

ĐỀ THI CHÍNH THỨC

Môn: TIN HỌC

Thời gian làm bài: **180 phút** (không kể thời gian giao đề)

Ngày thi: **22/02/2022**

(Đề thi có 05 trang, gồm 05 bài)

TỔNG QUAN CÁC BÀI THI

	Tên bài	File chương trình	File dữ liệu vào	File kết quả
Bài 1.	SỐ SIÊU ĐẶC BIỆT	SUPNUM.*	SUPNUM.INP	SUPNUM.OUT
Bài 2.	ROBOT BẮY RUỒI	FLYTRAP.*	FLYTRAP.INP	FLYTRAP.OUT
Bài 3.	DÒ MẬT KHẨU	PASSWORD.*	PASSWORD.INP	PASSWORD.OUT
Bài 4.	CÁP TREO	CABCAR.*	CABCAR.INP	CABCAR.OUT
Bài 5.	HƯỚNG DẪN ĐƯỜNG ĐI	COVID.*	COVID.INP	COVID.OUT

*Dấu * được thay thế bởi PAS hoặc CPP của ngôn ngữ lập trình được sử dụng tương ứng là Pascal, C++ hoặc phần mở rộng tương ứng với NNLT khác.*

Bài 1: SỐ SIÊU ĐẶC BIỆT (4.0 điểm) – Tên chương trình: SUPNUM.*

Bạn An rất thích số nguyên tố đồng thời cũng yêu thích số 5. Do đó, bạn ấy luôn coi 1 số nguyên dương M mà có tổng các ước nguyên tố P ($P \neq 1$) của nó chia hết cho 5 là số siêu đặc biệt. Lần này, thầy giáo của An đưa cho bạn ấy 1 dãy số nguyên dương a_i . An rất muốn biết trong dãy đó có bao nhiêu số siêu đặc biệt nên nhờ các bạn tìm giúp.

Dữ liệu vào: từ tệp văn bản SUPNUM.INP gồm hai dòng:

- Dòng đầu tiên chứa số nguyên dương n ($2 \leq n \leq 500$), là số lượng các phần tử của dãy số nguyên dương a_i .
- Dòng thứ hai chứa dãy số nguyên dương a_i ($2 \leq a_i \leq 10^9$), mỗi số phân tách nhau bởi một dấu cách.

Dữ liệu ra: tệp văn bản SUPNUM.OUT, ghi dãy số siêu đặc biệt, mỗi số cách nhau một dấu cách. Nếu dãy số nguyên dương a_i không có số siêu đặc biệt thì ghi ra -1.

Ví dụ:

SUPNUM.INP	SUPNUM.OUT	GIẢI THÍCH
5 2 5 12 20 30	5 12 30	<ul style="list-style-type: none">Số 5 là số nguyên tố chia hết cho 5Số 12 có 2 ước nguyên tố là 2 và 3 ($2 + 3 = 5$ chia hết cho 5)Số 30 có 3 ước nguyên tố 2, 3 và 5 ($2 + 3 + 5 = 10$ chia hết cho 5)
6 2 20 100 200 15 45	-1	

Ràng buộc:

- Subtask #1: Có 30% số test ứng với $n \leq 10$;
- Subtask #2: Có 30% số test ứng với $n \leq 70$;
- Subtask #3: Có 40% số test ứng với $n \leq 500$;

Bài 2: ROBOT BẮY RUỒI (4.0 điểm) – Tên chương trình: FLYTRAP.*

Thời đại khoa học kỹ thuật phát triển mọi việc đều có thể giao cho các robot chuyên dụng làm việc. Các kỹ sư đã chế tạo ra một robot chuyên bắt ruồi. Robot có khả năng bắt ruồi rất nhanh và chính xác. Robot thiết kế là 1 cái “bẫy”, được tạo bởi miền trong của 1 hình tam giác (các cạnh của tam giác xem như thuộc miền trong của hình tam giác). Robot được cấp điện 1 chiều. Khi con ruồi bay vào “bẫy” sẽ bị điện giật chết. Để kiểm tra lại trước khi xuất xưởng, các kỹ sư tiến hành thử nghiệm khả năng “bẫy” ruồi của robot trên một mặt bàn phẳng.

Trên mặt bàn được xem là mặt phẳng Oxy, robot “bẫy” ruồi được xác định bởi 3 đỉnh, đỉnh thứ i ($i=1..3$) có tọa độ (x_i, y_i) , có m con ruồi đậu trên mặt bàn tại vị trí có tọa độ (u_k, v_k) ($k=1..m$). Bạn hãy cho biết có bao nhiêu con ruồi chết trong robot “bẫy”.

