



Ngôn ngữ lập trình C++

BÀI TẬP LẬP TRÌNH NÂNG CAO



1. Biến đổi dãy số bằng phép cộng – ReformSum.Cpp

Cho dãy số nguyên A_1, A_2, \dots, A_N . Bạn có thể xóa đi một số hạng hoặc không xóa số hạng nào. Sau đó thực hiện biến đổi.

Chọn hai số bất kì trong dãy \rightarrow Xóa hai số được chọn \rightarrow Viết thêm vào dãy một số hạng bằng tổng hai số vừa xóa.

Quá trình biến đổi sẽ dừng lại khi dãy còn đúng một số hạng. Số hạng cuối cùng này ta đặt là X . Chú ý là, cứ sau mỗi lần biến đổi, số số hạng sẽ giảm xuống 1 số hạng.

Yêu cầu: Tìm giá trị lớn nhất của X có thể nhận được.

Dữ liệu cho trong file ReformSum.Inp gồm:

- Dòng đầu ghi số nguyên dương N .
- Dòng thứ 2 ghi N số nguyên A_1, A_2, \dots, A_N .

Kết quả ghi ra file ReformSum.Out là giá trị lớn nhất của X có thể nhận được.

Ví dụ:

ReformSum.Inp	ReformSum.Out	Giải thích
4 -10 -9 2 3	-4	Xóa số -10 khỏi dãy, ta được dãy mới: -9, 2, 3. Thực hiện biến đổi: Chọn 2 số hạng có giá trị: 2, 3, được dãy: -9, 5. Chọn 2 số hạng có giá trị: -9, 5, được dãy: -4.

Giới hạn:

- $3 \leq N \leq 10^6$;
- $|A_i| \leq 10^9$.



2. Biến đổi dãy số bằng phép nhân – ReformProduct.Cpp

Cho dãy số nguyên A_1, A_2, \dots, A_N . Bạn có thể xóa đi một số hạng hoặc không xóa số hạng nào. Sau đó thực hiện biến đổi.

Chọn hai số bất kì trong dãy \rightarrow Xóa hai số được chọn \rightarrow Viết thêm vào dãy một số hạng bằng tích hai số vừa xóa.

Quá trình biến đổi sẽ dừng lại khi dãy còn đúng một số hạng. Số hạng cuối cùng này ta đặt là X . Chú ý là, cứ sau mỗi lần biến đổi, số số hạng sẽ giảm xuống 1 số hạng.

Yêu cầu: Tìm giá trị lớn nhất của X có thể nhận được.

Dữ liệu cho trong file ReformProduct.Inp gồm:

- Dòng đầu ghi số nguyên dương N .
- Dòng thứ 2 ghi N số nguyên A_1, A_2, \dots, A_N .

Kết quả ghi ra file ReformProduct.Out là giá trị $X \% (10^9 + 7)$, với X là giá trị lớn nhất có thể nhận được.

Ví dụ:

ReformProduct.Inp	ReformProduct.Out	Giải thích
4 -10 -9 2 3	540	Thực hiện biến đổi: Chọn 2 số hạng có giá trị: 2, 3, được dãy: -10, -9, 6. Chọn 2 số hạng có giá trị: -10, -9, được dãy: 90, 6. Chọn 2 số hạng có giá trị: 90, 6, được dãy 540.

Giới hạn:

- $3 \leq N \leq 10^6$;
- $|A_i| \leq 10^9$.

**3☀. Số nhỏ nhất tránh dãy – MinOut.Cpp**

Cho dãy số nguyên không âm A_1, A_2, \dots, A_N . Với mỗi chỉ số i ta gọi P_i là số nguyên thỏa mãn:

- $P_i \geq 0$;
- P_i khác A_1, A_2, \dots, A_i ;
- P_i đạt giá trị nhỏ nhất.

Yêu cầu: Tính P_1, P_2, \dots, P_N .

Dữ liệu cho trong file MinOut.Inp gồm:

- Dòng đầu ghi số nguyên dương N .
- Dòng 2 ghi N số nguyên không âm A_1, A_2, \dots, A_N .

Kết quả ghi ra file MinOut.Out gồm N dòng, dòng thứ i ghi số nguyên P_i tìm được.

