

ĐỀ CHÍNH THỨC
(Đề thi có 04 trang)

Thời gian làm bài: 180 phút
Ngày thi: 08/9/2021

TỔNG QUAN BÀI THI

	Tên bài	File chương trình	File dữ liệu đầu vào	File dữ liệu đầu ra
Câu 1	XẾP HẠNG	BRANK.*	BRANK.INP	BRANK.OUT
Câu 2	THÁM HIỂM LÒNG ĐẤT	MAPS.*	MAPS.INP	MAPS.OUT
Câu 3	NÓI ĐIỂM	WIRES.*	WIRES.INP	WIRES.OUT
Câu 4	TẬP BOI	SWIM.*	SWIM.INP	SWIM.OUT

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

Hãy lập trình để giải các bài toán sau:

Câu 1: XẾP HẠNG (5 điểm)

Có N học sinh, đánh số thứ tự $1, 2, \dots, N$, tham gia một cuộc thi lập trình. Các thí sinh được chấm bài ngay trong thời gian thi nên mỗi bạn đều biết điểm thi của mình. Ban tổ chức cho biết không có hai thí sinh nào có cùng điểm thi. Tuy nhiên, bảng xếp hạng cuối cùng lại chưa được công bố.

Vì muốn biết được thứ hạng để xác định màu huy chương, một số thí sinh đã so sánh điểm thi với nhau. Có tất cả M so sánh như vậy, kết quả mỗi so sánh là một cặp số nguyên $(A; B)$ được hiểu là thí sinh A có điểm cao hơn thí sinh B , đồng nghĩa A sẽ có thứ hạng kết quả cao hơn B .

Cho biết thông tin về các lần so sánh và chúng không gây mâu thuẫn.

Yêu cầu: Hãy xác định xem có bao nhiêu thí sinh đã có thể biết chính xác thứ hạng của mình.

Dữ liệu: Vào từ file văn bản **BRANK.INP** gồm:

- Dòng 1: hai số nguyên N, M ($1 \leq N \leq 100; 1 \leq M \leq 4500$)
- Dòng 2 ... $M+1$: mỗi dòng là thông tin về một so sánh gồm hai số nguyên A, B ($1 \leq A, B \leq N; A \neq B$) chỉ thí sinh A sẽ có thứ hạng cao hơn thí sinh B .

Kết quả: Ghi ra file văn bản **BRANK.OUT** gồm: 1 số nguyên là số lượng thí sinh có thể xác định chính xác thứ hạng kết quả.

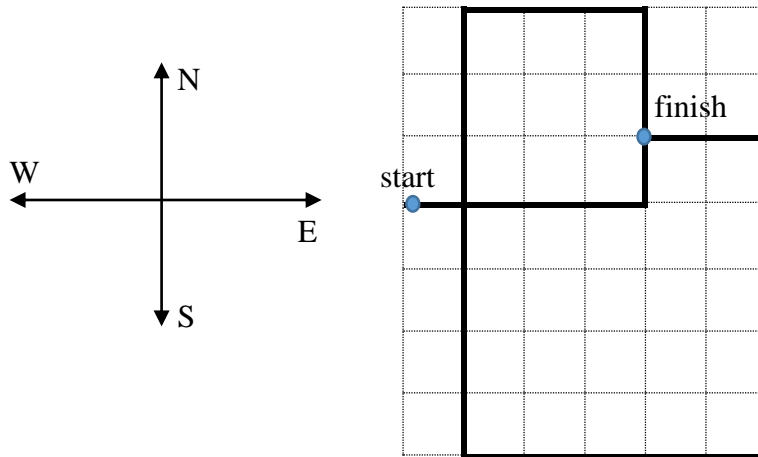
Ví dụ:

BRANK.INP	BRANK.OUT
5 5 4 3 4 2 3 2 1 2 2 5	2

Câu 2: THÁM HIỂM LÒNG ĐẤT (5 điểm)

Một nhà khảo cổ nghiên cứu những di sản văn hoá cổ đại ở một thành phố bị chôn vùi dưới lòng đất. Để thám hiểm thành phố đó, nhà khảo cổ của chúng ta buộc phải đào các đường ngầm.

Bắt đầu tại vị trí xuất phát, ông ta đào theo một trong 4 hướng Đông (E), Tây (W), Nam(S), Bắc (N), mỗi lần đào một đơn vị độ dài. Sau đó có thể đào tiếp theo hướng đó hoặc đổi hướng theo một trong 4 hướng trên. Giả sử rằng đường kính của đường ngầm đào được là không đáng kể. Để tránh bị lạc, ông ta ghi lại vào file văn bản MAPS.INP trong máy tính xách tay của mình một trong 4 ký tự E, W, S, N tương ứng với một trong bốn hướng mà ông ta sẽ đào tới mỗi lần. Ví dụ với điểm xuất phát và quy trình đào hàm dưới đây, sơ đồ các đường ngầm sẽ là:



Sau khi đã khảo sát xong, nhà khảo cổ muốn quay trở lại điểm xuất phát bằng đường ngầm đã đào.