Dữ liệu vào: từ tệp văn bản FLYTRAP.INP gồm:

- Dòng đầu tiên chứa 3 cặp số nguyên, mỗi cặp số là tọa độ đỉnh của tam giác robot “bẫy”, các số tách nhau bởi dấu cách; các đỉnh tam giác không trùng nhau.
- Dòng thứ 2 chứa số nguyên dương m ($1 \leq m \leq 10^3$), là số ruồi đậu trên mặt bàn.
- Từ dòng thứ 3 trở đi, mỗi dòng chứa 1 cặp số thực (u_k, v_k) ($-10^3 \leq u_k, v_k \leq 10^3$), là tọa độ đậu của 1 con ruồi trên mặt bàn, các số tách nhau bởi dấu cách.

Dữ liệu ra: tệp văn bản FLYTRAP.OUT, ghi một số nguyên là số ruồi bị chết.

Ví dụ :

FLYTRAP.INP	FLYTRAP.OUT
0 0 3 0 3 3 5 1.0 1.0 1.0 2.0 4.0 1.0 2.0 2.0 2.0 1.0	3
0 0 3 0 3 3 2 1.5 2.0 3.1 4.3	0

Bài 3: DÒ MẬT KHẨU (4.0 điểm) – Tên chương trình: PASSWORD.*

Việc bảo vệ máy tính của mình để hạn chế người khác xâm nhập vào là 1 vấn đề đặt ra cho mọi người sử dụng máy tính. Để tăng tính an toàn trong lưu trữ, bạn Bình đã quyết định giấu mật khẩu truy cập máy tính của mình vào 1 xâu T với 1 quy ước sao cho khi cần bạn ấy có thể lấy lại mật khẩu từ T như sau:

Mật khẩu là xâu P (gồm các kí tự chữ thường trong bảng chữ cái tiếng Anh). Xâu P là 1 chuỗi liên tiếp các kí tự trong T xuất hiện đúng 1 lần và dịch chuyển (mã hóa) nó qua m kí tự trong bảng mã ASCII, chẳng hạn trong bảng mã ASCII kí tự a được dịch chuyển qua 3 kí tự là d . Bạn hãy giúp Bình viết chương trình tìm lại mật khẩu.

Ví dụ: xâu $T = "aavwwnysccwcn"$. Với $m = 2$, mật khẩu P là **"xau"**

Dữ liệu vào: từ tệp văn bản **PASSWORD.INP** gồm:

- Dòng đầu tiên chứa số nguyên dương m ($0 \leq m \leq 25$), là số ký tự cần dịch chuyển.
- Dòng thứ hai chứa xâu T có chiều dài L ($1 \leq L \leq 10^6$), gồm các ký tự chữ thường $a...z$ trong bảng chữ cái tiếng Anh.

Dữ liệu ra: tệp văn bản **PASSWORD.OUT** chứa xâu P cần tìm. Nếu không tồn tại xâu P thì ghi 0.

Ví dụ:

PASSWORD.INP	PASSWORD.OUT
2 aavwwnysccwcn	xau
3 arabbbrbbccc	0

Bài 4: CÁP TREO (4.0 điểm) – Tên chương trình: CABCAR.*

Du lịch bằng cáp treo không chỉ rút ngắn thời gian di chuyển lên đỉnh núi "cưỡi mây" mà còn là dịp ngắm nhìn toàn cảnh thành phố từ trên cao. Các vị trí xuất phát của cáp treo được đặt dọc trên con đường thứ nhất ở ven biển, các vị trí đến sẽ được đặt dọc trên con đường thứ hai đi qua các điểm tham quan đẹp nhất trong thành phố. Hai con đường này không cắt nhau và cùng có hướng từ Đông sang Tây. 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.

Trong quá trình khảo sát xây dựng cáp treo, ban quản lý thiết kế phát hiện một số tuyến cắt nhau có nguy cơ xảy ra va chạm do đó cần phải loại bỏ các tuyến này và chỉ xây dựng các tuyến không cắt nhau để đảm bảo an toàn cho khách du lịch.

Yêu cầu: Tìm số tuyến nhiều nhất không cắt nhau.

Dữ liệu vào: từ tệp văn bản **CABCAR.INP** gồm hai dòng:

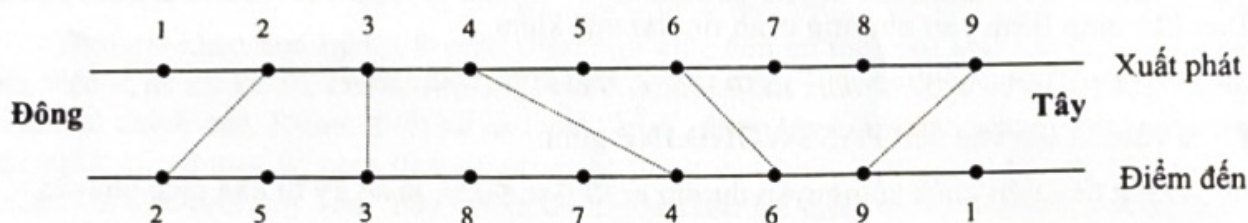
- Dòng thứ nhất số nguyên dương n ($1 \leq n \leq 10000$)
- Dòng thứ hai ghi n số nguyên p_1, p_2, \dots, p_n , là một hoán vị của số hiệu $1, 2, \dots, n$ ($1 \leq p_i \leq n; i = 1..n$), các số trên cùng một dòng ghi tách nhau một dấu cách.

