

**Задача L. ДЕЛИТЕЛИ**

Да се напише програма, която намира:

- броя на простите числа в интервала $[a, b]$;
- за всяко число от същия интервал, което не е просто – неговия най-малък прост делител.

Вход: Програмата трябва да може да обработва при едно изпълнение няколко тестови примера, които са зададени в **двоичен (binary) файл**, който ще бъде зададен като **стандартен вход**. Всеки тестов пример се състои от двете 32-битови положителни цели числа, които са краища на интервала – първо във файла е записан левият край, а после – десният.

Изход: За всеки тестов пример програмата трябва да изведе на **стандартния изход**, първо, ред с текста `test case #<t>`: (спазвайте стриктно форматирането от примера), където `<t>` е номерът на текущия тестов пример, а след това ред с текста `<p> primes in [<a>,]` ., където `<a>` и `` са двата края на интервала, а `<p>` е броят на простите числа в него. Всеки от следващите редове в изхода за тестовия пример е във вида `<q> <k>` и означава, че простото число `<q>` е най-малък прост делител на `<k>` непости числа в интервала. Двойките трябва да бъдат сортирани във възходящ ред на `<q>`. Между резултатите от всеки два поредни тестови примера трябва да има по един празен ред.

Ограничения: $2 \leq a \leq 10^7$, $2 \leq b \leq 10^7$.

Пример:

Вход	Изход
2 10 17 17 9 9 20 200	test case #1: 4 primes in [2, 10]. 2 4 3 1 test case #2: 1 primes in [17, 17]. test case #3: 0 primes in [9, 9]. 3 1 test case #4: 38 primes in [20, 200]. 2 91 3 30 5 12 7 6 11 3 13 1

Забележка: По понятни причини входните данни в примера са показани в текстов, а не в двоичен вид, както ще бъдат подадени на стандартния вход.



Task L. DIVISORS

Write a program to find:

- the number of primes in the interval $[a, b]$;
- the smallest prime divisor for each number from the same interval, which is not prime.

Input: The program must be able to process in one run several test cases that are set in **binary file** which will be given as **standard input**. Each test case consists of two 32-bit positive integers that are the ends of the interval – the left end is saved first in the file, and then – the right end.

Output: For each test case, the program must write to the **standard output**, first, a line with the text `test case #<t>`: (strictly follow the format of the example), where $\langle t \rangle$ is the number of the current test case; on the second line – the text `<p> primes in [<a>,].`, where $\langle a \rangle$ and $\langle b \rangle$ are the two ends of the interval, and $\langle p \rangle$ – the number of prime numbers in it. Each of the following lines in the output for the test case should be in the form `<q> <k>` and means that the prime number $\langle q \rangle$ is the smallest prime divisor of $\langle k \rangle$ not prime numbers in the interval. Couples must be sorted in ascending order of $\langle q \rangle$. Between each two consecutive test cases a blank line should be printed.

Restrictions: $2 \leq a \leq 10^7, 2 \leq b \leq 10^7$.

Example:

Input	Output
2 10 17 17 9 9 20 200	test case #1: 4 primes in [2, 10]. 2 4 3 1 test case #2: 1 primes in [17, 17]. test case #3: 0 primes in [9, 9]. 3 1 test case #4: 38 primes in [20, 200]. 2 91 3 30 5 12 7 6 11 3 13 1

Note: For obvious reasons, the input data in the example is shown in text form, not binary, as will be submitted on standard input.