

**HỘI THI TIN HỌC TRẺ TỈNH ĐẮK NÔNG LẦN THỨ XIII – NĂM 2020**  
**BẢNG C – TRUNG HỌC PHỔ THÔNG**

*Ngày thi: ngày 11 tháng 7 năm 2020*

**ĐỀ THI THỨ**

*(Đề thi gồm có 4 trang)*

**TỔNG QUAN ĐỀ THI**

Bài	Tên bài	File chương trình	File dữ liệu	File kết quả	Điểm
1	Dãy số	SEQLEN.*	SEQLEN.INP	SEQLEN.OUT	5
2	Tiền	MONEY.*	MONEY.INP	MONEY.OUT	5
3	Tính hàm $F_m$	FM.*	FM.INP	FM.OUT	4
4	Lập trình Logo	LOGO.*	LOGO.INP	LOGO.OUT	3
5	Đèn trang trí	LAMPS.*	LAMPS.INP	LAMPS.OUT	3

Dấu \* được thay thế bởi PAS hoặc CPP của ngôn ngữ lập trình sử dụng tương ứng là Pascal hoặc C++

**Bài 1. Dây số**

Người ta viết liên tục các số tự nhiên không chứa khoảng cách từ 1 trở đi tạo thành một dãy các chữ số như sau 123456789101112131415...

**Yêu cầu** cho số nguyên dương  $n \leq 10^{18}$ . Hỏi dãy số tạo được bằng cách như trên có chiều dài bao nhiêu chữ số?

**Dữ liệu** nhập từ bàn phím số nguyên dương  $n$

**Kết quả** xuất ra màn hình số nguyên dương là chiều dài của số tạo được

**Ví dụ:**

SEQLEN.INP
15

SEQLEN.OUT
21

**Bài 2. TIỀN**

Hóa ra ai cũng cần tiền, kể cả phù thủy. Họ sử dụng các đồng vàng, bạc và đồng, gọi tương ứng là Galeon, Sikel và Knut. Một Galeon ăn 17 sikel, một sikel ăn 29 knut. Mọi giá cả nêu sau đều theo các đơn vị kể trên. Trong mỗi giá số sikel không quá 16, số knut – không quá 28.

Trước khi vào nhập học ở Hogvard Harry Potter rút ở ngân hàng Gringot một số tiền để mua một số học cụ cần thiết như đũa thần, cú, chậu thiếc, áo choàng, ... Số tiền Harry rút ra là  $g$  Galeo,  $s$  Sikel và  $k$  Knut. Harry cần mua tất cả là  $n$  thứ. Vật thứ  $i$  có giá là  $(p_i, q_i, r_i), i = 1 \div n, (0 \leq n \leq 10^5)$ .

**Yêu cầu:** Hãy xác định số tiền Harry còn lại sau khi sắm mọi thứ. Nếu Harry không đủ tiền thì đưa ra số -1.

**Dữ liệu:** vào từ tập tin văn bản **MONEY.INP**

- Dòng đầu tiên chứa 3 số nguyên  $g, s, k (0 \leq g \leq 10^5)$
- Dòng thứ 2 chứa số nguyên  $n$
- Dòng thứ  $i$  trong  $n$  dòng sau chứa 3 số nguyên  $p_i, q_i, r_i (0 \leq p_i \leq 10^5)$

**Kết quả:** xuấtra tập tin văn bản **MONEY.OUT** trên một dòng 3 số nguyên xác định số tiền còn lại của Harry hoặc số -1.

**Ví dụ:**

MONEY.INP	MONEY.OUT
5 16 10	2 2 18
2	
3 9 21	
0 4 0	

### Bài 3. Tính hàm $F_m$

Cho dãy số gồm  $n$  số nguyên dương  $a_1, a_2, \dots, a_n$  và một số nguyên  $m$ . Người ta định nghĩa hai hàm số :

- $p_k$  = số lượng cặp chỉ số  $(i, j)$  sao cho  $a_i + a_j = k$  ( $1 \leq i, j \leq n$ )
- $F_k = p_1 + p_2 + \dots + p_k$

**Yêu cầu :** Cho số  $m$ . hãy tính  $H_m$ .

**Dữ liệu:** Từ file văn bản **FM.INP** có cấu trúc như sau :

- Dòng đầu gồm 2 số nguyên dương  $m, n \leq 10^5, m \leq 10^6$
- Dòng tiếp theo gồm  $n$  số nguyên dương  $a_1, a_2, \dots, a_n$  ( $\forall i : a_i \leq 10^6$ ) các số ghi trên một dòng và cách nhau ít nhất một dấu cách.