Dữ liệu ra: tệp văn bản **CABCAR.OUT**, ghi một số nguyên là số tuyến nhiều nhất không cắt nhau.

Ví dụ:

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

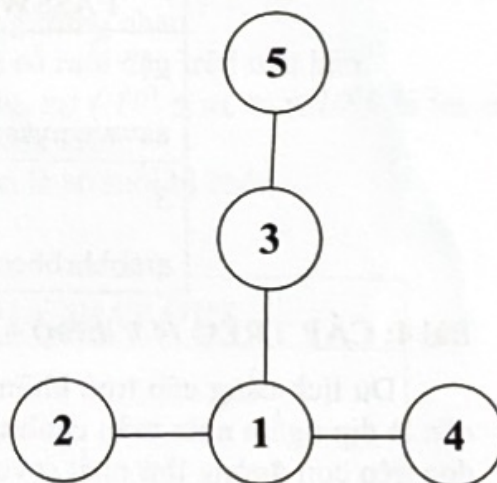
Hình vẽ dưới đây cho 1 ví dụ khi $n = 9$:



Bài 5: HƯỚNG DẪN ĐƯỜNG ĐI (4.0 điểm) – Tên chương trình: COVID.*

Để hỗ trợ TP.Hồ Chí Minh dập dịch trong đợt cao điểm dịch bệnh Covid19 vừa qua, một đoàn y bác sĩ từ miền Bắc đã được điều vào để hỗ trợ. Tuy nhiên, hệ thống đường đi ở TP.Hồ Chí Minh khá phức tạp. Vì thế, lãnh đạo TP.Hồ Chí Minh đã nhờ các lập trình viên tạo ra 1 hệ thống hướng dẫn đường đi tự động cho các y bác sĩ.

Để đơn giản hóa việc tìm đường bằng hệ thống dẫn đường tự động này, người ta đã ngăn bớt 1 số tuyến phố, chỉ đảm bảo sao cho giữa 2 khu phong tỏa chỉ có 1 đường đi duy nhất tới nhau. Tại mỗi điểm phong tỏa, sẽ đặt 1 máy hướng dẫn. Tại khu phong tỏa s , các y bác sĩ chỉ cần nhập vào số nguyên d là khu mình cần tới, máy sẽ hiển thị số nguyên t là khu trực tiếp nối với s và là khu tiếp theo mà các y bác sĩ phải di chuyển tới. Ví dụ với hình bên, tại khu số 5, nếu các y bác sĩ cần đến khu số 4 thì máy sẽ chỉ là cần di chuyển đến khu số 3.



Cho n là số khu phong tỏa và đường đi nối trực tiếp 2 khu là cặp số a_i, b_i , cho biết có đường đi nối trực tiếp hai khu a_i và b_i ($1 \leq a_i, b_i \leq n$; $a_i \neq b_i$; $i = 1..n$) và m truy vấn, mỗi truy vấn là một cặp số s và d , trong đó s là khu nơi y bác sĩ đang đứng, d nơi y bác sĩ muốn đến.

Yêu cầu: Hãy xác định số hiển thị trên màn hình ứng với mỗi truy vấn.

Dữ liệu vào: từ tệp văn bản COVID.INP gồm:

- Dòng đầu tiên chứa số nguyên n ($2 \leq n \leq 10^5$)
- $n-1$ dòng tiếp theo, mỗi dòng chứa 2 số nguyên a_i và b_i
- Dòng kế tiếp chứa số nguyên m ($1 \leq m \leq 10^5$)
- m dòng tiếp theo, mỗi dòng chứa 2 số nguyên s_j và d_j ($1 \leq s_j, d_j \leq n$, $s_j \neq d_j$).

Dữ liệu ra: tệp văn bản COVID.OUT ghi m số, mỗi số t nằm trên một dòng tương ứng với kết quả của các lần tra cứu.

5	3
1 2	4
1 3	1
1 4	
3 5	
3	
5 2	
1 4	
4 3	

Ràng buộc:

- Subtask #1: 30% số test của bài tương ứng $n \leq 50$.
- Subtask #2: 40% số test khác của bài tương ứng $n \leq 100$.
- Subtask #3: 30% số test còn lại có ràng buộc của đề bài.

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 :