Ví dụ:

MinOut.Inp	MinOut.Out
5	0
1 0 3 2 6	2
	2
	4
	4

Giới hạn:

- Sub 1: $N \leq 1000$; $A_i \leq 10^6$;
- Sub 2: $N \leq 10^6$; $A_i \leq 10^6$;

**4☀. Bộ số nhân 2 kì diệu – Magic2.Cpp**

Xét bộ gồm 4 số nguyên (a, b, c, k) với $1 \leq k \leq 10$, $|a|, |b|, |c| \leq 10^6$. Ta thực hiện biến đổi:

Mỗi lần thực hiện:

- Chọn một số trong ba số a, b, c .
- Nhân đôi số được chọn, tức là $u \leftarrow u \times 2$ với u là số được chọn.

Ta nói bộ 4 số nguyên (a, b, c, k) là một bộ số nhân 2 kì diệu nếu ta có thể thực hiện biến đổi không quá k lần thì có $a < b < c$.

Yêu cầu: Cho n ($1 \leq n \leq 100$) bộ gồm 4 số nguyên (a_i, b_i, c_i, k_i) với $1 \leq k_i \leq 10$, $|a_i|, |b_i|, |c_i| \leq 10^6$. Ứng với mỗi bộ số, kiểm tra xem có phải bộ số nhân 2 kì diệu không?

Dữ liệu cho trong file Magic2.Inp gồm:

- Dòng đầu ghi số nguyên dương n .
- Dòng thứ i trong n dòng tiếp theo, mỗi dòng ghi 4 số nguyên a_i, b_i, c_i, k_i .

Kết quả ghi ra file Magic2.Out gồm n dòng.

- Dòng thứ i ghi “Yes” nếu bộ 4 số nguyên (a_i, b_i, c_i, k_i) là bộ số nhân 2 kì diệu. Ngược lại ghi “No”.



Ví dụ:

Magic2.Inp	Magic2.Out
3	Yes
1 2 3 1	No
2 2 4 1	Yes
2 2 4 2	



5. Đoạn nguyên tố - SegPrime.Cpp

Cho số nguyên dương N ($2 \leq N \leq 10^7$). Tìm đoạn $[a; b]$ sao cho:

- a, b là các số nguyên tố;
- $a \leq N \leq b$;
- $b - a$ đạt giá trị nhỏ nhất.

Dữ liệu cho trong file SegPrime.Inp gồm số nguyên dương N .

Kết quả ghi ra file SegPrime.Out là 2 số a, b tìm được.

Ví dụ:

SegPrime.Inp	SegPrime.Out
9	7 11



6. Tập đoạn không giao nhau – NoIntersect.Cpp

Cho N đoạn nguyên $[a_i, b_i]$, $0 \leq a_i < b_i \leq 10^7$, với $i = 1, 2, 3, \dots, N$. Ta nói hai đoạn $[a_i, b_i]$ và $[a_j, b_j]$ là giao nhau nếu có số nguyên x thỏa mãn: $a_i \leq x \leq b_i$ và $a_j \leq x \leq b_j$; ngược lại ta nói hai đoạn $[a_i, b_i]$ và $[a_j, b_j]$ không giao nhau. Ví dụ, hai đoạn $[1, 5]$ và $[3, 5]$ là giao nhau; hai đoạn $[1, 5]$ và $[13, 15]$ là không giao nhau.

Yêu cầu: Hãy tìm cách chọn được nhiều đoạn nhất sao cho 2 đoạn bất kì trong tập các đoạn được chọn là không giao nhau.

Dữ liệu cho trong file NoIntersect.Inp gồm:

- Dòng thứ nhất ghi số nguyên dương N ($N \leq 10^5$) là số các đoạn.
- N dòng tiếp theo, dòng thứ i ghi hai số nguyên a_i, b_i .

Kết quả ghi ra file NoIntersect.Out là số đoạn nhiều nhất có thể chọn được.

Ví dụ:

NoIntersect.Inp	NoIntersect.Out	Giải thích
4	3	Ta có thể chọn được nhiều nhất 3 đoạn: $[1, 3]$, $[4, 5]$, $[10, 14]$.
1 3		
2 4		
4 5		
10 14		