|  |  |
| --- | --- |
| **SỞ GIÁO DỤC VÀ ĐÀO TẠO**  **QUẢNG TRỊ**  **ĐỀ THI CHÍNH THỨC** | **KỲ THI CHỌN HỌC SINH GIỎI VĂN HÓA LỚP 12 THPT**  **Năm học 2024 - 2025**  **MÔN THI: TIN HỌC**  *Thời gian làm bài: 180 phút, không kể thời gian phát đề* |

*(Đề thi có 03 trang)*

**TỔNG QUAN VỀ ĐỀ THI**

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **Câu** | **Tên bài** | **Tên tệp chương trình** | **Tên tệp INPUT** | **Tên tệp OUTPUT** | **Điểm** |
| Câu 1 | Đồng xu may mắn | COINLK.\* | COINLK.INP | COINLK.OUT | 5,0 |
| Câu 2 | Số ba ước | NUMBER.\* | NUMBER.INP | NUMBER.OUT | 5,0 |
| Câu 3 | Du lịch | TRAVEL.\* | TRAVEL.INP | TRAVEL.OUT | 5,0 |
| Câu 4 | Nén chuỗi | WSTRING.\* | WSTRING.INP | WSTRING.OUT | 5,0 |

* *Dấu \* là* ***CPP*** *hoặc* ***PY*** *hoặc* ***PAS*** *tùy theo ngôn ngữ lập trình được lựa chọn;*
* *Thời gian thực hiện mỗi test tùy thuộc vào từng bài, thường không quá 1s;*
* *Thí sinh tạo trên ổ đĩa D thư mục có tên là số báo danh, làm bài và lưu vào thư mục vừa tạo;*

*Ví dụ: Thí sinh có số báo danh là 10 sẽ tạo thư mục D:\10 và lưu bài làm vào thư mục này.*

**Câu 1. Đồng xu may mắn**

Ở một vương quốc xa xôi, có một trò chơi rất được ưa chuộng tại cung điện Hoàng gia. Trò chơi này được gọi là "Tung đồng xu may mắn". Mỗi lần chơi, họ sẽ tung một số đồng xu và ghi lại kết quả của mỗi lần tung. Nếu đồng xu có mặt sấp thì họ sẽ ghi lại ký tự ‘*S’*, còn nếu có mặt ngửa thì họ sẽ ghi lại ký tự ‘N’ và kết quả thu được một chuỗi ký tự P.

Sau mỗi ván chơi, nhà vua thường yêu cầu các quần thần tính toán tỷ lệ giữa số lần xuất hiện của mặt sấp và mặt ngửa để đánh giá vận may của mình. Tuy nhiên, không phải ai cũng nhanh nhạy trong việc tính toán tỷ lệ này. Vì vậy, nhà vua đã yêu cầu bạn, một vị cố vấn thông minh, viết một chương trình để tự động tính toán tỷ lệ này dưới dạng một phân số tối giản. Ví dụ: P = ‘SSNSN’ thì tỉ lệ giữa sấp và ngửa là 3/2.

**Yêu cầu:** Cho một chuỗi kết quả P từ trò chơi, in ra tỷ lệ giữa số lần xuất hiện của mặt sấp và mặt ngửa dưới dạng phân số tối giản.

**Dữ liệu:** Đọc từ tệp văn bản **COINLK.INP** có dạng:

- Dòng đầu tiên chứa số nguyên dương T (1≤ T ≤ 10) là số lượng chuỗi kết quả P;

- T dòng tiếp theo mỗi dòng chứa một chuỗi ký tự P (2 ≤ |P| ≤ 100000) chỉ bao gồm ký tự ‘S’ và ‘N’ (ký hiệu |P| là độ dài của chuỗi ký tự P). Dữ liệu đảm bảo có ít nhất một mặt sấp và ít nhất một mặt ngửa.

**Kết quả:** Ghi ra tệp văn bản **COINLK.OUT:**

Gồm T dòng theo thứ tự các chuỗi trong tệp dữ liệu vào, mỗi dòng gồm hai số nguyên A, B được viết dưới dạng A/B.

