# ẢNH HOA Tên file: FLOWERS.CPP hoặc FLOWERS.PAS

Steve có nhiệm vụ trang trí lối vào vườn thực vật của trường. Với sự lựa chọn công phu và chăm sóc chu đáo cả ***n*** bụi hoa chạy dọc theo đường thẳng dẫn tới vườn đều sống và nở hoa đồng thời, trông rất hấp dẫn. Bụi thứ ***i*** tính từ đầu đường là loại hoa ***ai***, ***i*** = 1 ÷ ***n***. Các bạn trong trường thường đến đứng trước luống hoa chụp ảnh đưa lên facebook. Theo kinh nghiệm của Steve, muốn được nhiều người like thì phần luống hoa nền ở đằng sau phải không chứa 3 bụi hoa liên tiếp cùng loại và Steve đưa lên trang Web của trường độ dài của đoạn luống hoa dài nhất nên đưa vào khung hình khi chụp ảnh.



Hãy xác định độ dài của đoạn luống hoa dài nhất được nêu trong trang Web.

***Dữ liệu:*** Vào từ file văn bản ***FLOWERS.INP***:

* Dòng đầu tiên chứa một số nguyên ***n*** (1 ≤ ***n*** ≤ 105),
* Dòng thứ 2 chứa ***n*** số nguyên ***a1***, ***a2***, . . ., ***an*** (1 ≤ ***ai***, ≤ 109, ***i*** = 1 ÷ ***n***).

***Kết quả:*** Đưa ra file văn bản ***FLOWERS.OUT*** trên một dòng 1 số nguyên – độ dài của đoạn luống hoa dài nhất tìm được.

***Ví dụ:***

|  |  |
| --- | --- |
| **FLOWERS.INP** | **FLOWERS.OUT** |
| 6  5 6 6 6 23 9 | 4 |

Giải thích: độ dài dài nhất của luống hoa là 6 6 23 9

# Sơn nhà Tên file: HOUSE.CPP hoặc HOUSE.PAS

Trên một con đường có *n* tòa nhà. Mỗi tòa nhà được sơn bằng một màu Đỏ, Vàng, Xanh, hoặc Tím. Người ta muốn sơn lại một số tòa nhà (bằng một trong các màu Đỏ, Vàng, Xanh, Tím) sao cho không có hai tòa nhà liên tiếp có cùng màu sơn. Tìm số nhà cần sơn lại ít nhất.

**Input: HOUSE.INP**

• Dòng đầu tiên chứa số nguyên dương *n* (1 ≤ *n* ≤ 2500).

• Dòng tiếp theo chứa một xâu kí tự có độ dài *n*, mỗi kí tự đại diện cho một tòa nhà. Mỗi kí tự có thể là *D*, *V*, *X*, hoặc *T*, lần lượt đại diện cho một ngôi nhà được sơn màu Đỏ, Vàng, Xanh, hoặc Tím.

**Output: HOUSE.OUT**

• In ra số lượng nhà cần sơn lại ít nhất.

Ví dụ:

|  |  |
| --- | --- |
| **HOUSE.INP** | **HOUSE.OUT** |
| 3  TXV | 0 |
| 17  DDTVVTDVVVTVVTDVV | 5 |

# MUA HÀNG Tên file: BUY.CPP hoặc BUY.PAS

Một công ty muốn mua m máy tính. Sau khi lấy thông tin tại n cửa hàng (1 ≤ n ≤ 10000), người ta biết được rằng cửa hàng thứ i có bán ai máy tính và với giá mỗi máy tính là bi. (ai, bi là những số nguyên dương: ai ≤ 100; bi ≤ 2000).

***Giả sử rằng các cửa hàng có đủ máy để bán cho công ty. Hãy tìm cách mua rẻ nhất.***

**Dữ liệu:** Vào từ file văn bản BUY.INP

* Dòng 1: Chứa hai số m, n cách nhau ít nhất một dấu cách.
* n dòng tiếp theo, dòng thứ i chứa hai số ai, bi cách nhau ít nhất một dấu cách.

