

SPRING CONTEST DAY 2018 (Ngày 2)

Lưu ý: Các thí sinh có thể nộp bằng ngôn ngữ Pascal, C++ hoặc Java tùy theo ngôn ngữ nào thí sinh thấy thuận tiện. Tên các bài là Tenbai.*, trong đó * tương ứng với pas, cpp hay java. Tenbai là tên của bài sẽ được quy định trong cụ thể từng bài.

Bài 4 (26 điểm – SPC4.*):

Khi nghe hay tin anh Nhật Khôi rớt thi học sinh giỏi tỉnh, lớp 10Tin quyết định tổ chức party để ăn mừng. Lớp 10Tin sẽ kêu một miếng bánh pizza bự ơ là bự bởi vì đây là một sự kiện cực kì đáng mừng. Lớp 10Tin gồm N bạn, và không biết đây là định mệnh hay là sự trùng hợp, chữ cái bắt đầu tên của tất cả các thành viên trong lớp 10Tin đều khác nhau. Miếng bánh pizza có thể được coi như một miếng bánh hình chữ nhật. N bạn sẽ cùng vào phân chia chiếc bánh pizza và mỗi bạn chắc chắn phải có một phần bánh dành cho mình. Mỗi phần bánh cho mỗi bạn là một miếng hình chữ nhật và ta không cần quan tâm đến kích thước của những miếng bánh. Trước khi phân chia, chiếc bánh sẽ được chia như một ma trận hình chữ nhật gồm R hàng và C cột cách đều nhau, ô ở hàng i và cột j là ô (i, j). Mỗi bạn sẽ đánh dấu chữ cái đầu tiên của tên mình vào một ô mà bạn đó thích, đồng thời nếu ô nào đã có bạn nào đánh dấu rồi thì ô đó sẽ không được bạn khác đánh dấu. Nhiệm vụ của lớp trưởng là hãy phân chia chiếc bánh này thành những miếng hình chữ nhật sao cho thỏa mãn điều kiện sau:

1. Mỗi bạn đều phải nhận được một miếng bánh.
2. Những miếng bánh dành cho mỗi bạn đều phải là hình chữ nhật.
3. Ký tự mà mỗi bạn ban đầu đánh dấu lên chiếc bánh, thì đều phải thuộc về miếng bánh mà bạn đó nhận.
4. Toàn bộ chiếc bánh pizza ban đầu phải được phân chia hết và không còn phần dư.

Hãy giúp lớp trưởng chia chiếc bánh pizza này, nếu như có nhiều cách chia thì bạn in ra một cách thỏa mãn là được.

- Dữ liệu đầu vào (SPC4.INP):

- Đọc từ file SPC4.INP
- Dòng đầu tiên gồm một số nguyên dương T ($1 \leq T \leq 200$) là số lượng bộ dữ liệu của đề bài.
- T bộ dữ liệu tiếp theo, mỗi bộ dữ liệu được tổ chức như sau:
 - Dòng đầu tiên gồm 2 số nguyên dương R và C lần lượt là số dòng và số cột của miếng bánh pizza.
 - R dòng tiếp theo, mỗi dòng gồm C ký tự, ký tự thứ j trên dòng i gồm 2 loại ký tự:

- Ký tự ở ô (i, j) là dấu '?', khi ô (i, j) của chiếc bánh ban đầu chưa được một bạn nào đánh dấu chữ cái ban đầu của tên mình.
 - Ký tự ở ô (i, j) là một ký tự in hoa c trong bảng chữ cái Alphabet, thể hiện rằng ở ô (i, j) có một bạn có chữ cái bắt đầu tên là c đánh dấu vào ô đó.
 - Input bảo đảm các ký tự chữ cái trên bảng hình chữ nhật là hoàn toàn khác nhau.
- **Dữ liệu đầu ra (SPC4.OUT):**
- Ghi ra file SPC4.OUT
 - Với bộ dữ liệu thứ i, ta in ra R dòng và C cột. Ký tự ở ô (i, j) là một chữ cái c thể hiện ô bánh (i, j) thuộc về bạn có chữ cái bắt đầu tên là c.
- **Ví dụ:**

SPC4.INP	SPC4.OUT
4	DDD
3 3	NNN
D??	TTT
?N?	DNK
??T	AAAAA
1 3	AAAAA
DNK	AAAAA
3 5	AAABB
?????	AAABB
??A??	DDCCC
?????	DDCCC
4 5	
??A??	
???B?	
??C??	
D????	

