

ĐỀ CHÍNH THỨC

(Đề thi gồm 02 trang)

TỔNG QUAN BÀI THI

Tên bài	Tên chương trình	Tập tin dữ liệu	Tập tin kết quả
SỐ TRANG	SOTRANG.*	SOTRANG.INP	SOTRANG.OUT
THAY THẾ	THAYTHE.*	THAYTHE.INP	THAYTHE.OUT
TÌM ĐƯỜNG	TIMDUONG.*	TIMDUONG.INP	TIMDUONG.OUT

Dấu * được thay thế bởi PAS hoặc CPP của ngôn ngữ lập trình được sử dụng tương ứng là Pascal hoặc C++. Các tập tin chương trình lưu trong cùng một thư mục với tên thư mục là TIN<số báo danh>. Ví dụ: thí sinh có số báo danh là 01234 thì tên thư mục là TIN01234.

Hãy lập trình giải 3 bài toán sau:

Bài 1: SỐ TRANG (6 điểm)

An buộc phải ngồi đọc cho xong quyển sách mà sư phụ cho mượn mấy hôm trước. Đọc được một lúc, An cảm thấy buồn chán và nghĩ cách làm sao cho đỡ buồn.

Chợt An nghĩ ra một cách: An bắt đầu đếm số chữ số dùng để đánh số trang trong toàn bộ quyển sách. Cuối cùng, An đếm được N chữ số.

An còn phát hiện ra một điều đặc biệt là ở ba trang đầu tiên của quyển sách không hề có ghi số trang gì cả mà chỉ có ghi ở trang thứ tư trở đi (trang 4 được ghi số 4, trang 5 được ghi số 5, trang 6 được ghi số 6, vân vân...)

Sau khi trả lại quyển sách cho sư phụ, An không nhớ quyển sách có bao nhiêu trang mà chỉ còn nhớ được số N.

Yêu cầu: Cho trước số N, hãy lập trình giúp An tìm ra số trang của quyển sách đó.

Dữ liệu vào: Từ tập tin văn bản **SOTRANG.INP**, gồm một số nguyên N ($1 \leq N \leq 10000$) cho biết số chữ số dùng để đánh số trang cho quyển sách.

Kết quả: Ra tập tin văn bản **SOTRANG.OUT**, gồm một số nguyên P cho biết tổng số trang của quyển sách.

Luôn đảm bảo An đã đếm đúng số chữ số, tức là luôn tồn tại lời giải.

Ví dụ:

SOTRANG.INP	SOTRANG.OUT	SOTRANG.INP	SOTRANG.OUT
1	4	2	5

SOTRANG.INP	SOTRANG.OUT	SOTRANG.INP	SOTRANG.OUT
3	6	10	11

Bài 2: THAY THẾ (7 điểm)

Phép toán mod được áp dụng trên số nguyên và cho kết quả là số dư của phép chia số nguyên. Ví dụ: $24 \bmod 16$ cho kết quả 8, còn $15 \bmod 16$ cho kết quả 15.

Cho trước bốn số nguyên dương N, P, A, B. Lập lại thao tác sau đây nhiều lần: lấy số N và thay thế nó bằng một trong ba số nguyên: $((N+A) \bmod P)$ hoặc $((N+B) \bmod P)$ hoặc $((N+A+B) \bmod P)$.

Yêu cầu: Hãy cho biết cần thực hiện ít nhất bao nhiêu thao tác thay thế ở trên để từ số N ta được số R.

Dữ liệu vào: Từ tập tin văn bản **THAYTHE.INP**, gồm:

- Dòng đầu tiên ghi hai số N, P.
 - Dòng thứ hai ghi ba số A, B, R.
- ($0 < N \leq 10^9$; $0 < A, B \leq P \leq 10^6$; $0 \leq R < P$; $N \neq R$)

Kết quả: Ra tập tin văn bản **THAYTHE.OUT**, gồm một số nguyên cho biết số thao tác ít nhất, nếu không thể thực hiện được thì ghi -1.

Ví dụ:

THAYTHE.INP	THAYTHE.OUT	Giải thích
20 16 3 4 15	2	Với $N=20$, thực hiện thay thế N như sau: • Lần 1: thay $N=8$ vì $(20+4) \bmod 16 = 8$ • Lần 2: thay $N=15$ vì $(8+3+4) \bmod 16 = 15$ Sau 2 lần thay thế, kết quả được $N=R=15$

THAYTHE.INP	THAYTHE.OUT
6 8 2 4 1	-1

Bài 3: TÌM ĐƯỜNG (7 điểm)

Bạn An đang đứng ở vị trí có tọa độ (1,1) trên bản đồ và muốn đi đến thành phố ByteCity ở tọa độ (N,N).

Đây là bản đồ hình vuông. Bản đồ có chứa các độ cao $A[i,j]$ tại mỗi tọa độ (i,j). Bạn An chỉ đi được từ một ô sang các ô kề cạnh.

Yêu cầu: Bạn An không giỏi leo lên (hoặc xuống) đồi nên muốn nhờ bạn lập trình tìm con đường sao cho độ lệch lớn nhất của độ cao hai ô kề cạnh là nhỏ nhất có thể được. Độ lệch của độ cao hai ô được hiểu là trị tuyệt đối của hiệu độ cao của hai ô này.

Dữ liệu vào: Từ tập tin văn bản **TIMDUONG.INP**, gồm:

- Dòng đầu tiên ghi số nguyên dương N ($1 < N \leq 500$).
- N dòng tiếp theo, mỗi dòng ghi N số nguyên dương. $A[i,j]$ cho biết độ cao của các vị trí tương ứng tại dòng i và cột j trên bản đồ ($0 \leq A[i,j] \leq 10^6$).

Kết quả: Ra tập tin văn bản **TIMDUONG.OUT**, gồm một số nguyên dương duy nhất cho biết độ lệch lớn nhất của độ cao hai ô kề cạnh trên con đường tìm được.

Ví dụ:

TIMDUONG.INP	TIMDUONG.OUT	Giải thích
4 3 5 7 7 2 4 4 7 3 3 5 4 9 5 8 5	1	Con đường cho kết quả tối ưu: (1,1)-(2,1)-(3,1)-(3,2)-(2,2)-(2,3)-(3,3)-(3,4)-(4,4) Trên con đường này, độ lệch lớn nhất của độ cao hai ô kề cạnh là 1.