**Ví dụ:**

|  |  |
| --- | --- |
| **COINLK.INP** | **COINLK.OUT** |
| 3  SNSSSS  SNSSNNSSN  SNN | 5/1  5/4  1/2 |

**Ràng buộc:**

* *Có 30% số test tương ứng với 30% số điểm có 2≤ ≤ 255;*
* *Có 70% số test tương ứng với 70% số điểm không có ràng buộc gì thêm.*

**Câu 2. Số ba ước**

Minh rất thích những số có 3 ước. Một hôm, Minh đố Bảo rằng: Minh cho Bảo 2 số nguyên dương L và R(L ≤ R)*,* hỏi trong đoạn nguyên [L, R] có bao nhiêu số có 3 ước?

**Yêu cầu**: Đếm số lượng số nguyên dương có 3 ước trong đoạn nguyên [L, R].

**Dữ liệu:** Đọc từ tệp văn bản **NUMBER.INP** có 1 dòng chứa 2 số nguyên L và R cách nhau một dấu cách.

**Kết quả:** Ghi ra tệp văn bản **NUMBER.OUT** một số nguyên duy nhất là kết quả của bài toán.

**Ví dụ:**

|  |  |
| --- | --- |
| **NUMBER.INP** | **NUMBER.OUT** |
| 1 30 | 3 |
| 12 100 | 2 |

**Ràng buộc:**

* *Có 60% số test tương ứng với 60% số điểm có 1 ≤ L ≤ R ≤ 104;*
* *Có 20% số test tương ứng với 20% số điểm có 1 ≤ L ≤ R ≤ 1010;*
* *Có 20% số test tương ứng với 20% số điểm có 1 ≤ L ≤ R ≤ 1014.*

**Câu 3. Du lịch**

Minh và Bảo đang đi du lịch tại nước Pháp. Ở đây có n thành phố, tất cả nằm dọc theo một con đường cao tốc. Các thành phố được đánh số liên tiếp bắt đầu từ 1 đến n. Khoảng cách từ thành phố thứ i đến vị trí bắt đầu con đường cao tốc (d0= 0) là di (i=1, 2,…, n).

Minh đang ở thành phố x, Bảo ở thành phố y (1 ≤ x, y ≤ n). Hai bạn muốn tìm một thành phố z để gặp nhau sao cho là nhỏ nhất.

**Yêu cầu:** Cho d1, d2,…, dn và . Hãy tìm giá trị nhỏ nhất của .

**Dữ liệu:** Đọc từ tệp văn bản **TRAVEL.INP** có dạng:

* Dòng đầu chứa 2 số nguyên n, Q (2 ≤ n, Q ≤ 105);
* Dòng thứ hai chứa n số nguyên dương d1, d2,…, dn (d1, d2,…, dn ≤ 109);
* Q dòng sau, mỗi dòng chứa 2 số x, y mô tả cho một câu hỏi (1 ≤ x, y ≤ n).

Các số trên một dòng được viết cách nhau bởi một dấu cách.

**Kết quả:** Ghi ra tệp văn bản **TRAVEL.OUT:**

Gồm Q dòng, mỗi dòng là một giá trị nhỏ nhất tìm được tương ứng với một câu hỏi theo thứ tự trong tệp dữ liệu vào.

**Ví dụ:**

|  |  |  |  |
| --- | --- | --- | --- |
| **TRAVEL.INP** | **TRAVEL.OUT** | **TRAVEL.INP** | **TRAVEL.OUT** |
| 5 2 1 2 3 4 5 1 5 2 3 | 2  1 | 5 2  15 28 48 56 100  1 5  2 4 | 44  20 |

**Ràng buộc:**

* *Có 40% số test tương ứng với 40% số điểm có n,* ***Q***≤ 100 *;*
* *Có 30% số test tương ứng với 30% số điểm có n,* ***Q***≤ 105 *và* d1< d2<…<dn*;*
* *Có 30% số test tương ứng với 30% số điểm có n,* ***Q***≤ 105.