- **Giải thích test ví dụ:**
- Với test ví dụ 1: Bạn tên D đã chọn ô (1, 1), bạn tên N chọn ô (2, 2), bạn tên T chọn ô (3, 3). Như vậy ta có thể chia miếng bánh như output đã trình bày ở trên.
 - Với test ví dụ 2: Có 3 bạn và có 3 ô bánh, như vậy mỗi bạn tên 'D', 'N', 'K' mỗi bạn 1 ô.
 - Với test ví dụ 3: Lớp chỉ có 1 bạn và bạn này tên A, như vậy toàn bộ phần bánh là của bạn í.
 - Với test ví dụ 4: Bạn đọc hãy tự hiểu cách phân chia.
- **Bài toán chỉ có duy nhất 1 Dataset:**
- $1 \leq R, C \leq 25$.
- **Giới hạn thời gian và bộ nhớ:**

- 10s / Dataset.
- 512Mb / Dataset.

Bài 5 (40 điểm – SPC5.*):

Bạn được cho một số nguyên dương N và K . Bạn hãy đếm xem có bao nhiêu hoán vị của các số từ 1 đến N có đúng K vị trí tốt. Với một hoán vị p gồm N số $\{p_1, p_2, \dots, p_N\}$, ta định nghĩa “vị trí tốt” là một vị trí i thỏa mãn $|p_i - i| = 1$. $|x|$ là giá trị tuyệt đối của số x .

- **Dữ liệu đầu vào (SPC5.INP):**
 - Đọc từ file SPC5.INP.
 - Dòng đầu tiên gồm một số nguyên dương T ($1 \leq T \leq 20$) là số lượng bộ dữ liệu cần xử lý
 - T bộ dữ liệu tiếp theo, mỗi bộ dữ liệu được tổ chức như sau:
 - Gồm một dòng chứa 2 số nguyên dương N và K .
- **Dữ liệu đầu ra (SPC5.OUT):**
 - Ghi vào file SPC5.OUT.
 - Gồm T dòng, dòng thứ i là kết quả của bộ dữ liệu thứ i , là số lượng hoán vị thỏa mãn có đúng K “vị trí tốt”. Vì kết quả rất lớn nên ta in kết quả là phần dư cho số $10^9 + 7$.
- **Ví dụ:**

SPC5.INP	SPC5.OUT
7	1
2 0	0
3 1	1
1 0	42
5 2	77732
10 5	71644370
100 34	809845111
1000 800	

- **Giải thích test ví dụ:**
 - Ví dụ 1: Hoán vị thỏa mãn là $\{1, 2\}$. Hoán vị $\{2, 1\}$ không thỏa vì có tận 2 vị trí tốt.
 - Ví dụ 2: Ta có 4 hoán vị có 2 vị trí tốt là $\{1, 3, 2\}, \{2, 1, 3\}, \{2, 3, 1\}, \{3, 1, 2\}$. Hai hoán vị không có vị trí tốt nào là hoán vị $\{1, 2, 3\}, \{3, 2, 1\}$, như vậy không có hoán vị nào chỉ có 1 vị trí tốt.
 - Ví dụ 3: Hoán vị $\{1\}$ không có vị trí tốt nào nên số lượng là 1.
 - Những test ví dụ còn lại, dùng để bạn kiểm chứng chương trình của mình ra có đúng đáp án hay không.
- **Bài toán sẽ được chia thành 2 Dataset:**
 - Small Dataset (12 điểm) : $1 \leq K \leq N \leq 10$.

- Large Dataset (28 điểm): $1 \leq K \leq N \leq 1000$.
- **Giới hạn dữ liệu:**
 - 15s / Dataset.
 - 1024Mb / Dataset.

Bài 6 (34 điểm – SPC6.*):

Bạn đang sở hữu một bản đồ gồm N địa điểm trong thành phố. Bạn được giao việc xây dựng thêm những con đường một chiều nối giữa các cặp đỉnh sao cho thỏa mãn 2 điều kiện sau:

