



Ngôn ngữ lập trình C++

BÀI TẬP LẬP TRÌNH NÂNG CAO



1. Tìm số nguyên – FindNum.Cpp

Cho bốn số nguyên a, b, c, d thỏa mãn:

- $1 \leq a \leq b \leq 100;$
- $1 \leq c \leq d \leq 200;$

Tìm tất cả các số nguyên x thỏa mãn:

- $a \leq x \leq b;$
- $c \leq 2x \leq d;$

Dữ liệu cho trong file FindNum.Inp gồm:

- Dòng 1 ghi hai số nguyên a, b .
- Dòng 2 ghi hai số nguyên c, d .

Kết quả ghi ra file FindNum.Out gồm tất cả các số x tìm được, mỗi số ghi trên 1 dòng theo thứ tự từ giá trị nhỏ đến giá trị lớn. Nếu không có số nguyên x thỏa mãn thì ghi “No solution”.

Ví dụ:

FindNum.Inp	FindNum.Out
4 6	4
2 10	5
2 4 100 300	No solution



2. Kí tự thuộc chuỗi CharInString.Cpp

Cho hai chuỗi X và Y . Hãy kiểm tra xem, kí tự ‘ a ’ có thuộc chuỗi X hay không? Kí tự ‘ a ’ có thuộc chuỗi Y hay không?

Dữ liệu cho trong file CharInString.Inp gồm:

- Dòng 1 ghi chuỗi kí tự X gồm các chữ cái latin thường, độ dài không quá 100.
- Dòng 2 ghi chuỗi kí tự Y gồm các chữ cái latin thường, độ dài không quá 100.

Kết quả ghi ra file CharInString.Out gồm:

- Dòng 1 ghi “Yes” nếu kí tự ‘ a ’ thuộc chuỗi X , ngược lại ghi “No”.
- Dòng 2 ghi “Yes” nếu kí tự ‘ a ’ thuộc chuỗi Y , ngược lại ghi “No”.

Ví dụ:

CharInString.Inp	CharInString.Out
babb	Yes
bbbx	No



3. Tìm chữ số (1) – FindDigits1.Cpp

Cho số tự nhiên $N = \overline{2021x}$ và số nguyên dương K ($1 \leq K \leq 10$). Hãy tìm chữ số x để N chia hết cho K .

Dữ liệu cho trong file FindDigits1.Inp gồm một số nguyên dương K .

Kết quả ghi ra file FindDigits1.Out gồm tất cả các chữ số x tìm được theo thứ tự tăng dần, mỗi số được ghi trên một dòng. Nếu không có giá trị x thỏa mãn thì ghi “No solution”.

Ví dụ:

FindDigits1.Inp	FindDigits1.Out	Giải thích
2	0	$x = 0 \rightarrow N = 20210 : 2$
	2	$x = 2 \rightarrow N = 20212 : 2$
	4	$x = 4 \rightarrow N = 20214 : 2$
	6	$x = 6 \rightarrow N = 20216 : 2$
	8	$x = 8 \rightarrow N = 20218 : 2$



4. Tìm chữ số (2) – FindDigits2.Cpp

Cho số tự nhiên $N = \overline{2021x2021y}$ và số nguyên dương K ($1 \leq K \leq 1000$). Hãy tìm số cặp chữ số (x, y) để N chia hết cho K .

Dữ liệu cho trong file FindDigits2.Inp gồm một số nguyên dương K .

Kết quả ghi ra file FindDigits2.Out là số cặp chữ số (x, y) để N chia hết cho K .

Ví dụ:

FindDigits2.Inp	FindDigits2.Out	Giải thích
10	10	Ta có 10 cặp chữ số (x, y) . $(0,0) \rightarrow N = 2021020210 : 10$ $(1,0) \rightarrow N = 2021120210 : 10$ $(2,0) \rightarrow N = 2021220210 : 10$ $(3,0) \rightarrow N = 2021320210 : 10$ $(4,0) \rightarrow N = 2021420210 : 10$ $(5,0) \rightarrow N = 2021520210 : 10$ $(6,0) \rightarrow N = 2021620210 : 10$ $(7,0) \rightarrow N = 2021720210 : 10$ $(8,0) \rightarrow N = 2021820210 : 10$ $(9,0) \rightarrow N = 2021920210 : 10$



5. Cắt khúc gỗ thành 3 đoạn – Cut3.Cpp

Bác Vinh có 2 khúc gỗ có độ dài tương ứng là a và b . Đối với mỗi khúc gỗ, bác muốn cắt thành 3 đoạn có độ dài nguyên dương sao cho:

- Đoạn thứ 1 có độ dài x ;
- Đoạn thứ 2 có độ dài $x + 1$;
- Đoạn thứ 3 có độ dài $x + 2$.

Phần gỗ thừa bác Vinh cho mấy đứa cháu nhỏ của mình làm đồ chơi. Gọi X là tổng độ dài của các đoạn gỗ thừa này.

Yêu cầu: Tính xem, X có giá trị nhỏ nhất bằng bao nhiêu.

Dữ liệu cho trong file Cut3.Inp gồm hai số nguyên dương a và b ($10 \leq a, b \leq 10^9$).

Kết quả ghi ra file Cut3.Out là giá trị nhỏ nhất của X .

Ví dụ:

Cut3.Inp	Cut3.Out	Giải thích
20 30	2	Đoạn 1: Cắt thành 3 đoạn có độ dài: 5; 6; 7. Thừa 2. Đoạn 2: Cắt thành 3 đoạn có độ dài: 9, 10, 11. Thừa 0.



6. Kiểm tra biến đổi tăng giảm – CheckInc.Inp

Cho 2 số nguyên dương X, Y và 2 số nguyên dương a, b . Ta cần biến đổi giá trị của X về giá trị Y bằng cách sau: Mỗi lần biến đổi, ta thực hiện

- Chọn một trong hai số a và b .
- Cộng số được chọn vào X , tức là nếu chọn số a thì $X \leftarrow X + a$; nếu chọn số b thì $X \leftarrow X + b$;

Yêu cầu: Tìm số lần biến đổi ít nhất để đưa X về giá trị Y .

Dữ liệu cho trong file CheckInc.Inp gồm:

- Dòng đầu ghi số nguyên dương N ($N \leq 50$) là số bộ dữ liệu:
- N dòng sau, mỗi dòng ghi 4 số nguyên dương X, Y và a, b ($X, Y, a, b \leq 10^6$).

Kết quả ghi ra file CheckInc.Out gồm N dòng. Mỗi dòng là kết quả của một bộ dữ liệu tương ứng. Nếu có thể biến đổi để X thành Y thì ghi số lần biến đổi ít nhất, nếu không biến đổi được thì ghi “No”.

Ví dụ:

CheckInc.Inp	CheckInc.Out	Giải thích
2	2	Bộ dữ liệu 1: Thực hiện
2 6 1 2	No	2 lần, mỗi lần chọn giá
2 7 2 4		trị b .
		Bộ dữ liệu 2: Không thể
		thực hiện biến đổi để X
		bằng Y .