**Kết quả :** Ghi ra file văn bản **FM.OUT** một số nguyên duy nhất là giá trị của  $F_m$ .

**Ví dụ :**

FM.INP	FM.OUT
5 6	6
1 2 3 4 5	

- **Ràng buộc:** 50% số điểm ứng với các test có  $m, n \leq 1000$   
Giải thích :  $F_6 = p_1 + p_2 + \dots + p_6 = 0 + 0 + 1 + 1 + 2 + 2 = 6$

### Bài 4. Lập trình Logo

Logo là một ngôn ngữ lập trình được xây dựng để học lập trình qua việc điều khiển con rùa robot. Các câu lệnh trong ngôn ngữ để điều khiển con rùa di chuyển. Con rùa có một cây bút gắn với nó. Khi rùa di chuyển, nó vẽ các đường thẳng trên trang giấy. Con rùa có thể được lập trình để vẽ những bức tranh thú vị.

Mít muốn điều khiển con rùa vẽ một bức tranh, sau đó quay trở lại điểm mà nó xuất phát. Ví dụ, Mít có thể cung cấp cho rùa chương trình sau:

**fd 100 lt 120 fd 100 lt 120 fd 100**

Ban đầu, rùa ở tọa độ  $(0,0)$  và hướng sang bên phải. Lệnh **fd** làm cho con rùa di chuyển về phía trước một số đơn vị được chỉ định. Lệnh **lt** làm cho con rùa rẽ trái so với hướng hiện tại một số độ được chỉ định. Do đó, các lệnh trên làm cho con rùa vẽ một hình tam giác đều với các cạnh dài 100 đơn vị. Lưu ý rằng sau khi thực hiện các lệnh, con rùa kết thúc ở cùng một vị trí mà nó xuất phát. Con rùa còn hiểu hai lệnh nữa: Lệnh **bk** khiến cho rùa di chuyển lùi theo số đơn vị được chỉ định; Lệnh **rt** làm cho con rùa rẽ phải theo một số độ được chỉ định.

Sau khi thực hiện nhiều lệnh, con rùa có thể bị lạc, cách xa vị trí xuất phát của nó. Nhiệm vụ của bạn là xác định khoảng cách đường thẳng từ vị trí của con rùa khi kết thúc hành trình trở lại vị trí nó xuất phát.

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

Dòng đầu tiên chứa một số nguyên là số lượng bộ test. Mỗi bộ test bắt đầu bằng một dòng chứa một số nguyên là số lượng lệnh cần thực hiện. Tất cả khoảng cách và độ là số nguyên. Các lệnh tiếp theo, một lệnh trên mỗi dòng. Mỗi bộ test sẽ chứa không quá 1000 lệnh.

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

Đối với mỗi bộ test, ghi ra khoảng cách từ vị trí bắt đầu đến vị trí cuối cùng, được làm tròn đến số nguyên gần nhất.

**Ví dụ:**

LOGO.INP	LOGO.OUT
1	0
5	
fd 100	
lt 120	
fd 100	
lt 120	
fd 100	

## Bài 5: Đèn trang trí

Một hệ thống đèn trang trí gồm  $n$  đèn được đánh số từ 1 đến  $n$  và  $n-1$  đoạn dây nối điều khiển, mỗi đoạn nối một cặp hai đèn khác nhau. Hệ thống dây nối điều khiển thỏa mãn tính chất sau đây: Không có đoạn dây nào nối một đèn với chính nó; không có hai đoạn dây nào nối cùng một cặp đèn và hơn nữa không tìm được dãy các đèn  $v_1, v_2, \dots, v_k, v_1$ , trong đó hai đèn liên tiếp là có đoạn dây nối và không có đoạn dây nối nào xuất hiện quá một lần. Tại mỗi thời điểm, mỗi đèn sẽ sáng màu xanh hoặc đỏ. Bộ điều khiển hệ thống đèn có thể thực hiện tác động nhiều lần việc thay đổi trạng thái các đèn, mỗi lần tác động là thay đổi màu của một đèn nào đó và tất cả các đèn có dây nối với nó, cụ thể nếu đèn đang sáng màu xanh sẽ chuyển sang sáng màu đỏ, ngược lại nếu đèn đang sáng màu đỏ sẽ chuyển sang sáng màu xanh. Vì lý do kỹ thuật, hệ thống đã bổ sung thêm một đoạn dây nối giữa hai đèn  $i, j$  (giữa hai đèn này chưa có dây nối).

