

149. Phân tích

Program	SUMS.*
Input	SUMS.INP
Output	SUMS.OUT
Score	100

Cho n số nguyên dương a_1, a_2, \dots, a_n và số nguyên dương S . Hãy đếm xem có bao nhiêu cách phân tích S thành tổng các số nguyên dương trong n số ở trên. Mỗi số sử dụng không quá một lần

Input:

- Dòng đầu tiên ghi hai số nguyên dương n và S ($n \leq 40, S \leq 10^{18}$)
- Dòng thứ hai ghi n số nguyên dương a_1, a_2, \dots, a_n ($a_i \leq 10^{18}$)

Output: Một số nguyên duy nhất là số cách phân tích S thành tổng các số trong n số a_1, a_2, \dots, a_n mỗi số sử dụng không quá một lần

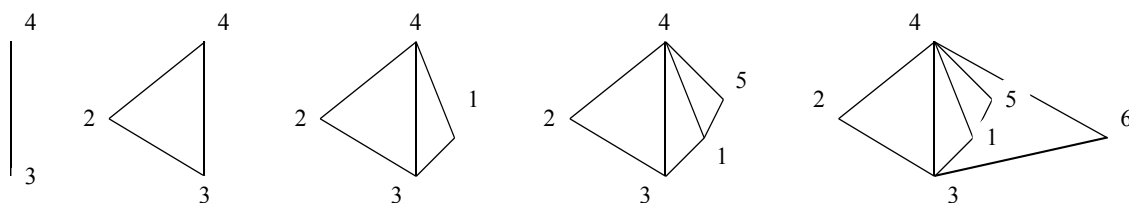
Example:

input	output
4 6 1 2 3 4	2

150. Hành tinh X

Program	XPLANET.*
Input	XPLANET.INP
Output	XPLANET.OUT
Score	100

Mọi người dân trong hành tinh X đều dựng nhà hình tam giác. Để tránh thủ thời gian, cách xây dựng của họ rất đặc biệt. Đầu tiên họ xây một bức tường thẳng, sau đó, họ chỉ việc xây thêm hai bức tường mới ghép với một bức tường đã có sẵn, kết quả được một ngôi nhà kín hình tam giác. Tất nhiên, hai bức tường mới được xây thêm vẫn có thể dùng để bắt đầu xây thêm các nhà mới. Có khi một số nhà nằm trong một số nhà khác (xem hình vẽ trong đó thể hiện quá trình xây một số ngôi nhà).



Để chiếu sáng các nhà, họ đặt ở mỗi góc nhà một bóng đèn, các nhà chung góc có thể dùng chung đèn đặt tại góc đó. Tại mỗi góc có một công tắc chuyển đổi trạng thái (tắt \rightarrow bật, bật \rightarrow tắt) không những bóng đèn ở góc đó mà còn các bóng đèn ở các góc cùng tường với góc đó.

Cho trước tình trạng của hệ các bóng đèn, hãy tìm một dãy bấm các công tắc sao cho mọi đèn tại mọi góc đều sáng và số lượng công tắc được bấm là ít nhất có thể được.

Input: trong đó dòng thứ nhất ghi số N là số góc nhà, $3 \leq N \leq 200$, các góc nhà đánh số từ 1 đến N . Tiếp theo là $2N-3$ dòng, mỗi dòng ghi hai số i, j với ý nghĩa có tường nối hai góc i và j . Dòng cuối cùng ghi N số mà số thứ i bằng 1 hoặc 0 tương ứng với đèn ở góc i sáng hoặc tối, dãy này thể hiện tình trạng cho trước của dãy các bóng đèn tại các góc. Dữ liệu phù hợp với mô tả bài toán.

Output: Dòng thứ nhất ghi số nguyên K mà nếu bài toán không có lời giải, $K=-1$, nếu có lời giải, K là số các công tắc cần bấm, tiếp theo là K dòng, mỗi dòng ghi một số hiệu của công tắc được bấm.

Example:

Input	Output
6	2
1 3	1
1 4	6
1 5	
2 3	
2 4	
3 4	
3 6	
4 5	
4 6	
0 1 1 1 0 0	

151. Khu vui chơi

Program LAND.*
 Input LAND.INP
 Output LAND.OUT
 Score 100

Một khu đất kích thước $m \times n$. Theo qui hoạch ban đầu trên khu đất đó sẽ xây dựng k khu vui chơi, khu vui chơi thứ i sẽ là một khu vực với kích thước $a_i \times b_i$. Tuy nhiên một vấn đề mới phát sinh, người ta muốn qui hoạch lại để có thêm một khu vui chơi có dạng hình vuông và hình vuông diện tích càng lớn càng tốt. Các khu vui chơi được xây dựng có thể tiếp xúc với nhau nhưng không được giao nhau.

Yêu cầu: Cho m, n và k khu vui chơi, hãy tính diện tích của khu đất hình vuông lớn nhất tìm được.

Input:

- Dòng đầu ghi ba số nguyên m, n, k ($m, n \leq 10, k \leq 5$)
- k dòng sau, mỗi dòng ghi hai số nguyên a_i, b_i

Output: Gồm một dòng ghi một số nguyên là diện tích của khu đất hình vuông lớn nhất tìm được.

Example:

input	output
5 5 2	16
1 5	
1 4	

152. Lát gạch

Program BRICK.*
 Input BRICK.INP
 Output BRICK.OUT
 Score 100

Cho một hình chữ nhật kích thước $2 \times n$ ($n \leq 10^9$). Hãy đếm số cách lát các viên gạch có kích thước 1×2 và 2×1 vào hình trên sao cho không có phần nào của các viên gạch thừa ra ngoài, cũng không có vùng diện tích nào của hình chữ nhật không được lát.

Input: Dòng đầu ghi T là số lượng test ($T \leq 100$). T dòng tiếp theo mỗi dòng ghi một số n .

Output: Ghi ra T dòng là kết quả các test tương ứng. Vì con số này rất lớn nên các bạn chỉ cần in ra phần dư khi chia cho 10^9+7

Example:

input	output
3	1
1	2
2	3
3	

153. Tính tổng

Program SUMK.*
 Input SUMK.INP
 Output SUMK.OUT
 Score 100

Viết chương trình tính tổng:

$$1^k + 2^k + \dots + n^k$$

Input: Dòng đầu tiên ghi T ($T \leq 5$) - số lượng bộ dữ liệu. T dòng tiếp theo, mỗi dòng ghi hai số nguyên dương n, k ($k \leq 50, n \leq 10^9$)

Example:

input	output
2	1
1 1	9
2 3	