

Bài 1. Xin chào

Nam là người thích chat với bạn bè trên Internet. Cậu ấy đã lập ra một phòng chat với điều kiện rằng trước khi vào phòng chat, mọi người phải chào hỏi trước.

Một câu chào được định nghĩa rằng, câu chào đó phải là một xâu kí tự, chỉ gồm các chữ cái, không chứa kí tự trắng, sao cho khi xóa đi một số chữ cái, nó sẽ trở thành từ “hello”, tất nhiên là sẽ không được phép tráo đổi vị trí các chữ cái, mà chỉ được xóa bớt một số chữ cái.

Ví dụ khi Bình muốn vào phòng chat, Bình gõ “ahhellllloou” thì hệ thống sẽ xem xét xâu này và tự động loại bỏ các chữ cái để trở thành từ “hello”. Như vậy Bình được vào phòng chat.

Nhưng khi Bình gõ “HellhcLudoo”, hệ thống không thể làm cách nào xóa bớt chữ cái để trở thành từ “hello” được. Như vậy, Bình không được vào phòng chat.

Yêu cầu: cho câu chào, hãy xác định xem câu chào có được chấp nhận hay không.

Dữ liệu: vào từ tệp HELLO.INP gồm một dòng là câu chào mà Bình gõ vào, số lượng kí tự không vượt quá 200.

Kết quả: ghi ra tệp HELLO.OUT ghi “YES” nếu câu chào được chấp nhận, ghi “NO” nếu câu chào không hợp lệ.

Ví dụ:

HELLO.INP	HELLO.OUT
ahhellllloou	YES
HellhcLudoo	NO

Bài 2. Điền vào chỗ trống

Cho đẳng thức $a + b = c$, trong 3 số này có một số bị mờ đi một chữ số (được thay bằng dấu ?), hãy tìm chữ số đó.

Dữ liệu: vào từ tệp FILL.INP gồm 3 dòng:

- + Dòng đầu chứa một số nguyên không âm a.
- + Dòng thứ hai chứa một số nguyên không âm b.
- + Dòng thứ ba chứa một số nguyên không âm c.

Cả ba số đều không vượt quá 10^6 , một trong ba số sẽ bị thay một vị trí bằng dấu ?.

Kết quả: ghi ra tệp FILL.OUT gồm một dòng chứa chữ số cần tìm.

FILL.INP	FILL.OUT
128	
?	7
200	

Bài 3. Định lý 6174

Cho 1 số nguyên dương N1 (có 4 chữ số không đồng thời bằng nhau).

Số $N_2=a_1-b_1$ (a_1 là số gồm 4 chữ số của N_1 được sắp xếp giảm dần từ trái sang phải, b_1 là số gồm 4 chữ số của N_1 được sắp xếp tăng dần từ trái sang phải).

Số $N_3=a_2-b_2$ (a_2 là số gồm 4 chữ số của N_2 được sắp xếp giảm dần từ trái sang phải, b_2 là số gồm 4 chữ số của N_2 được sắp xếp tăng dần từ trái sang phải).

...

Dãy số kết thúc khi $N_k=6174$.

Chú ý: Chữ số 0 ở đầu số vẫn có nghĩa.

Ví dụ: $N_1 = 1234$

$$N_2 = 4321 - 1234 = 3087$$

$$N_3 = 8730 - 0378 = 8352$$

$$N_4 = 8532 - 2358 = 6174$$

Yêu cầu: Cho trước 1 số nguyên dương N_1 có 4 chữ số không đồng thời bằng nhau. Hãy viết chương trình tìm số k để $N_k = 6174$.

Dữ liệu: vào từ tệp DL6174.INP gồm 1 dòng chứa số N_1 .

Kết quả: ghi ra tệp DL6174.OUT gồm 1 dòng chứa số k .

DL6174.INP	DL6174.OUT
1234	4
1549	4

Bài 4. Biến đổi xâu

Cho trước một xâu nhị phân có độ dài bất kỳ. Cần biến đổi xâu nhị phân này về dạng toàn số 0.

Các phép biến đổi chỉ có thể là một trong các loại sau:

+ Biến đổi xâu con 11 thành 00.

+ Biến đổi xâu con 010 thành 000.

Yêu cầu: Hãy chỉ ra một cách biến đổi xâu đã cho thành xâu có toàn số 0 sao cho số phép biến đổi là ít nhất.

Dữ liệu: vào từ tệp CHASTR.INP xâu nhị phân có độ dài d ($1 \leq d \leq 200$) trong đó có ít nhất một số 1.

Kết quả: ghi ra tệp CHASTR.OUT một số nguyên là số phép biến đổi, nếu không biến đổi được thì ghi -1.

Ví dụ:

CHASTR.INP	CHASTR.OUT
01011110	3

Bài 5. Đường đi trên ma trận

Cho một bảng số kích thước $M \times N$. Bảng số có M dòng được đánh số từ trên xuống dưới và N cột đánh số từ trái sang phải. Cặp tọa độ (i, j) biểu diễn ô có vị trí dòng i cột j . Mỗi ô (i, j) chứa một số nguyên $A[i, j]$ ($|A[i, j]| \leq 30000$). Bắt đầu từ 1 ô ở cột đầu tiên, ta di chuyển đến ô kế tiếp cho đến khi đến cột N . Từ ô (i, j) ta có thể di chuyển đến 1 trong 3 ô $(i-1, j+1)$ hoặc $(i, j+1)$ hoặc $(i+1, j+1)$. Chú ý là ta chỉ có thể đến ô (i, j) nếu $1 \leq i, j \leq N$. Trọng số của đường đi là tổng các số trên đường đi đó. Hãy tìm đường đi có trọng số lớn nhất bắt đầu từ một ô bất kì ở cột 1 đến một ô bất kì ở cột N .

Dữ liệu: vào từ tệp BANG.INP

+ Dòng đầu chứa 2 số M, N ($1 \leq M, N \leq 100$).

Kết quả: ghi ra tệp BANG.OUT một số duy nhất là trọng số tìm được.

Ví dụ:

BANG.INP	BANG.OUT
4 5 3 2 6 5 9 7 1 0 6 4 3 1 1 7 8 2 5 9 8 4 3 1	42

Giải thích:

3	2	6	5	9
7	10	6	4	3
11	7	8	2	5
9	8	4	3	1

Bài 6. Robot

Cho một bảng vuông ($n \times n$) ô ($2 \leq n \leq 50$). Trong mỗi ô có ghi số 0 hoặc 1. Tìm đường đi của Robot từ góc trái trên xuống góc phải dưới theo nguyên tắc chỉ được dịch chuyển sang phải hoặc xuống dưới sao cho các số trên đường đi tạo thành một số nhị phân nhỏ nhất.

Dữ liệu: vào từ tệp ROBOT.INP gồm:

+ Dòng đầu tiên là giá trị n.

+ n dòng tiếp theo, trên mỗi dòng ghi n số (0 hoặc 1), các số này cách nhau ít nhất một khoảng trắng.

Kết quả: ghi ra tệp .ROBOT.OUT là một số nhị phân có giá trị nhỏ nhất.

Ví dụ:

ROBOT.INP	ROBOT.OUT
6 1 0 1 0 1 0 1 1 0 0 1 1 1 1 1 0 1 0 1 1 1 1 0 0 1 1 1 0 0 0 1 1 1 1 1 1	10100010001