giá độ phức tạp

+ Vòng lặp đầu tiên: $O(n)$

+ Vòng lặp thứ 2: $O(10^4 * c)$

+ Vòng lặp thứ 3: $O(q * c)$

=> Độ phức tạp của thuật toán là: $O(n)$