

TỔNG QUAN ĐỀ THI

Bài	Tên bài	Tên file chương trình	Hạn chế thời gian	Hạn chế bộ nhớ
1	BẢNG XOẮN ỐC	SPIRAL.*	1 giây	256 M
2	SỐ NGUYÊN TỐ LỚN	MAXPRIME.*	1 giây	256 M
3	PYTHAGORE	PYTHAGORE.*	1 giây	256 M
4	ROBOT	ROBOT.*	1 giây	256 M

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

LẬP TRÌNH GIẢI CÁC BÀI TOÁN SAU:

Bài 1: BẢNG XOẮN ỐC

Cho bảng vuông kích thước $n \times n$. Người ta điền các số từ 1 đến n^2 vào bảng theo chiều xoắn ốc từ ngoài vào trong, theo chiều kim đồng hồ bắt đầu từ ô góc trái trên. Hình bên dưới minh họa cho bảng kích thước 4×4 và 5×5 tương ứng.

1	2	3	4
12	13	14	5
11	16	15	6
10	9	8	7

1	2	3	4	5
16	17	18	19	6
15	24	25	20	7
14	23	22	21	8
13	12	11	10	9

1 2 3 4 5 6
8 21 22 23 24 7
19 28 27 26 25 8
18 29 30 31 32 9
17 36 35 34 33 10
16 15 14 13 12 11

Yêu cầu: Hãy tính tổng các phần tử lớn nhất trên mỗi dòng của bảng.

Dữ liệu: Vào từ file văn bản SPIRAL.INP một số nguyên dương $n (1 \leq n \leq 10^6)$.

Kết quả: Ghi ra file văn bản SPIRAL.OUT một số nguyên dương là tổng các phần tử lớn nhất trên mỗi dòng của bảng.

Ví dụ:

SPIRAL.INP	SPIRAL.OUT
5	85

Bài 2: SỐ NGUYÊN TỐ LỚN

Số nguyên tố là số nguyên dương có đúng 2 ước số dương là 1 và chính nó. Số nguyên tố có rất nhiều ứng dụng trong thực tế. Trong phương pháp mã hóa, người ta sử dụng các số nguyên tố lớn để tạo ra khóa công khai. Số nguyên tố càng lớn thì khóa càng tin cậy, tính bảo mật của thông tin được mã hóa càng cao.

Số nguyên tố lớn nhất mà các nhà toán học tìm được hiện nay có khoảng 23 triệu chữ số. An cũng khá hứng thú với việc tìm ra số nguyên tố lớn. Tuy nhiên do khả năng có hạn nên cậu ta chỉ tìm được số nguyên tố có k ($2 \leq k \leq 12$) chữ số.

An hào hứng viết số nguyên tố vừa tìm được lên bảng và định khoe với lớp. Tuy nhiên do bất cẩn, An lỡ tay xóa mất đi đúng 2 chữ số.

Yêu cầu: Cho chuỗi độ dài k gồm $k - 2$ ký tự số và 2 ký tự '?' biểu diễn số nguyên tố của An trong đó các ký tự '?' là các chữ số bị xóa. Hãy xác định số nguyên tố lớn nhất có thể có của An. Biết rằng trong số nguyên tố của An không chứa chữ số 0.

Dữ liệu: Vào từ file văn bản MAXPRIME.INP

- Gồm một dòng chứa chuỗi độ dài không quá 12 ký tự, trong đó có đúng 2 ký tự '?', các ký tự còn lại là chữ số.
- Dữ liệu đảm bảo luôn có lời giải.

Kết quả: Ghi ra file văn bản MAXPRIME.OUT số nguyên tố tìm được.

Ví dụ:

MAXPRIME.INP	MAXPRIME.OUT
??9	929

Bài 3: PYTHAGORE

Định lý Pythagore do Pythagoras, nhà toán học người Hy Lạp sống vào khoảng thế kỉ thứ 6 trước công nguyên, phát minh. Ngày nay, định lý Pythagore được ứng dụng rộng rãi trong nhiều lĩnh vực khác nhau như kiến trúc, xây dựng và đo lường. Những ứng dụng phổ biến của định lý Pythagore có thể kể ra là tính khoảng cách giữa 2 điểm trong mặt phẳng hoặc kiểm tra một tam giác có phải là tam giác vuông.