**Câu 4. Nén chuỗi**

Bạn là một kỹ sư phần mềm tại một công ty chuyên phát triển hệ thống gửi email tự động. Hệ thống này cần gửi đi hàng triệu email mỗi ngày, nhiều email chứa các chuỗi ký tự dài lặp lại. Việc gửi những chuỗi dài như vậy làm tăng đáng kể chi phí băng thông và thời gian xử lý.

Bạn quyết định áp dụng một thuật toán nén đơn giản để giảm độ dài chuỗi: Tìm kiếm các đoạn lặp lại trong chuỗi và thay thế chúng bằng một biểu diễn ngắn gọn hơn. Phương pháp nén này sử dụng các chữ số để biểu thị số lần lặp lại, kèm theo đoạn lặp được đặt trong dấu ngoặc đơn. Ví dụ: Chuỗi ‘abababaaaaa’ có thể được nén thành ‘3(ab)5(a)’, trong đó ‘3(ab)’ biểu thị rằng chuỗi ‘ab’ xuất hiện 3 lần liên tiếp, ‘5(a)’ biểu thị rằng ký tự ‘a’ xuất hiện 5 lần liên tiếp.

Định nghĩa cú pháp nén:

* <L> là một ký tự chữ thường trong khoảng từ '' đến '';
* <D> là một số nguyên lớn hơn hoặc bằng , biểu thị số lần lặp lại của chuỗi con bên trong dấu ngoặc đơn;
* <R> có thể là một ký tự <L> duy nhất hoặc một chuỗi nén bằng cách sử dụng một số nguyên <D> và một phần tử <S>;
* <S> biểu diễn một chuỗi ký tự đã nén, có thể bao gồm một hoặc nhiều phần tử <R>.

**Yêu cầu:** Hãy viết chương trình để tìm ra cách nén chuỗi đầu vào A có độ dài là N sao cho biểu diễn kết quả có độ dài ngắn nhất. Nếu có nhiều cách nén chuỗi đầu vào A với độ dài ngắn nhất, hãy đưa ra cách có thứ tự từ điển nhỏ nhất.

**Dữ liệu:** Đọc từ tệp văn bản **WSTRING.INP** có dạng:

* Dòng đầu tiên gồm một số nguyên N (1 ≤ N ≤ 200) là độ dài của chuỗi đầu vào A;
* Dòng thứ hai gồm chuỗi đầu vào A, chuỗi A chỉ bao gồm các ký tự Latin thường.

**Kết quả:** Ghi ra tệp văn bản **WSTRING.OUT**: Một chuỗi ký tự, là chuỗi nén theo yêu cầu.

**Ví dụ:**

|  |  |
| --- | --- |
| **WSTRING.INP** | **WSTRING.OUT** |
| 15  zzzzzzzzzzzzzzz | 15(z) |
| 4  aaaa | 4(a) |
| 3  bbb | bbb |
| 7  abcdefg | abcdefg |
| 24  abababcaaaaaabababcaaaaa | 2(3(ab)c5(a)) |
| 11  abababaaaaa | 3(ab)5(a) |

**Giải thích:**

* Testcase 2, chúng ta có 2 cách biểu diễn ngắn nhất, một là giữ nguyên ‘aaaa’ với độ dài là 4; hai là nén thành ‘4(a)’ cũng với độ dài 4. Cả 2 cách đều có độ dài là 4, tuy nhiên cách thứ 2 lại cho ta thứ tự từ điển bé hơn, vậy nên kết quả là ‘4(a)’;
* Testcase 3, chúng ta giữ nguyên sẽ có độ dài còn nếu chúng ta nén thành ‘3(b)’ thì sẽ có độ dài là .

**Ràng buộc:**

* *Có 60% số test tương ứng với 60% số điểm có* 1 ≤ N ≤ 10*;*
* *Có 20% số test tương ứng với 20% số điểm có* 5 ≤ N ≤ 25 *và chuỗi A chỉ bao gồm ký tự ‘a’, ‘b’;*
* *Có 20% số test tương ứng với 20% số điểm không có ràng buộc gì thêm.*

**---------HẾT---------**

*Họ và tên thí sinh:……………………………….……………….Số báo danh:…………………….*