

# BÀI CHO ĐỘI TUYỂN NĂM 2020

Ngày 10 tháng 11 năm 2020

## Bóng bay

Có  $n$  quả bóng bay, mỗi quả có một trong số 4 màu  $\{R, B, G, Y\}$  xếp thành một cột. Ở mỗi bước được phép chọc thủng dây bóng cùng màu từ 2 quả trở lên. Nếu dây bị chọc thủng có  $k$  quả thì điểm số nhận được ở bước đi đó là  $k^2$ . Các quả bóng ở trên sẽ rơi xuống lấp chỗ trống trong cột.

Ví dụ, cột ban đầu có 10 bóng, tính từ trên xuống cột có cấu hình **BBGGBBYYY**. Để chọc thủng hết dây bóng, ta có thể chọc các quả liên tiếp **BB**, **GG**, **BB**, **YYY**. Khi đó, điểm số nhận được sẽ là  $2^2 + 2^2 + 2^2 + 3^2 = 21$ . Có cách khác để nhận được tổng điểm số cao hơn là **GG**, **BBBB**, **YYY** với tổng điểm  $2^2 + 4^2 + 3^2 = 29$ .

**Yêu cầu:** Cho chuỗi  $S$  độ dài  $n$  chỉ chứa các ký tự trong tập  $\{R, B, G, Y\}$  xác định màu các quả bóng tính từ trên xuống dưới. Hãy xác định tổng điểm số lớn nhất nhận được khi phá hết bóng trong cột. Nếu không thể phá hết bóng thì tổng điểm là 0.

**Dữ liệu:** Vào từ file văn bản BALLOON.INP, gồm nhiều tests, mỗi test cho trên một dòng chứa chuỗi  $S$ .

**Kết quả:** Đưa ra file văn bản BALLOON.OUT, kết quả mỗi test đưa ra trên một dòng dưới dạng số nguyên.

**Ví dụ:**

BALLOON.INP	BALLOON.OUT
BBGGBBYYY	29
BGGGB	13
BGGBGGGG	24
GBGB	0

Sub 1:  $n \leq 15$

Sub 2:  $n \leq 200$ ; chỉ có 2 màu  $G, B$

Sub 3:  $n \leq 200$

## Thoát hiểm

Một hành lang của một tòa nhà có  $n$  cửa cách đều nhau đánh số từ 1 tới  $n$ . Trong số đó có một số cửa thoát hiểm. Khi sự cố xảy ra, mọi người từ các vị trí gần các cửa sẽ đổ ra cửa thoát hiểm gần nhất. Bạn cần lập một kế hoạch thoát hiểm là một dãy các số nguyên  $p_1, p_2, \dots, p_n$  với  $p_i$  là chỉ số của cửa thoát hiểm mà những người đang ở vị trí cửa  $i$  cần di chuyển tới. Một cặp  $(i, j)$  ( $i \leq j$ ) được coi là xung đột nếu như  $p_i = p_j$  và khoảng cách từ  $i$  và  $j$  tới cửa thoát hiểm được chọn là như nhau. Hay nói cách khác  $p_i - i = j - p_j$ .

Yêu cầu: Hãy lập một kế hoạch thoát hiểm để đảm bảo số cặp xung đột là ít nhất có thể.

**Dữ liệu:** vào từ file EMERGENCY.INP

- Dòng đầu tiên chứa số nguyên dương  $n$
- Dòng thứ hai chứa  $n$  số nguyên  $t_1, t_2, \dots, t_n$  với  $t_i = 0$  tương ứng với cửa thứ  $i$  là cửa thoát hiểm.  $t_i = 1$  trong trường hợp ngược lại.

**Kết quả:** Ghi ra file EMERGENCY.OUT

- Dòng đầu chứa số nguyên  $S$  xác định số cặp xung đột nhỏ nhất tìm được.
- Dòng thứ 2 là các số nguyên dương  $p_1, p_2, \dots, p_n$  xác định kế hoạch đưa ra.

**Ví dụ:**

EMERGENCY.INP	EMERGENCY.OUT
12	3
0 1 1 1 0 1 1 1 0 1 1	1 1 1 5 5 5 5 10 10 10 10

**Ràng buộc:**

- 30% số test có  $n \leq 200$  và mỗi cửa chỉ có duy nhất 1 cửa gần nhất
- 30% số test khác có  $n \leq 200$
- 40% số test còn lại có  $n \leq 10^5$ .

