|  |  |
| --- | --- |
|  | **ĐỀ THI HỌC SINH GIỎI CẤP CƠ SỞ**  **NĂM HỌC 2024 -2025** |
|  | Môn: Tin học -THPT |
|  | *Thời gian:120 phút (Không kểt hời gian phát đề)* |

**TỔNG QUAN BÀI THI**

**Thí sinh lập trình trên máy tính**

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **Tên bài** | **Tên tệp**  **chương trình** | **Tên tệp**  **dữ liệu vào** | **Tên tệp**  **kết quả** | **Giới hạn**  **thời gian** | **Điểm** |
| **Câu 1: Cực trị** | EXT.\* | EXT.INP | EXT.OUT | 1 giây/ 1test | 8 |
| **Câu 2: Xóa phần tử** | CELE.\* | CELE.INP | CELE.OUT | 1 giây/ 1test | 8 |
| **Câu 3: Số may mắn** | LUCKYSUM.\* | LUCKYSUM.INP | LUCKYSUM.OUT | 1 giây/ 1test | 4 |

***Yêu cầu thí sinh đọc kỹ hướng dẫn dưới đây:***

* *Dấu \* được thay thế bởi PAS hoặc CPP của ngôn ngữ lập trình được sử dụng tương ứng là Pascal hoặc C++*
* *Chương trình chỉ in kết quả theo yêu cầu đề bài, không in bất kỳ thông tin nào khác*

**Viết chương trình giải các bài toán sau:**

**Câu1(8,0điểm): Cực trị**

Cho một dãy số *a*1, *a*2, …, *an*. Phần tử *ai* của dãy số gọi là *cực tiểu* nếu nó nhỏ hơn hai phần tử láng giềng của nó, tức là *ai* < *ai*-1 và *ai* < *ai*+1. Phần tử *ai* của dãy số gọi là *cực đại* nếu nó lớn hơn hai phần tử láng giềng của nó, tức là *ai* > *ai*-1 và *ai* > *ai*+1. Chú ý rằng *a*1 và *an* chỉ có một láng giềng, vì vậy nó không bao giờ là cực tiểu hoặc cực đại.

Một phần tử gọi là *cực trị* nếu nó là cực tiểu hoặc cực đại. Nhiệm vụ của bạn là đếm số phần tử cực trị của một dãy số cho trước.

**Dữ liệu:** Dòng đầu tiên chứa số nguyên *n* (1 ≤ *n* ≤ 105) là số phần tử của dãy số. Dòng thứ hai chứa *n* số nguyên *a*1, *a*2, …, *an* (-109 ≤ *ai* ≤ 109).

**Kết quả:** Ghi ra một số nguyên là số phần tử cực trị của dãy số.

**Ví dụ:**

|  |  |
| --- | --- |
| EXT.INP | EXT.OUT |
| 3  1 2 3 | 0 |
| 4  1 5 2 5 | 2 |

**Câu 2(8 điểm): Xóa phần tử**

Cho dãy số 𝐴 = 𝑎1, 𝑎2, … , 𝑎𝑛. Chi phí để xóa phần tử thứ 𝑖 của dãy 𝐴 bằng tổng của 𝑎𝑖 với các phần tử liền kề với nó. Ví dụ, với dãy 𝐴 = 1, 3, 2 thì:

- Chi phí để xóa phần tử thứ hai là 𝑎1 + 𝑎2 + 𝑎3 = 1 + 3 + 2 = 6;

- Chi phí để xóa phần tử thứ nhất là 𝑎1 + 𝑎2 = 1 + 3 = 4 do không có phần tử liền kề bên trái;

- Chi phí để xóa phần tử thứ ba là 𝑎2 + 𝑎3 = 3 + 2 = 5 do không có phần tử liền kề bên phải.

Nếu dãy chỉ còn một phần tử thì chi phí xóa phần tử duy nhất này bằng chính giá trị của nó vì không có phần tử nào liền kề với nó.

