



Ngôn ngữ lập trình C++

BÀI TẬP LẬP TRÌNH NÂNG CAO



1. Dãy con liên tiếp có tổng chia hết cho K - Kcontinue.Cpp

Cho dãy số nguyên A_1, A_2, \dots, A_N . Tìm số cặp chỉ số (i, j) sao cho tổng $A_i + A_{i+1} + \dots + A_j$ chia hết cho K , $(1 \leq i \leq j \leq N)$.

Dữ liệu cho trong file **Kcontinue.Inp** gồm:

- Dòng đầu ghi hai số nguyên dương N và K .
- Dòng thứ 2 ghi N số nguyên A_1, A_2, \dots, A_N ($|A_i| \leq 10^6$).

Kết quả ghi ra file **Kcontinue.Out** là số cặp chỉ số (i, j) thỏa mãn đề bài.

Ví dụ:

Kcontinue.Inp	Kcontinue.Out	Giải thích
4 4 2 2 3 -4	2	Có 2 cặp chỉ số (1, 2) và (4, 4).

Giới hạn:

- $K \leq 10^6$;
- Sub1: $N \leq 1000$;
- Sub2: $N \leq 5 \times 10^5$;



2. Dãy con dư K – RemainK.Cpp

Cho dãy số nguyên dương A_1, A_2, \dots, A_N và số nguyên dương K . Dãy con gồm các số hạng ở vị trí liên tiếp A_i, A_{i+1}, \dots, A_j là một dãy con **RemainK** nếu thỏa mãn:

$$(A_i + A_{i+1} + \dots + A_j) \% K = (j - i + 1).$$

Yêu cầu: Tính số cặp (i, j) để A_i, A_{i+1}, \dots, A_j là một dãy con **RemainK**.

Dữ liệu cho trong file **RemainK.Inp** gồm:

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

Kết quả ghi ra file **RemainK.Out** là số cặp (i, j) để A_i, A_{i+1}, \dots, A_j là một dãy con **RemainK**.

Ví dụ:

RemainK.Inp	RemainK.Out	Giải thích
5 4 1 4 2 3 5	4	Ta có các dãy RemainK: [1] ứng với cặp chỉ số (1, 1). [4, 2] ứng với cặp chỉ số (2, 3). [1, 4, 2] ứng với cặp chỉ số (1, 3). [5] ứng với cặp (5, 5).

Giới hạn:

- $1 \leq N \leq 2 \times 10^5$;
- $1 \leq K \leq 10^6$;
- $1 \leq A_i \leq 10^9$.

**3☀ Chia hết 2019**

Cho chuỗi kí tự S gồm các kí tự chữ số thuộc tập $\{'1', \dots, '9'\}$. Gọi N là độ dài của chuỗi S . Các kí tự trong chuỗi S được đánh chỉ số từ 0 đến $N - 1$. Ta kí hiệu $S[i..j]$ là chuỗi gồm các kí tự từ chỉ số i đến chỉ số j .

Yêu cầu: Tính xem có bao nhiêu cặp chỉ số (i, j) , $0 \leq i \leq j \leq N - 1$ để số ứng với chuỗi $S[i..j]$ tạo thành số chia hết cho 2019.

Dữ liệu cho trong file **Divisor2019.Inp** gồm một chuỗi kí tự S .

Kết quả ghi ra file **Divisor2019.Out** là số cặp chỉ số tìm được.

Ví dụ:

Divisor2019.Inp	Divisor2019.Out
1817181712114	3

Giới hạn:

- Có 20% số test ứng với $N \leq 18$.
- Có 40% số test ứng với $N \leq 1000$.
- Có 40% số test ứng với $N \leq 200000$.

**4☀ Biến đổi đối xứng**

Dãy số $A[1], A[2], \dots, A[n]$ được gọi là dãy đối xứng nếu $A[i] = A[n-i+1]$ với $i = 1, 2, \dots, n$.

Cho dãy $A[1], A[2], \dots, A[n]$. Ta thực hiện các phép biến đổi trên dãy như sau:

Chọn hai số hạng kề nhau, thay thế hai số hạng đó bằng tổng của chúng, ta được một dãy mới, ít hơn dãy trước khi biến đổi một số hạng.

