

D – Árvore Binária

Árvore binária é uma estrutura de dados bastante comum em ciência da computação. Neste problema, vamos olhar para uma árvore binária infinita, onde os nós contêm um par de números inteiros. A árvore é construída assim:

- A raiz contém o par $(1, 1)$;
- Se um nó contém (a, b) , seu filho da esquerda contém $(a + b, b)$ e seu filho direito contém $(a, a + b)$.

Dado o conteúdo (a, b) de algum nó da árvore binária descrita acima, suponha que você está percorrendo a árvore, da raiz até o nó dado, ao longo do caminho mais curto possível. Você pode descobrir quantas vezes você tem que ir para um nó filho à esquerda e quantas vezes para um nó filho à direita?

Entrada

A primeira linha da entrada conterá um inteiro T , o número de casos de teste. Cada caso de teste é composto de uma única linha contendo dois inteiros i e j ($1 \leq i, j \leq 2 \cdot 10^9$) que representam um nó (i, j) . Você pode assumir que este é um nó válido na árvore binária descrita acima.

Saída

A saída para cada caso de teste começa com uma linha contendo “Cenário # i :”, onde i é o número do caso de teste, a partir de 1. Em seguida, imprima uma única linha contendo dois números inteiros e e d , separados por um único espaço. O inteiro e é quantas vezes você tem que ir para a esquerda e d é quantas vezes você tem que ir para a direita, ao percorrer a árvore da raiz até o nó dado na entrada. Termine a saída para um caso de teste com uma linha em branco.

Exemplo

Entrada:	Saída:
3	Cenário #1:
42 1	41 0
3 4	Cenário #2:
17 73	2 1
	Cenário #3:
	4 6