**ĐỀ SỐ 26 - Ngày 12 – 10 -2022**

**Bài 1. Chênh lệch chiều cao**

Tuyến đường anh ta đang đi được mô tả như một chuỗi gồm các số N đại diện cho chiều cao của con đường tại các điểm cách đều nhau của tuyến đường, từ đầu đến cuối của nó. Tomislav quan tâm đến đoạn đường lớn nhất đi lên ngọn đồi mà anh phải đi, theo thông tin anh có. Hãy gọi cho một phân khúc như vậy là một ‘leo lên’. Bỏ qua các chi tiết, chỉ tính đến sự chênh lệch chiều cao của một lần leo chứ không phải chiều dài của nó.

Một cuộc leo núi được định nghĩa chặt chẽ hơn là một chuỗi tăng liên tiếp của ít nhất hai số mô tả con đường. Kích thước của sự leo lên là sự khác biệt giữa số cuối cùng và số đầu tiên trong phần tiếp theo.

Ví dụ: hãy để cân nhắc một tuyến đường được mô tả bởi chuỗi độ cao sau: 12 3\_5\_7\_10 6 1\_11. Số gạch chân đại diện cho hai leo núi khác nhau. Kích thước của lần leo đầu tiên là 7. Lần leo thứ hai lớn hơn, với kích thước 10. Điểm có chiều cao 12 và 6 không phải là một phần của bất kỳ lần leo nào. Giúp Tomislav và tính toán leo lên lớn nhất!

INPUT

Dòng đầu tiên chứa số nguyên dương N (1 ≤ N 1000), số lượng điểm đo trên tuyến.

Dòng đầu vào thứ hai chứa N số nguyên dương Pi (1 ≤ Pi ≤ 1000), độ cao của các điểm đo trên tuyến.

OUTPUT

Dòng đầu ra đầu tiên và duy nhất nên chứa kích thước của lần leo lớn nhất. Nếu tuyến đường trong đầu vào không chứa bất kỳ sự leo trèo nào, đầu ra 0.

Ví dụ:

|  |  |
| --- | --- |
| **Uspon.inp** | **Uspon.out** |
| 5  1 2 1 4 6 | 5 |
| 8  12 20 1 3 4 4 11 1 | 8 |
| 6  10 8 8 6 4 3 | 0 |

Ở ví dụ thứ 2: ‘leo lên’ là 12-20; 1-3-4; và 4-11. 1-3-4-4-11 không phải là 1 ‘leo lên’ vì số thứ tự phải được tăng lên.

**Bài 2. Nguyên tố tốt nhất**

Nói số ***a*** tốt hơn ***b*** nếu tổng bình phương các chữ số của ***a*** (trong hệ cơ số 10) lớn hơn tổng bình phương các chữ số của ***b*** hoặc các tổng này bằng nhau nhưng ***a*** < ***b***.

***Yêu cầu***: Cho hai số nguyên ***l*** và ***r*** (2 ≤ ***l*** ≤ ***r*** ≤ 50 000). Hãy tìm số nguyên tố tốt nhất trong khoảng [***l***, ***r***]. Nếu trong khoảng này không có số nguyên tố nào thì đưa ra số -1.

*Input:*

Một dòng chứa hai số l, r

*Output:*

Kết quả tìm được

Ví dụ:

|  |  |
| --- | --- |
| **Primemost.inp** | **Primemost.out** |
| 2 28 | 19 |

Bài 3. Chữ số

Xét dãy số tư nhiên {an} được xây dựng theo quy tắc sau:

* Cho trước số a0 là một số tự nhiên có tối đa 10 chữ số.
* Số ai (i>0) là một số tự nhiên nhận được từ số ai bằng cách viết thêm vào sau số của ai những chữ số của số ai-1 nhưng theo thứ tự ngược lại.

Ví dụ: Với a0 = 246 thì a1 = 246642, a2 = 246642246642, a3 = 246642246642246642246642

Với hai số N và M cho trước (1 < N < 25, 1 < M < 109), hãy tìm chữ số thứ M trong aN.

Dữ liệu vào: Cho file văn bản với tên là CHUSO.INP trons: đó dòng đầu chứa số a0, dòng thứ hai chứa hai số tự nhiên N và M.

Kết quả ghi ra file văn bản với tên là CHUSO.OUT. Trong trường hợp có lời giải, file này sẽ chứa số tìm được, ngược lại file này chứa số -1.

Ví dụ:

|  |  |  |  |
| --- | --- | --- | --- |
| **CHUSO.INP** | **CHUSO. OUT** | **CHUSO.INP** | **CHUSO. OUT** |
| 246  3 7 | 2 | 12  3 17 | -1 |

**Bài 4. Cặp điểm gần nhất**

Trên trục số, cho n điểm A1, A2,... An có tọa độ lần lượt: x1, x2,... xn. Các điểm không nhất thiết phân biệt.

***Yêu cầu:*** Hãy xác định các cặp điểm Ai, Aj (0 < i < j < n + 1) sao cho khoảng cách Ai, Aj ngắn nhất.

***Dữ liệu vào*:** File văn bản NEAREST.INP gồm 2 dòng:

* Dòng 1: Số nguyên n, 2 ≤ n ≤ 106
* Dòng 2: n số: x1, x2,... xn. Các tọa độ có giá trị tuyệt đối không quá 109.

***Dữ liệu ra*:** File văn bản NEAREST.OUT. Ghi ra hai giá trị trên một dòng: khoảng cách ngắn nhất tìm được và số lượng cặp điểm có khoảng cách bằng khoảng cách ngắn nhất.

***Ví dụ:***

|  |  |  |
| --- | --- | --- |
| **NEAREST.INP** | **NEAREST.OUT** | **Giải thích** |
| 4  5 2 1 3 | 1 2 | Khoảng cách: 1  2 cặp điểm có khoảng cách 1: (2,1) và (2,3) |

**Bài 5. Ước dãy**

Cho một mảng phần tử, ta định nghĩa vẻ đẹp của một mảng là ước chung lớn nhất của tất cả các phần tử của nó. Vì muốn vẻ đẹp càng lớn càng tốt, ta có thể giảm giá trị các phần tử đi không quá đơn vị.

***Yêu cầu*:** Cho mảng phần tử và .Hãy tính vẻ đẹp lớn nhất có thể của mảng đã cho.

***Dữ liệu*:** Vào từ file văn bản **SUBKGCD.INP** gồm

* Dòng đầu chứa 2 số ()
* Dòng thứ 2 chứa số nguyên dương

***Kết quả*:** Ghi ra file văn bản **SUBKGCD.OUT** một số duy nhất là vẻ đẹp lớn nhất có thể của mảng đã cho.

***Ví dụ*:**

|  |  |
| --- | --- |
| **SUBKGCD.INP** | **SUBKGCD.OUT** |
| 5 3  8 21 52 15 77 | 7 |