## Festivals

Có  $n$  lễ hội, lễ hội thứ  $i$  được tổ chức từ ngày  $l_i$  tới ngày  $r_i$ . An xin nghỉ phép và dự định đi chơi vào bắt đầu từ ngày  $x$ , và các thời điểm tiếp theo cách nhau  $d$  ngày ( $x, x + d, x + 2d, \dots$ ). Vào ngày không có lễ hội, An có thể thành thoi ngủ nướng cả ngày.

**Yêu cầu:** Hãy giúp An lựa chọn ngày bắt đầu  $x$  để số lễ hội An có thể tham gia là nhiều nhất. Nếu có nhiều ngày có thể chọn, đưa ra số hiệu ngày có giá trị nhỏ nhất.

**Dữ liệu:** vào từ file FESTIVALS.INP

- Dòng đầu tiên chứa 2 số nguyên dương  $n, d$
- $n$  dòng tiếp theo, dòng thứ  $i$  chứa 2 số nguyên không âm  $l_i, r_i$  xác định ngày bắt đầu và ngày kết thúc. Các ngày đánh số từ 0.

**Kết quả:** Ghi ra file FESTIVALS.OUT 2 số nguyên dương là số lễ hội lớn nhất mà An có thể tham gia và ngày bắt đầu An dự định đi chơi.

**Ví dụ:**

FESTIVALS.INP	FESTIVALS.OUT
4 4	3 1
5 5	
0 2	
10 11	
15 23	

**Giải thích:** An có thể tham gia lễ hội 1 (ngày 5), lễ hội 2 (ngày 1), lễ hội 4 (ngày 17,21).

**Ràng buộc:**

- 30% số test có  $n, d \leq 10^3; l_i, r_i \leq 10^5$
- 30% số test khác có  $n, d \leq 10^3$
- 40% số test còn lại có  $n \leq 10^5; d, l_i, r_i \leq 10^{15}$

## Tắc đường

Thành phố Alpha có  $n$  nút giao thông với  $m$  con đường hai chiều nối giữa các nút. Vào giờ cao điểm, lượng xe di chuyển đông nên thường xuyên xảy ra tắc nghẽn tại một số nút hoặc giữa một số cung đường. Tại những thời điểm như vậy, người dân thường nhờ đến sự trợ giúp của hệ thống điều hành giao thông thông minh được triển khai trực tuyến. Để hỗ trợ được tốt cho các lái xe, hệ thống thường phải tính toán trả lời một trong 2 loại truy vấn:

- 1  $A\ B\ C\ D$ : Có thể di chuyển từ nút  $A$  tới nút  $B$  khi con đường nối giữa  $C$  và  $D$  bị tắc hay không?
- 2  $A\ B\ C$ : Có thể di chuyển từ nút  $A$  tới nút  $B$  khi nút giao thông  $C$  bị tắc hay không?

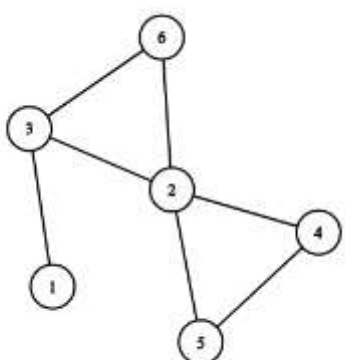
Hãy viết chương trình thực hiện tính toán giải quyết các yêu cầu của hệ thống.

**Dữ liệu:** vào từ file JAM.INP

- Dòng đầu tiên chứa 2 số nguyên dương  $n, m$  ( $n \leq 10^5, m \leq 5 \cdot 10^5$ )
- $m$  dòng tiếp theo, mỗi dòng chứa 2 số nguyên  $u, v$  xác định đường nối 2 chiều trực tiếp từ  $u$  tới  $v$ . Không có 2 thành phố nào có nhiều hơn 1 đường nối.
- Dòng tiếp theo chứa số nguyên dương  $Q$  ( $Q \leq 3 \cdot 10^5$ ) xác định số truy vấn.
- $Q$  dòng tiếp theo, mỗi dòng chứa thông tin truy vấn hệ thống nhận được.

**Kết quả:** ghi ra file JAM.OUT  $Q$  dòng, mỗi dòng là câu trả lời cho từng truy vấn theo thứ tự trong file input. Đưa ra "yes" nếu vẫn di chuyển được, đưa ra "no" trong trường hợp ngược lại.