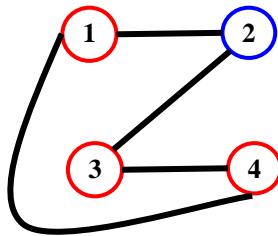
**Yêu cầu:** Cho biết trạng thái ban đầu về màu của  $n$  đèn và thông tin về các dây nối điều khiển ban đầu và dây nối bổ sung, hãy tìm cách điều khiển để tất cả các đèn sáng màu xanh.

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

- Dòng đầu chứa hai số nguyên dương  $n, T$  lần lượt là số lượng đèn, số trường hợp thử nghiệm;
- Dòng thứ  $k$  trong số  $n-1$  dòng tiếp theo chứa thông tin về đoạn dây nối điều khiển thứ  $k$  bao gồm hai số nguyên  $u_k, v_k$ , trong đó  $u_k, v_k$  là chỉ số của hai đèn là các đầu mút của đoạn dây nối điều khiển thứ  $k$  ( $k = 1, 2, \dots, n-1$ );
- Tiếp theo là một dòng chứa hai số nguyên  $i, j$ ;
- Dòng thứ  $i$  trong số  $T$  dòng cuối cùng chứa  $n$  số  $c_{i1}, c_{i2}, \dots, c_{in}$ , trong đó  $c_{ij}$  là màu của đèn thứ  $j$  trong trường hợp thử nghiệm thứ  $i$  ( $c_{ij} = 1$  nếu đèn  $j$  sáng màu xanh và  $c_{ij} = 0$  nếu đèn  $j$  sáng màu đỏ),  $j = 1, 2, \dots, n$ ;  $i = 1, 2, \dots, T$ .

**Kết quả:** Đưa ra file văn bản LAMPS.OUT  $T$  dòng, mỗi dòng là phương án điều khiển tìm được cho trường hợp thử nghiệm tương ứng trong dữ liệu vào, theo khuôn dạng sau: Ghi số -1 nếu không tồn tại cách điều khiển thỏa mãn. Ngược lại số đầu tiên của dòng là số  $s$ ; tiếp theo là  $s$  số  $l_1, l_2, \dots, l_s$  mô tả cách điều khiển, trong đó tác động thứ  $h$  ( $h = 1, 2, \dots, s$ ) làm đảo màu của đèn  $l_h$  và các đèn có dây nối với  $l_h$ .

**Ví dụ:**

LAMPS.INP	LAMPS.OUT	LAMPS.OUT
<b>4 1</b> <b>1 2</b> <b>2 3</b> <b>3 4</b> <b>1 4</b> <b>0 1 0 0</b>	<b>1 4</b>	

**Ràng buộc:**

- Có 30% số test ứng với 30% số điểm của bài có  $n = 4$  mà đèn 1, 2, 3, 4 lần lượt được nối thành 1 vòng tròn,  $T = 1$ .
- Có 20% số test khác ứng với 20% số điểm của bài có  $n \leq 20$ ,  $T = 1$ .
- Có 20% số test khác ứng với 20% số điểm của bài có  $n \leq 30$ ,  $T \leq 5$ .
- Có 10% số test khác ứng với 10% số điểm của bài có  $n \leq 300$ ,  $T \leq 50$ .
- Có 20% số test còn lại với 20% số điểm của bài có  $n \leq 3000$ ,  $T \leq 500$ .

----- **HẾT** -----

*(Thí sinh không được sử dụng tài liệu. Cán bộ coi thi không giải thích gì thêm)*

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