



# OLYMPIC TIN HỌC SINH VIÊN LẦN THỨ XXIX, 2020

## Khởi thi: Cá nhân Không Chuyên & Cao đẳng

Thời gian làm bài: 180 phút

Ngày thi: 09/12/2020

Nơi thi: Đại học Bách khoa Cần Thơ

### TỔNG QUAN ĐỀ THI

STT	Tên bài	File nguồn nộp	File dữ liệu	File kết quả	Điểm
1	Chăn bò	cows.*	cows.inp	cows.out	100
2	Phân số	fraction.*	fraction.inp	fraction.out	100
3	Dự trữ nước	buildcol.*	buildcol.inp	buildcol.out	100
4	Cây xanh	trees.*	trees.inp	trees.out	50
5	Trò chơi xếp số	game11.*	game11.inp	game11.out	50

Chú ý: Dấu \* được thay thế bởi đuôi ngầm định của ngôn ngữ được sử dụng.

Hãy lập trình giải các bài toán dưới đây:

### Bài 1. CHĂN BÒ (100 điểm)

Phú Ông có một đàn bò gồm  $n$  con, các con bò mang số hiệu từ 1 đến  $n$ . Bờm được Phú Ông giao nhiệm vụ, hàng ngày, thả bò ra và chiều tối lừa bò về chuồng. Một hôm, do mải chơi nên Bờm đã để lạc mất 1 con bò. Bờm rất lo lắng và muốn xác định số hiệu của con bò bị lạc để đăng tin tìm kiếm vì nếu không sẽ bị Phú Ông phạt nặng. Đàn bò khá đông và Bờm chỉ nhớ, Bờm đã tính tổng số hiệu của các con bò còn lại là một số nguyên  $S$ .

**Yêu cầu:** Cho 2 số nguyên dương  $n$  và  $S$ . Hãy giúp Bờm tìm ra số hiệu của con bò bị lạc.

**Dữ liệu:** Vào từ file văn bản **cows.inp** gồm 2 số nguyên  $n, S$ . Dữ liệu đảm bảo bài toán có nghiệm.

**Kết quả:** Ghi ra file văn bản **cows.out** một số nguyên là số hiệu của con bò đi lạc.

**Ví dụ:**

cows.inp	cows.out
5 12	3

**Giới hạn:**

Subtask 1 (50% số điểm):  $n \leq 10^4$ ;

Subtask 2 (50% số điểm):  $n \leq 10^9$ .

### Bài 2. PHÂN SỐ (100 điểm)

Khi tìm hiểu về *Số hữu tỉ*, Bờm biết rằng, số hữu tỉ có thể được biểu diễn dưới dạng phân số  $X/Y$ , trong đó  $X$  và  $Y (Y \neq 0)$  là các số nguyên. Số hữu tỉ cũng được phân thành 2 loại: thập phân hữu hạn hoặc vô hạn tuần hoàn. Ví dụ, số hữu tỉ  $116/495 = 0,2(34)$  là số thập phân vô hạn tuần hoàn, còn số  $3/20 = 0,15$  là số thập phân hữu hạn. Với số hữu tỉ  $X/Y$ , Bờm muốn xác định số hữu tỉ là loại thập phân hữu hạn hay vô hạn tuần hoàn.

**Yêu cầu:** Cho  $n$  số nguyên dương  $a_1, a_2, \dots, a_n$  và  $n$  số nguyên dương  $b_1, b_2, \dots, b_n$ . Gọi  $X = a_1 \times a_2 \times \dots \times a_n$  và  $Y = b_1 \times b_2 \times \dots \times b_n$ . Hãy giúp Bờm xác định phân số  $X/Y$  là thập phân hữu hạn hay vô hạn tuần hoàn.

**Dữ liệu:** Vào từ file văn bản **fraction.inp**

- Dòng đầu tiên chứa số nguyên dương  $T$  ( $1 \leq T \leq 10$ ) là số lượng bộ dữ liệu.
- $T$  dòng sau, mỗi dòng có khuôn dạng: Số đầu tiên mỗi dòng là  $n$ , tiếp theo là dãy gồm  $2 \times n$  số nguyên dương  $a_1, a_2, \dots, a_n, b_1, b_2, \dots, b_n$  mô tả một bộ dữ liệu. Các số trên cùng một dòng cách nhau bởi dấu cách.

**Kết quả:** Ghi ra file văn bản **fraction.out** gồm  $T$  dòng, mỗi dòng ghi "finite" hoặc "repeating" tương ứng với câu trả lời phân số  $X/Y$  là thập phân hữu hạn hay vô hạn tuần hoàn.