Bạn cần thực hiện lặp lại thao tác sau cho tới khi xóa hết toàn bộ các phần tử của dãy đã cho:

Chọn phần tử có giá trị lớn nhất trong dãy (nếu có nhiều hơn một phần tử cùng có giá trị lớn nhất thì chọn phần tử có chỉ số nhỏ nhất) rồi xóa nó khỏi dãy.

Hãy lập trình tính tổng chi phí để xóa hết toàn bộ các phần tử của dãy đã cho theo cách nói trên.

**Dữ liệu**

Dòng 1: chứa một số nguyên 𝑛 (1 ≤ 𝑛 ≤ 106

Dòng 2: chứa 𝑛 số nguyên 𝑎1, 𝑎2, … , 𝑎𝑛 (1 ≤ 𝑎𝑖 ≤ 109 , ∀𝑖 = 1 … 𝑛).

**Kết quả**

Ghi một số nguyên duy nhất là tổng chi phí tính được theo yêu cầu của đề bài.

**Ví dụ**

|  |  |  |
| --- | --- | --- |
| CELE.INP | CELE.OUT | **Giải thích** |
| 5  4 5 3 4 5 | 38 | Dãy hiện tại: (4, 5, 3, 4, 5) hai phần tử có giá trị lớn nhất trong dãy là 𝑎2 = 𝑎5 = 5. Theo yêu cầu, ta cần xóa phần tử có chỉ số nhỏ nhất, tức là xóa 𝑎2, chi phí xóa là 4+ 5 +3 = 12;  - Dãy hiện tại: (4, 3, 4, 5). Xóa phần tử 𝑎4 = 5 với chi phí là 4+ 5 = 9;  - Dãy hiện tại: (4, 3, 4). Xóa phần tử 𝑎1 = 4 với chi phí là 4 + 3 = 7;  - Dãy hiện tại: (3, 4). Xóa phần tử 𝑎2 = 4 với chi phí là 3+ 4 = 7;  - Dãy hiện tại: (3). Xóa phần tử 𝑎1 = 3 với chi phí là 3;  - Đã xóa hết dãy, tổng chi phí là 12 +9 + 7+ 7 + 3 = 38. |

**Ràng buộc**

Subtask 1: 29% số điểm có 𝑎1 ≥ 𝑎2 ≥ ⋯ ≥ 𝑎𝑛;

Subtask 2: 11% số điểm có 𝑛 ≤ 1000;

Subtask 3: 35% số điểm có 𝑎1 ≤ 𝑎2 ≤ ⋯ ≤ 𝑎𝑛;

Subtask 4: 25% số điểm còn lại không có thêm ràng buộc bổ sung.

**Câu 3(4 điểm): SỐ MAY MẮN**

Steve rất thích các số may mắn. Chúng ta biết rằng những số may mắn là những số nguyên dương mà trong dạng biểu diễn thập phân chỉ chứa số 4, số 7 hoặc cả 4 và 7. Ví dụ các số 44, 744, 4, 7 là may mắn và 5, 17, 467 là các số không may mắn.

Đặt next(x) là số may mắn nhỏ nhất mà lớn hơn hoặc bằng x. Steve quan tâm đến giá trị biểu thức sau: next(l) + next(l + 1) + ... + next(r - 1) + next(r).

**Yêu cầu**: Bạn hãy giúp Steve tính giá trị biểu thức trên.

**Dữ liệu** vào cho trong tệp LUCKYSUM.INP gồm một dòng duy nhất chứa hai số nguyên l và r (1 ≤ l ≤ r ≤ 109) là giới hạn bên trái và phải trong biểu thức.

**Kết quả** đưa ra tệp LUCKYSUM.OUT gồm một số duy nhất là kết quả của biểu thức next(l) + next(l + 1) + ... + next(r - 1) + next(r).

**Ví dụ:**

|  |  |
| --- | --- |
| **LUCKYSUM.INP** | **LUCKYSUM.OUT** |
| 2 7 | 33 |

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ **HẾT**\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_