



Ngôn ngữ lập trình C++

BÀI TẬP ÔN THI HSG TỈNH 2021



1. CROSS WORD

Cho bảng kí tự gồm N dòng và M cột. Ban đầu, tất cả các ô vuông của bảng có ghi một kí tự là dấu chấm '.'. Trò chơi CROSS WORD được mô tả như sau:

Có hai chuỗi kí tự X và Y . Chuỗi X gồm M kí tự chữ cái in hoa, chuỗi Y gồm N kí tự chữ cái in hoa. Người chơi có nhiệm vụ đặt chuỗi X vào hàng ngang của bảng, chuỗi Y vào cột của bảng sao cho:

- Các kí tự từ trái sang phải (trên dòng đặt chuỗi X) tạo thành chuỗi X .
- Các kí tự từ trên xuống dưới (trên cột đặt chuỗi Y) tạo thành chuỗi Y .
- Chuỗi X và Y giao nhau tại một kí tự (kí tự này thuộc cả hai chuỗi), kí tự này phải xuất hiện đầu tiên trong chuỗi X và chuỗi Y (ưu tiên xuất hiện đầu tiên trong X trước).

Yêu cầu: Bạn là người chơi trò chơi CROSS WORD, hãy tìm vị trí đặt chuỗi X và Y theo đúng yêu cầu của trò chơi.

Dữ liệu cho trong file CROSSWORD.INP gồm một dòng duy nhất chứa hai chuỗi X và Y . Hai chuỗi cách nhau bởi kí tự trống, gồm các kí tự chữ cái in hoa, có ít nhất một kí tự chung. Độ dài của chuỗi X và Y không quá 30.

Kết quả ghi ra file CROSSWORD.OUT gồm bảng sau khi đặt chuỗi X và Y vào đúng quy định của trò chơi.

Ví dụ:

CROSSWORD.INP	CROSSWORD.OUT
BANANA PIDZAMA	.P.... .I.... .D.... .Z.... BANANA .M.... .A....
MAMA TATA	.T.. MAMA .T.. .A..
REPUBLIKA HRVATSKA	H..... REPUBLIKA V..... A..... T..... S..... K..... A.....

**2☀. Số xấu xí**

Một số được gọi là số xấu xí nếu số đó là 1 hoặc khi phân tích thành các thừa số nguyên tố, chỉ chứa các thừa số thuộc $\{2, 3, 5\}$. Ví dụ: 1, 2, 3, 4, 5, 6, 8, 9, 10, 12,...

Yêu cầu: Cho số nguyên dương n , tìm số xấu xí thứ n (trong dãy các số xấu xí được sắp xếp tăng dần).

Dữ liệu cho trong file UGLYNUM.INP gồm số nguyên dương n ($n \leq 500$).

Kết quả ghi ra file UGLYNUM.OUT là số xấu xí thứ n tìm được.

Ví dụ:

UGLYNUM.INP	UGLYNUM.OUT
1	1
9	10
2	2

**3☀. Số dư**

Cho n số nguyên dương a_1, a_2, \dots, a_n không đồng thời bằng nhau. Tìm số nguyên dương d lớn nhất sao cho $a_1 \% d = a_2 \% d = \dots = a_n \% d$.

Dữ liệu cho trong file remain.inp gồm:

- Dòng đầu ghi số nguyên dương n ($n \leq 100$).
- Dòng sau ghi n số nguyên dương a_1, a_2, \dots, a_n .

Kết quả ghi ra file remain.out là số nguyên dương d tìm được.

Ví dụ:

remain.inp	remain.out
3	2
3 7 9	

Giới hạn:

- Sub1: $a_i \leq 10^6$;
- Sub2: $a_i \leq 10^{16}$.

**4☀. Sắp xếp đặc biệt – SpecSort.Cpp**

Cho ba dãy số nguyên A_1, A_2, \dots, A_N ; B_1, B_2, \dots, B_N ; C_1, C_2, \dots, C_N . Độ đặc biệt của ba dãy là số các chỉ số i ($1 \leq i \leq N$) thỏa mãn: $A_i < B_i < C_i$.