**Ví dụ:**

JAM.INP	JAM.OUT	Minh họa
6 7 1 3 3 6 4 2 5 4 2 5 2 6 2 3 4 1 1 6 3 6 1 1 6 1 3 2 3 5 2 2 3 5 6	yes no no yes	

## Nhảy hố

Đạt rất lười học và rất thích chơi. Lần này Đạt không được chơi game trên máy tính mà thầy giáo cho Đạt chơi một trò như sau:

Có  $n$  hố được đánh số  $1, 2, 3, \dots, n$  từ trái qua phải. Hố thứ  $i$  có một năng lượng là  $a_i$  ( $a_i > 0$ ). Khi một quả bóng bay vào hố  $i$ , bóng sẽ bị bắn nhảy ngay sang hố  $i + a_i$ . Tới khi bóng bị bay vượt quá khỏi hàng. Có  $m$  lượt chơi. Với mỗi lượt chơi, người chơi đưa ra 1 trong 2 hành động:

- 0  $i$   $x$ : Biến đổi năng lượng của hố thứ  $i$  thành  $x$ .
- 1  $i$ : Thả quả bóng vào hố thứ  $i$ , hỏi sau bao nhiêu bước nhảy, bóng sẽ bay khỏi hàng và hố cuối cùng bóng bay vào là hố nào?

**Yêu cầu:** Đưa ra câu trả lời với mỗi hành động loại 1.

**Dữ liệu:** vào từ file **JUMPOUT.INP**

- Dòng đầu chứa 2 số nguyên dương  $N, M$  ( $N, M \leq 10^5$ )
- Dòng thứ 2 chứa  $n$  số nguyên  $a_1, a_2, \dots, a_n$
- $M$  dòng tiếp, mỗi dòng một truy vấn mô tả một hành động loại 0 hoặc 1.

**Kết quả:** Ghi ra file **JUMPOUT.OUT** đưa ra chỉ số của hố cuối cùng trước khi bóng bay ra khỏi hàng và số bước nhảy của quả bóng với các truy vấn loại 1 theo thứ tự.

JUMPOUT . INP	JUMPOUT . OUT
8 5	8 7
1 1 1 1 1 2 8 2	8 5
1 1	7 3
0 1 3	
1 1	
0 3 4	
1 2	

**Ràng buộc:**

- 50% số test có  $n, m \leq 1000$

## Ghép dãy

Với 2 dãy số  $X_1 X_2 \dots X_n, Y_1, Y_2, \dots, Y_m$  Ta định nghĩa phép toán  $\times$  với hai dãy như sau:

Dãy  $Z = X \times Y$  là dãy gồm  $m \times n$  phần tử được sinh ra từ tổng  $m \times n$  cặp số, một số xuất hiện bên dãy  $X$ , một số bên dãy  $Y$ .

Ví dụ:  $X = \{1,2\}, Y = \{1,2,3\}$ . Ta có các phần tử dãy  $Z$  là  $1 + 1, 1 + 2, 1 + 3, 2 + 1, 2 + 2, 2 + 3$ .  $\rightarrow Z = \{2,3,4,3,4,5\}$

Cho 2 dãy số  $A_1 A_2 \dots A_n, B_1 B_2 \dots B_m$ . Xây dựng các dãy:

$$C = A \times A$$

$$D = B \times B$$

$$E = C \times D$$

Dễ thấy số phần tử của các dãy  $C, D, E$  lần lượt là  $n^2, m^2, n^2 \times m^2$ .

**Yêu cầu:** Tính tổng  $K$  phần tử lớn nhất của dãy  $E$ .

**Dữ liệu:** vào từ file **SUMKMAX.INP**

- Dòng đầu chứa 3 số nguyên dương  $n, m, k$  ( $k \leq n^2 m^2$ )
- Dòng thứ hai chứa  $n$  số nguyên dương  $A_1 A_2 \dots A_n$
- Dòng cuối cùng là  $m$  số nguyên dương  $B_1 B_2 \dots B_m$  ( $A_i, B_i \leq 10^6$ ).

**Kết quả:** Ghi ra file **SUMKMAX.OUT** một số nguyên duy nhất là tổng của  $K$  phần tử lớn nhất trong dãy  $E$

**Ví dụ:**

SUMKMAX.INP	SUMKMAX.OUT
2 2 1	4
1 1	
1 1	

**Ràng buộc:**

- 25% test  $n, m \leq 30$
- 25% test  $30 \leq n, m \leq 1000, k \leq 2.10^5$
- 25% test tiếp theo  $n, m \leq 400$
- 25% test còn lại  $n, m \leq 1000$