Yêu cầu: Cần phải biến đổi dãy ít nhất bao nhiêu lần để biến đổi dãy $A[1], A[2], \dots, A[n]$ thành dãy đối xứng.

Dữ liệu cho trong file **MPALIN.INP** như sau:

- Dòng đầu ghi số nguyên dương n ;
- Dòng sau ghi n số nguyên dương $A[1], A[2], \dots, A[n]$. ($A[i] \leq 10^9$).

Kết quả ghi ra file **MPALIN.OUT** là số lần biến đổi ít nhất để được dãy đối xứng.

Ví dụ:

MPALIN.INP	MPALIN.OUT	Giải thích
3 1 2 3	1	1, 2, 3 \rightarrow 3, 3
5 1 2 4 6 1	1	1, 2, 4, 6, 1 \rightarrow 1, 6, 6, 1
4 1 4 3 2	2	1, 4, 3, 2 \rightarrow 5, 3, 2 \rightarrow 5, 5

Giới hạn:

- 30% số test $n \leq 10$;
- 60% số test $n \leq 1000$;
- 10% số test $n \leq 1000000$.

**5 Smart Ants**

Có hai đàn kiến cùng đi qua một cái cầu hẹp và đi ngược chiều nhau. Đàn kiến thứ nhất đi từ bên trái sang bên phải cầu, đàn kiến còn lại đi từ bên phải sang bên trái của cầu. Do cái cầu hẹp nên khi đi đến đối diện nhau, hai đàn kiến đưa ra một cách đi thông minh đó là: Nếu hai con kiến ở hai đàn khác nhau và đối diện nhau thì nhảy qua để đổi chỗ cho nhau, thời gian để hai con kiến đổi chỗ cho nhau là 1 giây. Hai đàn kiến cứ làm như vậy cho đến khi cả hai đàn kiến cùng đi qua cầu.

Yêu cầu: Hãy đưa ra thứ tự của hai đàn kiến sau T giây, thời điểm bắt đầu tính thời gian là lúc con kiến đầu tiên của dãy thứ nhất gặp con kiến đầu tiên của dãy thứ 2.

Các con kiến được đánh dấu bằng các chữ cái La tinh in hoa, không có hai con kiến nào (trong cả hai đàn) được đánh dấu bởi một chữ cái in hoa.

Ví dụ:

Đàn kiến thứ nhất gồm: CBA (đi từ trái sang phải)

Đàn kiến thứ hai gồm: DFTY (đi từ phải sang trái)

Tại thời điểm gặp nhau, hai đàn kiến là:

ABCDFTY

Sau 1 giây, thứ tự của hai đàn kiến là: ABDCFTY

Sau 2 giây, thứ tự của hai đàn kiến là: ADBFCTY

Sau 3 giây, thứ tự của hai đàn kiến là: DAFBTCY

.....

Dữ liệu cho trong file **ANTS.INP** như sau:

- Dòng đầu ghi hai số nguyên dương n_1 và n_2 tương ứng là số kiến của đàn thứ nhất và đàn thứ hai.
- Dòng thứ 2 gồm xâu kí tự (chỉ gồm các kí tự in hoa) mô tả đàn kiến thứ nhất, kiến ở vị trí 1 đi đầu.
- Dòng thứ 3 gồm xâu kí tự mô tả đàn kiến thứ 2, kiến ở vị trí 1 đi đầu.
- Dòng thứ 4 ghi số nguyên T .

Kết quả ghi ra file **ANTS.OUT** gồm một xâu mô tả trạng thái của hai đàn kiến sau T giây (chú ý rằng, khi hai đàn kiến đã đi qua nhau, tức là đàn kiến 1 sang bên phải, đàn kiến 2 sang bên trái mà vẫn chưa hết T giây thì ghi ra trạng thái cuối này).

Ví dụ:

ANTS.INP	ANTS.OUT	ANTS.INP	ANTS.OUT	ANTS.INP	ANTS.OUT
3 3 ABC DEF 0	CBADEF	3 3 ABC DEF 2	CDBEAF	2 3 AV BCE 10	BCEVA