

CONTEST 03

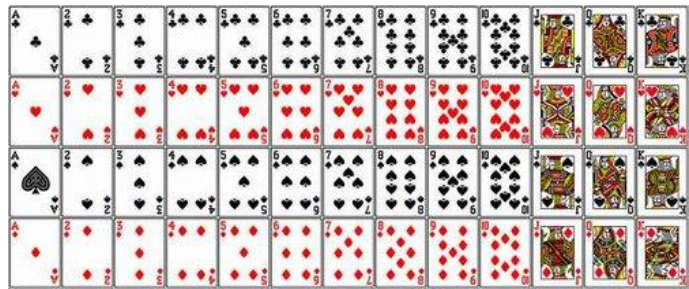
(01/6/2020)

<i>Bài</i>	<i>CODE</i>	<i>Input</i>	<i>Output</i>	<i>Time</i>
Poker	POKER.*	POKER.INP	POKER.OUT	1s
Bàn tiệc	TABLE.*	TABLE.INP	TABLE.OUT	1s
Cuộc gặp	MEETING.OUT	MEETING.OUT	MEETING.OUT	1s
Ternary	TERNARY.*	TERNARY.INP	TERNARY.OUT	2s

Bài 1: Bài Poker

Một nhà máy sản xuất bài Poker có một dây chuyền đang bị lỗi nên một bộ bài khi được sản xuất ra có thể dư hoặc thiếu những con bài.

Bộ bài Poker gồm 52 con bài được chia thành 4 chất được ký hiệu bằng 4 chữ cái



P, K, H và T. Mỗi quân bài được ký hiệu bởi một chữ cái tương ứng với chất của nó và một số có hai chữ số viết liền nhau. Phần số có giá trị từ 1 đến 13, nếu số đó là số có 1 chữ số thì thêm chữ số 0 ở trước. Ví dụ P07 (7 chuồn), T12 (già bích).

Yêu cầu: Cho một chuỗi là các ký hiệu của một bộ bài gồm nhiều quân bài, bạn hãy kiểm tra xem bộ bài này có bị lỗi không?

Dữ liệu vào: Tệp văn bản POKER.INP chỉ ghi một chuỗi là các ký hiệu của các quân bài. Các ký hiệu này được viết liền nhau.

Dữ liệu ra: Ghi ra tệp văn bản POKER.OUT theo yêu cầu: Nếu trong bộ bài xuất hiện hai quân bài giống nhau thì ghi số -1. Ngược lại ghi 4 số nguyên trên cùng một dòng tương ứng với số quân bài của mỗi chất bị thiếu theo thứ tự (P, K, H, T).

POKER.INP	POKER.OUT
P01K02H03H04	12 12 11 13

POKER.INP	POKER.OUT
H02H10P11H02	-1

Bài 2: Bàn tiệc năm mới

Bờm đang sắp xếp một cái bàn tiệc cho bữa tiệc năm mới, mặt bàn có hình dạng là một hình tròn có bán kính R. Bờm đã mời rất nhiều khách và đang phân vân liệu cái bàn có đủ chỗ để đặt những cái đĩa cho các vị khách này không? Lưu ý các đĩa đều hình tròn và có bán kính là r. Mỗi đĩa đều phải

được đặt bên trong bàn và tiếp xúc với cạnh bàn và tất nhiên các đĩa không được đặt chồng nhau nhưng có thể tiếp xúc nhau. Hãy giúp Bờm xác định xem bàn có đủ chỗ để đặt N cái đĩa hay không?

Dữ liệu vào: file TABLE.INP gồm một dòng duy nhất ghi 3 số nguyên n , R và r cách nhau một dấu cách.

Kết quả: file TABLE.OUT ghi “YES” nếu có thể đặt được n cái đĩa lên bàn với quy tắc được nêu ở trên còn ngược lại ghi “NO”.

Ví dụ

TABLE.INP	TABLE.OUT
4 10 4	YES

TABLE.INP	TABLE.OUT
5 10 4	NO

Bài 3: Cuộc gặp

Thanh và Lê cùng ở trong một thành phố có nhiều khu phố, nhà Thanh ở khu phố A , nhà Lê ở khu phố B . Cuối tuần cả hai bạn được nghỉ nên Thanh dự định đến khu phố B để thăm Lê. Để đến nhà Lê, Thanh phải đi qua các khu phố được nối với nhau bởi những con đường hai chiều. Mỗi con đường nối trực tiếp giữa hai khu phố có thể có tuyến xe buýt. Việc di chuyển qua nhiều khu phố có thể mất nhiều thời gian nên Thanh phải tìm ra một hành trình bằng xe buýt để đi từ A đến B với thời gian nhanh nhất có thể.