**Ví dụ:**

<b>fraction.inp</b>	<b>fraction.out</b>	<b>Giải thích</b>
4	repeating	$7/30 = 0,2(3)$
2 1 7 5 6	finite	$3/20 = 0,15$
1 3 20	finite	$4/1 = 4$
1 4 1	repeating	$116/495 = 0,2(34)$
1 116 495		

**Giới hạn:** Gọi  $V = \max(a_1, a_2, \dots, a_n, b_1, b_2, \dots, b_n)$ .

Subtask 1 (25% số điểm):  $n = 1, V \leq 10^6$ ;

Subtask 2 (25% số điểm):  $n \leq 10^4, V \leq 10^6$ ;

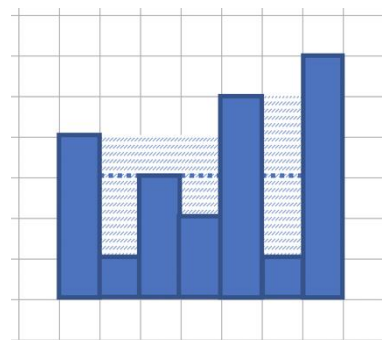
Subtask 3 (25% số điểm):  $n \leq 10^5, V \leq 10^5$ ;

Subtask 4 (25% số điểm):  $n \leq 10^5, V \leq 10^6$ .

### Bài 3. Dự trữ nước (100 điểm)

Ở quê Bờm có một con mương cung cấp nước cho đồng ruộng. Để điều tiết dòng chảy, người dân đặt  $n$  cột bê tông xếp cạnh nhau có cùng độ rộng với con mương và có chiều cao lần lượt là  $a_1, a_2, \dots, a_n$ .

Khi trời mưa, nước sẽ đọng lại ở các cột có độ cao thấp hơn. Giả sử có 7 cột với độ cao lần lượt là 4, 1, 3, 2, 5, 1, 6 thì lượng nước mưa đọng lại là 10 đơn vị. Để mặt đáy được bằng phẳng, mọi người muốn nâng các cột bê tông lên. Cụ thể, cần chọn một độ cao  $X$  lớn nhất và nâng các cột có độ cao thấp hơn  $X$  lên bằng  $X$  mà vẫn đảm bảo lượng nước mưa được giữ lại ít nhất là  $M$  đơn vị.



**Yêu cầu:** Cho độ cao của  $n$  cột bê tông và  $M$  đơn vị nước cần giữ lại. Hãy tìm độ cao  $X$  lớn nhất thỏa mãn yêu cầu. Giả sử lượng nước khi trời mưa là đủ để ngập các vũng nước và nước không bị thất thoát.

**Dữ liệu:** Vào từ file văn bản **buildcol.inp**

- Dòng đầu tiên gồm hai số nguyên dương  $n$  và  $M$  ( $M \leq 10^{18}$ ).
- Dòng thứ hai gồm  $n$  số nguyên dương  $a_1, a_2, \dots, a_n$ .

**Kết quả:** Ghi ra file văn bản **buildcol.out** số nguyên  $X$  thỏa mãn yêu cầu. Trường hợp không có phương án thì ghi số  $-1$ .

Ví dụ:

buildcol.inp	buildcol.out
7 4 4 1 3 2 5 1 6	3

**Giới hạn:**

Subtask 1 (40% số điểm):  $n \leq 500$ ;  $a_i \leq 1000$ ;

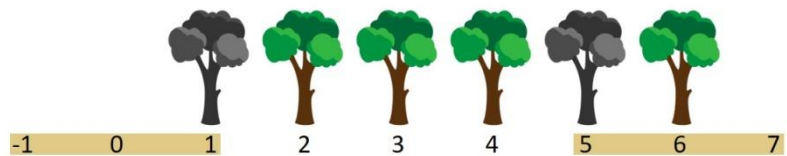
Subtask 2 (30% số điểm):  $n \leq 5000$ ;  $a_i \leq 5000$ ;

Subtask 3 (30% số điểm):  $n \leq 10^5$ ;  $a_i \leq 2 \times 10^9$ .