**Kết quả:** Ghi ra file văn bản BUY.OUT

* Dòng 1: Ghi tổng số tiền phải trả.
* n dòng tiếp theo, dòng thứ i ghi số máy tính mua ở cửa hàng thứ i.

**Ví dụ:**

|  |  |  |
| --- | --- | --- |
| BUY.INP |  | BUY.OUT |
| 22 5  3 30  5 10  6 8  10 5  2 20 |  | 168  0  5  6  10  1 |

# Trò chơi với dãy số Tên file: SEQGAME.CPP hoặc SEQGAME.PAS

Hai bạn học sinh trong lúc nhàn rỗi nghĩ ra trò chơi sau đây. Mỗi bạn chọn trước một dãy số gồm n số nguyên. Giả sử dãy số mà bạn thứ nhất chọn là: *b1, b2, ..., bn* còn dãy số mà bạn thứ hai chọn là *c1, c2, ..., cn*

Mỗi lượt chơi mỗi bạn đưa ra một số hạng trong dãy số của mình. Nếu bạn thứ nhất đưa ra số hạng bi (1 <= i <= n), còn bạn thứ hai đưa ra số hạng cj (1 ≤ j ≤ n) thì giá của lượt chơi đó sẽ là |bi+cj|.

Ví dụ: Giả sử dãy số bạn thứ nhất chọn là 1, -2; còn dãy số mà bạn thứ hai chọn là 2, 3. Khi đó các khả năng có thể của một lượt chơi là (1, 2), (1, 3), (-2, 2), (-2, 3). Như vậy, giá nhỏ nhất của một lượt chơi trong số các lượt chơi có thể là 0 tương ứng với giá của lượt chơi (-2, 2).

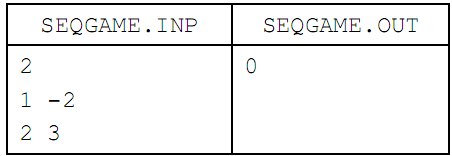
**Yêu cầu:** Hãy xác định giá nhỏ nhất của một lượt chơi trong số các lượt chơi có thể.

**INPUT:** vào từ file văn bản SEQGAME.INP

* Dòng đầu là số nguyên dương (1 ≤ n ≤ 105)
* Dòng thứ hai chứa các số là dãy b (|bi| ≤ 109)
* Dòng thứ hai chứa các số là dãy c (|ci| ≤ 109)

**OUTPUT:** ghi ra file văn bản SEQGAME.OUT giá trị nhỏ nhất tìm được

**Ví dụ:**



***Ràng buộc:*** 60% số test ứng với 60% số điểm có 1<= n <= 1000

# Ghép số lớn Tên file: NUMCON.CPP hoặc NUMCON.PAS

Vaxia đã viết được một số lớn trên một cuộn giấy dài và muốn khoe với anh trai Petia về thành quả vừa đạt được. Tuy nhiên,  khi Vaxia vừa ra khỏi phòng để gọi anh trai thì cô em Kachia chạy vào phòng và xé rách cuộn giấy thành một số mảnh. Kết quả là trên mỗi mảnh có một hoặc vài kí số theo thứ tự đã viết.

Bây giờ Vaxia không thể nhớ chính xác mình đã viết số gì. Vaxia chỉ nhớ rằng đó là một số rất lớn.

Để làm hài lòng cậu em trai, Petia quyết định truy tìm  số nào là lớn nhất mà Vaxia đã có thể viết lên cuộn giây trước khi bị xé. Bạn hãy giúp Petia làm việc này.

**Dữ liệu vào: từ file NUMCON.INP:**

Ghi một hoặc nhiều dòng. Mỗi dòng ghi một dãy kí số. Số dòng không vượt quá 100. Mỗi dòng ghi từ 1 đến 100 kí số. Bảo đảm rằng có ít nhất một dòng mà kí số đầu tiên khác 0.

**Dữ liệu ra: ghi ra file NUMCON.INP:** số lớn nhất đã có thể viết trên cuộn giấy trước khi bị xé rách.

Ví dụ:

|  |  |
| --- | --- |
| **NUMCON.INP** | **NUMCON.OUT** |
| 2  20  004  66 | 66220004 |