Định lý có thể được phát biểu như sau: Cho tam giác vuông có độ dài 3 cạnh a, b, c với c là độ dài cạnh huyền. Khi đó $a^2 + b^2 = c^2$.

Bình rất thích thú với tính chất này nhưng cậu ta chỉ quan tâm đến các tam giác vuông có độ dài ba cạnh a, b, c ($0 < a < b < c$) là số nguyên. Bình nhận thấy rằng, với cùng một chu vi thì có thể tồn tại nhiều tam giác vuông khác nhau có độ dài ba cạnh đều là số nguyên. Chẳng hạn với chu vi 60, tồn tại 2 tam giác vuông là (10,24,26) và (15,20,25). Còn với chu vi 12 chỉ tồn tại 1 tam giác vuông (3,4,5).

Yêu cầu: Cho trước chu vi P . Đếm số lượng tam giác vuông khác nhau có độ dài ba cạnh a, b, c ($a < b < c$) đều là số nguyên và chu vi là P .

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

- Dòng đầu tiên chứa số nguyên t ($1 \leq t \leq 5000$) – số bộ test.
- Dòng thứ i trong t dòng tiếp theo chứa số nguyên P_i ($1 \leq P_i \leq 5000$) – chu vi của các tam giác vuông thứ i .

Kết quả: Ghi ra file văn bản **PYTHAGORE.OUT** gồm t dòng, dòng thứ i ghi một số nguyên là số lượng tam giác vuông khác nhau có độ dài ba cạnh đều là số nguyên và chu vi P_i .

Ví dụ:

PYTHAGORE . INP	PYTHAGORE . OUT
2	1
12	2
60	

Bài 4: ROBOT

Robot sơn đường được sử dụng để thay thế con người làm công việc kẻ vạch phân cách giao thông trên các đại lộ. Ta có thể xem đại lộ như là một trục số. Có n robot thực hiện công việc, robot thứ i sơn đoạn đường từ điểm x_i đến điểm y_i trên trục số ($1 \leq x_i \leq y_i \leq 10^9$) và mỗi đơn vị độ dài tốn đúng 1 lít sơn. Như vậy để hoàn thành công việc, robot thứ i được nạp $y_i - x_i$ lít sơn.

Người ta lên kế hoạch hoạt động để n robot không bị va chạm nhau trong quá trình làm việc. Và sau khi hoàn thành công việc, tất cả các điểm trong đoạn từ L tới R đều được sơn. Tuy nhiên, do có một công việc đột xuất nên người ta cần điều gắp đúng một robot đi thực hiện công việc tại nơi khác.

Yêu cầu: Cho thông tin của n robot và hai số L, R . Hãy tìm robot cần điều đi sao cho $n - 1$ robot còn lại vẫn sơn được mọi điểm trên đoạn từ L tới R đồng thời lượng sơn trong robot điều đi là lớn nhất có thể.

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

- Dòng đầu tiên chứa 3 số nguyên n, L, R ($1 \leq n \leq 10^5; 1 \leq L < R \leq 10^9$).
- Dòng thứ i trong n dòng tiếp theo chứa cặp số nguyên x_i, y_i - mô tả công việc của robot thứ i . Dữ liệu đảm bảo rằng luôn tồn tại một cách chọn ra một robot để $n - 1$ robot còn lại vẫn sơn được đoạn từ L tới R .

Kết quả: Ghi ra file văn bản **ROBOT.OUT** một số nguyên là lượng sơn của robot được điều đi.

Ví dụ:

ROBOT . INP			ROBOT . OUT	
4	1	10	5	
1	5	4		
3	8	5		
5	8	3		
8	12	4		

.....Hết.....

3 8
5 8
8 12

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:.....