#### Bài 4. Cây xanh (50 điểm)

Trên một đoạn đường dẫn vào làng Bờm (có thể xem đoạn đường như một trục số), dân làng trồng  $n$  cây xanh tại các tọa độ  $1, 2, \dots, n$  (mỗi tọa độ trồng 1 cây), các vị trí khác ngoài  $n$  tọa độ này thì không được trồng cây (xem hình minh họa). Thời tiết khắc nghiệt kéo dài đã làm hư hại một số cây trồng. Bờm được giao nhiệm vụ điều khiển một robot để xử lý các cây hư hại. Robot hoạt động theo nguyên tắc: robot di chuyển theo một hướng từ đầu đến cuối con đường và xử lý các cây trồng tại  $k$  tọa độ liên tiếp. Mỗi lần cần xử lý các cây nào đó, Bờm cung cấp tọa độ  $x$ , robot sẽ xử lý các cây trồng tại các tọa độ  $x, x+1, \dots, x+k-1$ . Tọa độ  $x$  là một vị trí bất kỳ trên đường và không nhất thiết phải thuộc đoạn  $[1, n]$ . Khoảng cách giữa 2 tọa độ liên tiếp cung cấp điều khiển robot không được nhỏ hơn  $k$ .

Những cây không hư hại nếu bị xử lý có thể ảnh hưởng không tốt cho cây. Do đó, Bờm cần phải chỉ định các tọa độ thích hợp cho robot để số cây không hư hại mà robot không xử lý là nhiều nhất.



Chẳng hạn với 6 cây trồng, các cây tại tọa độ 1 và 5 bị hư hại. Với  $k=3$ , nếu chỉ định tọa độ làm việc cho robot lần lượt là  $-1$  và  $5$  thì số cây không hư hại mà robot không xử lý là 3.

**Yêu cầu:** Cho danh sách các cây trồng bị hư hại. Hãy xác định số lượng cây không bị hư hại mà robot không xử lý là nhiều nhất.

**Dữ liệu:** Vào từ file văn bản **trees.inp**

- Dòng đầu tiên chứa 3 số nguyên  $n, k, m$  ( $0 < k \leq n$ );
- Dòng tiếp theo chứa  $m$  số nguyên  $a_1, a_2, \dots, a_m$  là tọa độ các cây bị hư hại.

**Kết quả:** Ghi ra file văn bản **trees.out** số lượng nhiều nhất các cây không hư hại và không bị xử lý.

Ví dụ:

trees.inp	trees.out
6 3 2 1 5	3

**Giới hạn:**

Subtask 1 (40% số điểm):  $n \leq 100$ ;  $m \leq 50$ ;

Subtask 2 (40% số điểm):  $n \leq 2 \times 10^5$ ;  $m \leq 10^5$ ;

Subtask 3 (20% số điểm):  $n \leq 10^9$ ;  $m \leq 2 \times 10^5$ .

## Bài 5. Trò chơi xếp số (50 điểm)

Sau khi được học về dấu hiệu chia hết cho 11, Bờm lập trình một trò chơi như sau: Máy tính sinh một số nguyên dương  $n$ , sau đó chọn ra  $m$  số phân biệt trong các số  $1, 2, \dots, n$  và ghép lại theo một thứ tự nào đó. Nhiệm vụ của người chơi là sử dụng tất cả  $n - m$  số còn lại để ghép thành một số chia hết cho 11.

Để tính toán mức độ khó của trò chơi, Bờm muốn đếm số lượng hoán vị các số chưa dùng để ghép tiếp vào dãy mà máy tính đã ghép để tạo thành một số chia hết cho 11.

**Yêu cầu:** Cho số nguyên dương  $n$  và thứ tự của  $m$  số mà máy tính đã ghép. Hãy đếm số hoán vị các số còn lại để ghép tiếp vào dãy mà máy tính đã ghép để tạo thành một số chia hết cho 11.

**Dữ liệu:** Vào từ file văn bản **game11.inp**

- Dòng đầu chứa hai số nguyên  $n$  và  $m$  ( $0 \leq m < n$ );
- Dòng thứ hai gồm  $m$  số nguyên dương đôi một khác nhau có giá trị không vượt quá  $n$ .

**Kết quả:** Ghi ra file văn bản **game11.out** phần dư của  $r$  khi chia cho  $10^9 + 7$  với  $r$  là số lượng hoán vị các số còn lại để ghép tiếp vào dãy mà máy tính đã ghép để tạo thành một số chia hết cho 11.

**Ví dụ:**

game11.inp	game11.out
3 1 1	1

**Giới hạn:**

Subtask 1 (40% số điểm):  $n \leq 10$ ;

Subtask 2 (30% số điểm):  $n \leq 20$ ;

Subtask 3 (30% số điểm):  $n \leq 100$ .

----- **Hết** -----