Yêu cầu: Hãy dựa vào thông tin trong máy tính xách tay của nhà khảo cổ để chỉ cho ông ta đường đi ngắn nhất quay trở lại.

Dữ liệu: Vào từ file văn bản **MAPS.INP** của nhà khảo cổ gồm 1 dòng không quá 5000 ký tự $\in \{E, W, N, S\}$

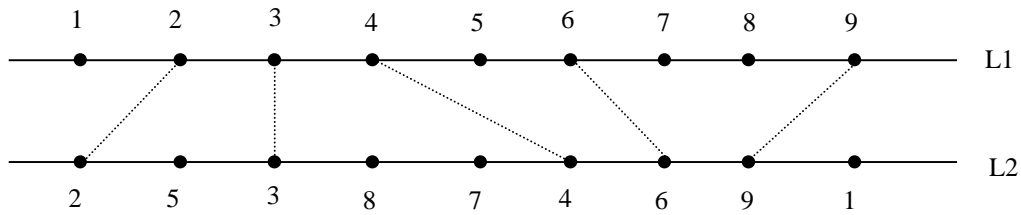
Kết quả: Ghi ra file văn bản **MAPS.OUT** gồm 1 dòng chứa các ký tự chỉ hướng đi dẫn về nơi xuất phát.

Ví dụ:

MAPS.INP	MAPS.OUT
EEEENNNWWSSSSSSSEEEEEENNNNNWW	SWWW

Câu 3: NỐI ĐIỂM (5 điểm)

Trên hai đường thẳng song song L1 và L2 người ta đánh dấu trên mỗi đường N điểm. Các điểm trên đường thẳng L1 được đánh số từ 1 đến N từ trái qua phải, còn các điểm trên đường thẳng L2 được đánh số bằng p_1, p_2, \dots, p_n cũng từ trái qua phải với p_1, p_2, \dots, p_n là một hoán vị của $1, 2, \dots, n$ (hình vẽ dưới đây là một ví dụ khi $n = 9$):



Ta gọi các số gán cho các điểm là số hiệu của chúng. Cho phép nối hai điểm trên hai đường thẳng có cùng số hiệu.

Yêu cầu: Tìm cách nối được nhiều cặp điểm nhất với điều kiện các đoạn nối không được cắt nhau.

Dữ liệu: Vào từ file văn bản **WIRES.INP** gồm:

- Dòng đầu tiên chứa số nguyên dương N ($N \leq 1000$).
- Dòng thứ hai chứa các số nguyên p_1, p_2, \dots, p_n cách nhau bởi dấu trắng.

Kết quả: Ghi ra file văn bản **WIRES.OUT** gồm:

- Dòng đầu tiên chứa k là số lượng các đoạn nối tìm được
- Dòng tiếp theo chứa k số hiệu của các đầu mút của các đoạn nối được ghi theo thứ tự tăng dần.

Ví dụ:

WIRES.INP	WIRES.OUT
9	5
2 5 3 8 7 4 6 9 1	2 3 4 6 9

Câu 4: TẬP BƠI (5 điểm)

Tại thành phố HT có một bể bơi duy nhất để phục vụ người dân. Bể bơi này chỉ có 2 làn. Người ta dùng mỗi làn để bơi theo một chiều. Để bơi 1 vòng bể, vận động viên sẽ bơi hết chiều dài bể rồi quay ngược lại. Vận động viên thứ I có tốc độ bơi là P_i nghĩa là sau P_i giây anh ta sẽ bơi hết chiều dài của bể. Sau $2 * P_i$ giây thì anh ta bơi hết một vòng (thời gian đổi làn được coi là không đáng kể). Trước buổi tập luyện, vận động viên thứ I đã dự kiến là sẽ bơi C_i vòng.

Do cơ sở vật chất hạn chế nên người ta phải đưa ra quy định như sau để đảm bảo tính an toàn:

- Các vận động viên không được phép vượt nhau khi đang bơi mà chỉ được vượt nhau tại 2 đầu bể khi đang đổi làn.
- Nếu một vận động viên bơi nhanh hơn nhưng bơi phía sau một người bơi chậm thì vận động viên đó sẽ phải giảm tốc độ bằng tốc độ của người bơi trước mình, giữ tốc độ đó cho tới khi đổi làn rồi mới được vượt lên bơi trước.

Tại thời điểm 0, có N vận động viên cùng xuất phát và người nào bơi nhanh hơn sẽ vượt lên trước ngay.

Bạn hãy tính xem sau bao nhiêu lâu thì người cuối cùng rời bể?

INPUT: SWIM.INP

- Dòng đầu tiên ghi số N ($1 \leq N \leq 50$)
- N dòng tiếp theo, mỗi dòng ghi 2 số P_i và C_i ($1 \leq P_i \leq 300, 1 \leq C_i \leq 250$)

OUTPUT: SWIM.OUT

- Ghi một số duy nhất là thời gian người cuối cùng rời bể?

Ví dụ:

SWIM.INP	SWIM.OUT
2 10 30 15 20	600
3 2 6 7 2 8 2	36

Hết

Họ tên thí sinh: Số báo danh:

Giám thị 1: Ký tên: Giám thị 2: Ký tên: