

TỔNG QUAN VỀ ĐỀ THI

	File nguồn nộp	File dữ liệu	File kết quả	Thời gian mỗi test	Biểu điểm
Bài 1	NPAIRS.*	NPAIRS.INP	NPAIRS.OUT	1,0 giây	6 điểm
Bài 2	KSTR.*	KSTR.INP	KSTR.OUT	1,0 giây	7 điểm
Bài 3	ROBOT.*	ROBOT.INP	ROBOT.OUT	1,0 giây	7 điểm

Dấu * được thay thế bởi PAS hoặc CPP của ngôn ngữ lập trình được sử dụng tương ứng là Pascal hoặc C++.

Hãy lập trình giải các bài toán sau:

Bài 1: Cho 4 số nguyên dương A, B, C, D .

Hãy xác định số lượng cặp số thực khác nhau $(X; Y)$ sao cho

- $0 < X, Y < 1$
- $A * X + B * Y$ và $C * X + D * Y$ là số nguyên

Dữ liệu: Vào từ file văn bản **Npairs.inp** gồm một dòng duy nhất chứa 4 số nguyên dương A, B, C, D ($0 < A, B, C, D < 10^9$).

Các số trên một dòng của input file được ghi cách nhau bởi dấu cách.

Kết quả: Ghi ra file văn bản **Npairs.out** một số nguyên duy nhất là đáp án bài toán.

Ví dụ:

NPAIRS.INP	NPAIRS.OUT	Giải thích
1 2 10 12	6	$(0.25, 0.375), (0.5, 0.25), (0.75, 0.125), (0.25, 0.875), (0.5, 0.75), (0.75, 0.625)$

Bài 2: Cho số nguyên dương K và N tập hợp khác rỗng S_1, S_2, \dots, S_N . Tập S_i ($1 \leq i \leq N$) gồm các phần tử khác nhau $\in [0; 9]$

Người ta định nghĩa phép toán $S_i - S_j$ là những phần tử chỉ thuộc tập S_i và không thuộc tập S_j

Ví dụ: $S_i = (1, 3, 8)$ và $S_j = (2, 9, 3)$ khi đó $S_i - S_j = (1, 8)$

Dễ dàng nhận thấy phép toán trên không có tính kết hợp, tức là $(S_i - S_j) - S_p \neq S_i - (S_j - S_p)$ nên chúng ta quy ước thứ tự thực hiện phép toán $S_{i_1} - S_{i_2} - \dots - S_{i_m}$ là thực hiện từ phải qua trái.

Ví dụ: $(1, 2, 3) - (2, 3) - (1, 3) = (1, 2, 3) - (2) = (1, 3)$.

Yêu cầu: Hãy xác định số cách chọn các tập $S_{i_1}, S_{i_2}, \dots, S_{i_m}$ ($1 \leq i_1 < i_2 < \dots < i_m \leq N$) từ tập S_1, S_2, \dots, S_N sao cho $S_{i_1} - S_{i_2} - \dots - S_{i_m}$ được kết quả là tập có ít nhất K phần tử khác nhau.

Dữ liệu: Vào từ file văn bản **KSTR.INP**

- Dòng đầu tiên chứa hai số nguyên dương K ($1 \leq K \leq 8$) và N ($2 \leq N \leq 50\,000$)
- N dòng tiếp theo, dòng thứ $i + 1$ mô tả tập S_i chứa các số $t \ c_1 \ c_2 \dots c_t$ trong đó t là số lượng phần tử của tập S_i , $c_1 \ c_2 \dots c_t$ là các phần tử của tập S_i .

Các số trên một dòng của input file được ghi cách nhau bởi dấu cách.

Kết quả: Ghi ra file văn bản **KSTR.OUT** một số nguyên duy nhất là kết quả bài toán (lấy theo modulo **123457**)

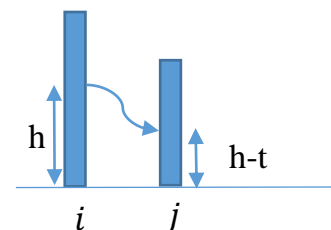
Ví dụ:

KSTR.INP	KSTR.OUT	Giải thích
3 3 5 5 6 7 8 9 3 4 5 6 3 7 8 9	6	Có thể chọn: $s_1, s_1 - s_2, s_1 - s_2 - s_3, s_2, s_2 - s_3, s_3$

Bài 3: ROBOT

HD vừa sáng tạo ra một trò chơi điều khiển robot mới cho 2 bé Bi, Bo chơi. Nội dung trò chơi như sau:

- Có N cây cột đánh số từ 1 đến N , cây cột thứ i có chiều cao $h[i](m)$
- Có M đường nhảy dạng i, j, t tương ứng là nhảy từ cây i sang cây j (hoặc từ cây j sang cây i) mất $t(s)$ và nếu nhảy từ độ cao h ($h \in \mathbb{N}, h \leq h[i]$) của cây i thì sang cây j sẽ có độ cao là $h - t$ với điều kiện $0 \leq h - t \leq h[j]$
- Nếu robot di chuyển lên xuống trên cột hiện tại, thời gian di chuyển mất $1(s)$ trên $1m$ di chuyển.



Hiện tại robot đang ở độ cao X của cây 1, Bi-Bo cần phải tìm phương án di chuyển nhanh nhất đến độ cao $h[N]$ của cây N . Bạn hãy giúp 2 bé Bi-Bo tính thời gian di chuyển ngắn nhất thỏa mãn yêu cầu đầu bài?

Dữ liệu: Vào từ file văn bản **ROBOT.INP**

- Dòng 1: Chứa 3 số nguyên dương N, M, X tương ứng là số lượng cây cột, số lượng đường nhảy và độ cao của robot đang ở cột 1. ($2 \leq N \leq 100.000; 1 \leq M \leq 300.000; 0 \leq X \leq h[1]$)
- N dòng tiếp theo, dòng thứ i chứa 1 số nguyên dương $h[i]$ tương ứng là chiều cao của cột i . ($1 \leq h[i] \leq 1.000.000.000 \forall i = 1..N$)
- M dòng tiếp theo, mỗi dòng chứa 3 số nguyên dương i, j, t tương ứng là nhảy từ cây i sang cây j (hoặc từ cây j sang cây i) mất $t(s)$ ($1 \leq t \leq 1.000.000.000$)

Các số trên một dòng của input file được ghi cách nhau bởi dấu cách.

Kết quả: Ghi ra file văn bản **ROBOT.OUT** một số duy nhất là thời gian ngắn nhất để robot di chuyển đến độ cao $h[N]$ của cây N , nếu không thể di chuyển đến thì ghi -1

Ví dụ:

ROBOT.INP	ROBOT.OUT	Giải thích
-----------	-----------	------------

5 5 0 50 100 25 30 10 1 2 10 2 5 50 2 4 20 4 3 1 5 4 20	110	Trèo lên 50(m) ở cây 1 mất 50(s) Nhảy từ cây 1 sang cây 2: - Mất 10(s) - ở độ cao 40 trên cây 2 Nhảy từ cây 2 sang cây 4: - Mất 20(s) - ở độ cao 20 trên cây 4 Nhảy từ cây 4 sang cây 5: - Mất 20(s) - ở độ cao 0 - trèo thêm 10(m) mất 10(s) Tổng thời gian: 110(s)
ROBOT.INP	ROBOT.OUT	Giải thích
2 1 0 1 1 1 2 100	-1	Từ cây 1, bất kỳ độ cao nào, khi nhảy sang cây 2 đều không thực hiện được vì $h - t < 0$
ROBOT.INP	ROBOT.OUT	Giải thích
4 3 30 50 10 20 50 1 2 10 2 3 10 3 4 10	100	Di chuyển xuống 10(m) ở cây 1 mất 10 (s) và đang ở độ cao 20(m) Nhảy sang cây 2: - Mất 10(s) - ở độ cao 10 trên cây 2. Nhảy sang cây 3: - Mất 10(s) - Ở độ cao 0(m), trèo lên 10(m) mất 10(s), ở độ cao 10(m); Nhảy sang cây 4: - Mất 10(s), - ở độ cao 0 (m), trèo lên 50(m) mất 50(s) Tổng thời gian: 100(s)

Chú ý:

- 25% số điểm tương ứng với các test có: $N \leq 1.000$; $M \leq 3.000$; $h[i] \leq 100 \forall i = 1..N$; $t \leq 100$
- 25% số điểm tương ứng với các test có $X = 0$

----- HẾT -----

(Thí sinh không sử dụng tài liệu, cán bộ coi thi không cần giải thích gì thêm)