

**Câu 1. (4,0 điểm) Cách xa nguyên tố****(XNT.CPP)**

Cho số nguyên dương  $A$  và số nguyên tố  $B$ , khoảng cách của số  $A$  với số nguyên tố  $B$  là  $|A - B|$ . Một số nguyên dương  $X$  được gọi là **cách xa nguyên tố** nếu khoảng cách từ  $X$  tới số nguyên tố gần nó nhất lớn hơn 10.

Bờm muốn biết trong các số nguyên từ 1 đến  $N$  có bao nhiêu số cách xa nguyên tố.

**Yêu cầu:** Hãy giúp Bờm tìm số lượng số cách xa nguyên tố từ 1 đến  $N$ .

**Dữ liệu** vào gồm một số nguyên dương  $N$  ( $10 \leq N \leq 10^6$ ).

**Kết quả** đưa ra số lượng số cách xa nguyên tố tìm được.

**Ví dụ:**

Input	Output
20	0
1150	1

**Giải thích:** số cách xa nguyên tố nhỏ nhất là 1140 với 2 số nguyên tố gần nhất là 1129 và 1151.

**Ràng buộc:**

- 80% số test tương ứng 50% số điểm có  $n \leq 10^5$
- 20% số test còn lại tương ứng 20% số điểm có  $10^5 < n \leq 10^6$

**Câu 2. Đoạn tăng (INCSEG.CPP)**

Cho dãy số nguyên  $A = (a_0, a_1, \dots, a_{n-1})$ . Hãy tìm một đoạn gồm các phần tử **liên tiếp** trong dãy  $A : a_L, a_{L+1}, \dots, a_{H-1}, a_H$  thỏa mãn hai điều kiện:

- Các phần tử trong đoạn có thứ tự không giảm:  $a_L \leq a_{L+1} \leq \dots \leq a_H$
- Số phần tử trong đoạn là nhiều nhất có thể

Quy ước: Đoạn chỉ gồm đúng 1 phần tử trong dãy  $A$  cũng được coi là có thứ tự không giảm

**Dữ liệu:** Vào từ thiết bị nhập chuẩn

- Dòng 1 chứa số nguyên dương  $n \leq 10^5$
- Dòng 2 chứa  $n$  số nguyên  $a_0, a_1, \dots, a_{n-1}$  cách nhau bởi dấu cách ( $\forall i: |a_i| \leq 10^9$ )

**Kết quả:** Ghi ra thiết bị xuất chuẩn một số nguyên duy nhất là số phần tử của đoạn không giảm dài nhất tìm được

**Ví dụ:**

Input	Output
9 1 3 2 2 3 4 7 1 2	5
5 2 3 1 2 1	2

**Ràng buộc:**

- 75% số test tương ứng 75% số điểm có  $n \leq 10^4$
- 25% số test còn lại tương ứng 25% số điểm có  $10^4 < n \leq 10^5$

### **Câu 3. (4,0 điểm) HÌNH CHỮ NHẬT**

**(HCN.CPP)**

Cho một hình chữ nhật gồm  $m$  dòng và  $n$  cột. Các dòng của bảng được đánh số từ  $1$  đến  $m$ , từ trên xuống dưới, các cột được đánh số từ  $1$  đến  $n$ , từ trái sang phải. Ô nằm trên giao của dòng  $i$  và cột  $j$  là ô  $(i, j)$ , chứa một số nguyên.

**Yêu cầu:** Hãy viết chương trình đếm số lượng hình chữ nhật con trong hình chữ nhật trên sao cho 4 ô nằm bốn góc hình chữ nhật con này có giá trị bằng nhau?

**Dữ liệu:**

- Dòng đầu tiên chứa hai số nguyên  $m, n$  ( $1 < m, n \leq 200$ ).
- $m$  dòng tiếp theo, dòng thứ  $i$  chứa  $n$  số nguyên, mỗi số có giá trị tuyệt đối không vượt quá  $10^5$ . Các số cách nhau ít nhất một dấu cách.