Yêu cầu: Cho biết thành phố mà Thanh và Lê đang ở có N khu phố được đánh số từ 1 đến N (bao gồm cả A và B). Nếu giữa hai khu phố bất kỳ u và v có tuyến xe buýt thì phải đi mất t thời gian và tốn tiền vé là x . Hiện tại Thanh chỉ có số tiền là K . Bạn hãy tính xem, liệu Thanh có còn dư tiền sau đi xe buýt đến nhà Lê hay không. Nếu có thì thời gian ít nhất để Thanh đi từ nhà mình đến nhà Lê là bao nhiêu?

Dữ liệu vào: Từ tệp văn bản MEETING.INP gồm:

+ Dòng đầu tiên ghi 3 số nguyên K , N , M cho biết K là số tiền mà Thanh có, N là số khu phố, M là số lượng các cặp khu phố có tuyến xe buýt.

+ Trong M dòng tiếp theo mỗi dòng ghi 4 số nguyên u , v , t , x cho biết thông tin về tuyến xe buýt giữa khu phố u và v mất t thời gian và tốn số tiền là x .

+ Dòng cuối cùng ghi hai số nguyên A và B là số thứ tự khu phố mà hai bạn đang ở. Các số trên cùng 1 dòng cách nhau ít nhất 1 ký tự trắng.

Kết quả: Ghi ra tệp văn bản MEETING.OUT một số nguyên duy nhất là tổng thời gian nhỏ nhất tìm được thỏa yêu cầu của bài toán, nếu không có cách đi thì ghi số -1.

Ví dụ:

MEETING.INP	MEETING.OUT
10 4 7 1 2 4 4 1 3 7 2 3 1 8 1 3 2 2 2 4 2 1 6 3 4 1 1 1 4 6 12 1 4	7
Đường đi: $1 \rightarrow 2 \rightarrow 3 \rightarrow 4$	

MEETING.INP	MEETING.OUT
3 3 3 1 2 5 1 3 2 8 2 1 3 1 3 1 3	-1
Không thể đi từ 1 qua 3	

Giới hạn:

- + Trong tất cả các test: $1 \leq K \leq 200$; $2 \leq N \leq 2000$; $1 \leq M \leq 10\,000$; $1 \leq t \leq 105$; $0 \leq x \leq 200$
- + Có 20% số điểm tương ứng với $K=1$ và $N \leq 200$
- + Có 20% số điểm tương ứng với $K=1$ và $200 < N \leq 2000$

Bài 4: Máy tính hệ tam phân

Petya rất thích máy tính. Gần đây, anh đã nhận được một máy tính “Ternatron IV” thế hệ mới là một món quà từ mẹ của mình. Không giống như các máy tính hiện đại khác, “Ternatron IV” hoạt động với hệ tam phân, không giống như hệ nhị phân. Ngay lập tức Petya tự hỏi các phép *xor* được thực hiện như thế nào trên máy tính loại này và liệu có bất cứ điều gì giống như hệ nhị phân không.

Nó đã đưa thông báo rằng phép toán *xor* không tồn tại, tuy nhiên nó được gọi là phép *tor* và nó làm việc như thế này: giả sử chúng ta cần tính giá trị của biểu thức $a \text{ tor } b$. Nếu hai số có số chữ số khác nhau thì thêm các chữ số 0 vào đầu số ngắn hơn để hai số có độ dài bằng nhau sau đó các chữ số được cộng lại với nhau theo từng chữ số. Lưu ý rằng kết quả được modulo cho 3 và không nhớ để chuyển cho số khác.

Ví dụ: $14_{10} \text{ tor } 50_{10} = 0112_3 \text{ tor } 1212_3 = 1021_3 = 34_{10}$

Petya viết số a và c trên một mảnh giấy và đưa cho bạn, hãy giúp anh ta tìm số b sao cho $a \text{ tor } b = c$.

Dữ liệu vào: file văn bản TERNARY.INP chỉ một dòng ghi số a và c ($0 \leq a, c \leq 10^9$) cách nhau một dấu cách.

Kết quả: file văn bản TERNARY.OUT chỉ ghi số nguyên b duy nhất ở hệ thập phân. Nếu có nhiều hơn một kết quả thì in kết quả nhỏ nhất.

Ví dụ

TERNARY.INP
14 34

TERNARY.OUT
50

Giới hạn thời gian 2 giây