1. Không tồn tại một chu trình đơn trong sơ đồ bạn vừa xây được
 2. Số lượng đường đi đơn từ địa điểm 1 đến địa điểm N bằng đúng số M cho trước.
- **Dữ liệu đầu vào (SPC6.INP):**
 - Dòng đầu tiên gồm một số nguyên dương T ($1 \leq T \leq 100$), là số lượng bộ dữ liệu.
 - T bộ dữ liệu tiếp theo, mỗi bộ dữ liệu được tổ chức dưới dạng như sau:
 - Mỗi bộ dữ liệu chỉ gồm một dòng duy nhất, là 2 số nguyên dương N và M cách nhau bởi một dấu cách.
 - **Dữ liệu đầu ra (SPC6.OUT):**
 - Ta in ra đáp án với T bộ dữ liệu tương ứng, mỗi đáp án được tổ chức như sau:
 - Dòng đầu tiên ta in ra một chuỗi POSSIBLE hoặc IMPOSSIBLE, với ý nghĩa rằng ta có thể xây được đồ thị gồm N đỉnh hay không.
 - Nếu dòng đầu tiên là POSSIBLE, N dòng sau, mỗi dòng gồm N ký tự '0' hoặc '1'. Ký tự thứ j trên dòng i nếu là '1', nghĩa rằng có một đường 1 chiều nối giữa đỉnh i đến j , ngược lại thì không tồn tại.
 - Nếu có nhiều đáp án thỏa mãn, bạn có thể in ra một đáp án bất kì.
 - **Ví dụ:**

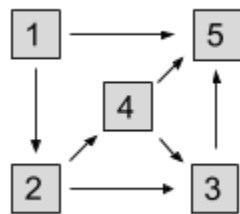
SPC6.INP	SPC6.OUT
4	POSSIBLE
5 4	01001
4 20	00110
2 1	00001
5 3	00101
	00000
	IMPOSSIBLE
	POSSIBLE
	01
	00
	POSSIBLE
	01000
	00110
	00001

	00101 00000
--	----------------

- **Giải thích test ví dụ:**

- o Với test ví dụ 1:

Ta có thể xây dựng một đồ thị như sau:



Ta có 4 đường đi đơn từ đỉnh 1 đến đỉnh 5 là: {1, 5}, {1, 2, 3, 5}, {1, 2, 4, 5}, {1, 2, 4, 3, 5}. Như ta thấy không hề tồn tại một chu trình đơn trong đồ thị.

- o Với test ví dụ 2:

Đồ thị này có 4 đỉnh. Giả sử đường đi là:

1 -> ? -> ? -> 4

Như vậy dấu ? ở trên chỉ có thể điền số 2 hoặc số 3 hoặc số 0. Nếu như ta điền số 0, điều đó có nghĩa 1 sẽ không nối với 0, mà sẽ nối tắt qua đỉnh tiếp theo. Ví dụ 1 -> 0 -> 3 -> 4 = 1 -> 3 -> 4.

Nếu ta nối 2 sang 3, mà lại có tiếp một cạnh 1 chiều từ 3 sang 2 thì sẽ vô tình tạo thành một chu trình. Như vậy, không mất tính tổng quát, ta có thể quy định dấu ? đầu tiên điền vào số 0 hoặc 2, dấu ? thứ hai điền vào số 0 hoặc số 3. Như vậy số lượng đường đi đơn trong đồ thị 4 đỉnh là $2 * 2 = 4$. Đó là những đường:

1 -> 0 -> 0 -> 4 = 1 -> 4

1 -> 0 -> 3 -> 4 = 1 -> 3 -> 4

1 -> 2 -> 0 -> 4 = 1 -> 2 -> 4

1 -> 2 -> 3 -> 4.

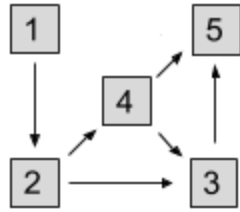
Như vậy ta chỉ có thể tạo tối đa 4 đường, nhưng $M = 20$ nên không thể tạo đồ thị được.

- o Với test ví dụ 3:

Ta tạo một đường duy nhất là nối từ đỉnh 1 sang đỉnh 2.

- o Với test ví dụ 4:

Là đồ thị ở test ví dụ 1 nhưng bỏ đi 1 đường {1, 5}.



- **Bài toán sẽ được chia làm 2 Dataset:**
 - Small Dataset (16 điểm): $2 \leq N \leq 6, 1 \leq M \leq 20$.
 - Large Dataset (18 điểm): $2 \leq N \leq 50, 1 \leq M \leq 10^{18}$.
- **Giới hạn dữ liệu:**
 - 5s / Dataset.
 - 512Mb / Dataset.

-----Chúc các bạn làm bài tốt <3 <3 <3 -----