Yêu cầu: Hãy sắp xếp các dãy A_1, A_2, \dots, A_N ; B_1, B_2, \dots, B_N ; C_1, C_2, \dots, C_N sao cho độ đặc biệt của ba dãy sau khi sắp xếp là lớn nhất.

Dữ liệu cho trong file SpecSort.Inp gồm:

- Dòng 1 ghi số nguyên dương N .
- Dòng 2 ghi N số nguyên A_1, A_2, \dots, A_N ;
- Dòng 3 ghi N số nguyên B_1, B_2, \dots, B_N ;
- Dòng 4 ghi N số nguyên C_1, C_2, \dots, C_N .



Kết quả ghi ra SpecSort.Out là độ đặc biệt lớn nhất của 3 dãy số có thể đạt được khi sắp xếp lại vị trí các số hạng trong từng dãy.

Ví dụ:

SpecSort.Inp	SpecSort.Out	Giải thích
5 9 6 14 1 8 2 10 3 12 11 15 13 5 7 4	3	Có thể sắp xếp lại 3 dãy như sau: 1, 6, 8, 9, 14 3, 2, 10, 12, 11 4, 7, 15, 13, 5

Giới hạn:

- $1 \leq N \leq 10^5$;
- $1 \leq A_i, B_i, C_i \leq 10^9$.



5☀. Ước bé thứ hai

Ta nhận thấy, với mọi số tự nhiên $x > 1$ đều có ít nhất là hai ước dương vì luôn có ước là 1 và x . Ta gọi ước nhỏ thứ 2 của x là *second*(x). Chú ý là ước dương nhỏ nhất của x luôn bằng 1.

Yêu cầu: Cho N số nguyên dương A_1, A_2, \dots, A_N ($A_i > 1$), hãy tính tổng $\text{second}(A_1) + \text{second}(A_2) + \dots + \text{second}(A_N)$.

Dữ liệu cho trong file SecondDivisor.Inp gồm:

- Dòng đầu ghi số nguyên dương N là số các số hạng của dãy.
- Dòng thứ 2 ghi N số nguyên dương A_1, A_2, \dots, A_N .

Kết quả ghi ra file SecondDivisor.Out là tổng $\text{second}(A_1) + \text{second}(A_2) + \dots + \text{second}(A_N)$.

Ví dụ:

SecondDivisor.Inp	SecondDivisor.Out	Giải thích
3 2 3 4	7	Ước nhỏ thứ 2 của 2 là 2 Ước nhỏ thứ 2 của 3 là 3 Ước nhỏ thứ 2 của 4 là 2 Tổng bằng $2 + 3 + 2 = 7$.

Giới hạn:

- 50% số test ứng với $N \leq 2000$; $2 \leq A_i \leq 10^9$;
- 50% số test còn lại ứng với $N \leq 500000$; $2 \leq A_i \leq 10^7$;



6☀. Dãy số

Ta gọi S_k là dãy các kí tự khi viết liên tiếp k số tự nhiên $1, 2, \dots, k$.

Ví dụ: $S_1 = 1, S_2 = 12, S_3 = 123, S_{12} = 123456789101112$.

Yêu cầu: Cho số tự nhiên i , tìm chữ số thứ i trong dãy vô hạn $S_1S_2S_3\dots$

Dữ liệu cho trong file SEQDIGIT.INP gồm:

- Dòng đầu ghi số nguyên dương t ($1 \leq t \leq 100$) là số testcase:
- t dòng tiếp theo, mỗi dòng ghi một số nguyên dương i ($1 \leq i \leq 2 \cdot 10^9$).

Kết quả ghi ra file SEQDIGIT.OUT gồm t dòng, mỗi dòng là chữ số thứ i trong dãy vô hạn $S_1S_2S_3\dots$

Ví dụ:

SEQDIGIT.INP	SEQDIGIT.OUT
2	2
8	1
2	

Giải thích: Dãy $S_1S_2S_3S_4S_5S_6\dots$ 112123123412345123456