**Kết quả:** gồm một số duy nhất là số hình chữ nhật con tìm được.

**Ví dụ:**

Input	Output
3 4 0 0 1 0 0 1 1 0 0 1 1 0	4

**Ràng buộc:**

- 80% số test tương ứng 80% số điểm có  $m, n \leq 50$
- 20% số test còn lại tương ứng 20% số điểm có  $50 < m, n \leq 200$

### **Câu 4: Đường đi ngắn nhất (ROBOT.CPP)**

Cho một bảng vuông kích thước  $N \times N$  (với  $2 < N < 100$ ). Mỗi ô trong bảng ghi một số nguyên  $a$  ( $a < |32000|$ ). Một robot di chuyển từ ô  $[1,1]$  xuống ô  $[n,n]$ , mỗi bước robot chỉ có thể di chuyển sang ô cạnh bên phải hoặc bên dưới so với ô nó đang đứng.

**Yêu cầu:** Tìm đường đi của robot sao cho tổng đường đi là ngắn nhất.

**Dữ liệu vào:**

- Dòng đầu ghi giá trị số  $n$ .
- Dòng thứ  $i$  trong  $n$  dòng tiếp theo ghi  $n$  số trên dòng  $i$  của bảng theo thứ tự từ trái qua phải.

**Dữ liệu ra:** một số nguyên là tổng đường đi ngắn nhất tìm được.

**Ví dụ:**

Sample Input	Sample Output	(Giải thích: đường đi có tổng bé nhất:
3 1 8 5 5 4 2 1 25 2	14	$(1,1) \Rightarrow (2,1) \Rightarrow (2,2) \Rightarrow (2,3) \Rightarrow (3,3)$ có tổng: $1 + 5 + 4 + 2 + 2 = 14$ )

### **Câu 5. ÉCH KIỂM MÔI**

**(FROG.CPP)**

Trong một cái ao có  $n$  cây nổi trên mặt nước. Các cây được đánh số từ 1 đến  $n$ . Khi nhìn từ trên, vị trí mỗi cây được cho bởi cặp tọa độ  $(x, y)$ . Có một con ếch sống trong ao, hàng ngày nó phải đi kiếm thức ăn bằng cách nhảy qua các cây. Từ cây thứ  $i$  có tọa độ  $(x_i, y_i)$  con ếch có thể nhảy đến cây thứ  $j$  có tọa độ  $(x_j, y_j)$  nếu thỏa mãn một trong hai điều kiện sau:

- $x_j > x_i$  và  $y_j = y_i$
- $y_j > y_i$  và  $x_j = x_i$

Ở mỗi cây, chúng ta biết lượng thức ăn mà con ếch có thể kiếm được và nó sẽ ăn hết lượng thức ăn khi ở trên cây đó.

Con ếch muốn đi từ cây 1 đến cây  $n$  và ăn được lượng thức ăn lớn nhất. Ban đầu, con ếch đứng ở vị trí cây 1 và ăn hết lượng thức ăn ở đó.

**Yêu cầu:** Tìm cách đi qua các cây để con ếch kiếm được lượng thức ăn nhiều nhất.

**Dữ liệu**

- Dòng thứ nhất chứa số nguyên  $n$  ( $2 \leq n \leq 10^5$ ).
- Dòng thứ  $i$  trong  $n$  dòng sau, mỗi dòng chứa 3 số nguyên  $x_i, y_i$  và  $v_i$  ( $0 \leq x_i, y_i \leq 10^5$ ,  $0 \leq v_i \leq 10^3$ ), trong đó  $(x_i, y_i)$  là tọa độ cây thứ  $i$  và  $v_i$  là lượng thức ăn ở cây  $i$ . Các số cách nhau bởi dấu cách và không có hai cây nào cùng một vị trí.

Chú ý: Dữ liệu vào đảm bảo rằng luôn tồn tại một dãy các bước nhảy mà con ếch đi từ cây 1 đến cây  $n$ .

**Kết quả** đưa ra một số duy nhất là tổng lượng thức ăn lớn nhất con ếch kiếm được.

**Ví dụ:**

Input	Output
6 1 1 5 2 1 5 1 2 4 2 3 5 3 2 30 3 3 5	44

