



PONTIFICIA UNIVERSIDAD CATOLICA DE CHILE
ESCUELA DE INGENIERIA
DEPARTAMENTO DE CIENCIA DE LA COMPUTACION

Diseño y Análisis de Algoritmos - IIC2283

Tarea 3

Entrega: Miércoles 15 de diciembre (hasta las 11:59pm)

La solución de cada problema debe ser un programa en Python 3. Estos programas deben ser subidos a su repositorio de git creado para este propósito. Sus archivos deben llamarse `p1.py` y `p2.py` para las preguntas 1 y 2 respectivamente, y estar en la carpeta T3 de su repositorio. Cualquier pregunta de la tarea hágala en el foro del curso (<https://github.com/marceloarenassaavedra/IIC2283-2-21/issues>). Si quiere usar alguna librería en sus soluciones debe preguntar en el foro si esta librería está permitida. El foro es el canal de comunicación oficial para todas las tareas.

Problema 1

Después de su junta con amigos, Martín quedó con curiosidad sobre la parte técnica de las criptomonedas, se dio cuenta de que en criptografía siempre se menciona que se usan primos “grandes”, él no sabe como encontrar primos “grandes” y por eso te pidió ayuda. Martín te va a dar un intervalo $[a, b]$ y quiere un primo p tal que $a \leq p \leq b$. (Usen el algoritmo visto en clase¹ y busquen como optimizarlo para tener el tiempo esperado.)

Formato

El input corresponde a una sola línea que contiene a a y b separados por un espacio. El output es un primo p tal que $a \leq p \leq b$ o -1 si es que no hay un primo que cumpla las condiciones anteriores.

Límites

$$1 \leq a \leq b \leq 10^{100}$$

Tiempo de ejecución

2 segundos

Complejidad y error esperado

La complejidad esperada es $O((\log^3(b) + k \cdot \log(b)) \cdot \log(b))$, con un k tal que el error sea menor a 10^{-10} .

¹El test de primalidad de Solovay-Strassen, hay una implementación base en el repositorio del curso bajo la función `test_primalidad`.

Ejemplos

Input 1

1 100

Output 1

59

Input 2

1000000000 2000000000

Output 2

1000000007

Hint

Los primos son más comunes de lo que uno cree, y la standard library de Python² es sorprendentemente útil.

Problema 2

Dados dos enteros positivos n y k , ¿cuántos subconjuntos de enteros entre 1 y n de tamaño k existen tales que todos los k enteros del subconjunto son primos relativos de n ? ¿Cuál es el **dígito menos significativo distinto de 0** de dicha cantidad? Recordar que dos número enteros x e y son primos relativos si el máximo común divisor entre ellos es 1.

Formato

El input corresponde a una sola línea que contiene a n y k separados por un espacio. El output es el dígito menos significativo no cero de la cantidad preguntada, o bien -1 si es que no hay dígitos no cero.

Límites

$$1 \leq n \leq 10^6$$

$$1 \leq k \leq 10^6$$

Tiempo de ejecución

1 segundo

Complejidad esperada

$$O(n \log \log(n))$$

²<https://docs.python.org/3/library/>

Ejemplos

Input 1

7 3

Output 1

2

Input 2

29 5

Output 2

8

Input 3

65 13

Output 3

6

Input 4

1 2

Output 4

-1

Explicación Ejemplo

El conjunto de números coprimos con 7 entre 1 y 7 es $\{1, 2, 3, 4, 5, 6\}$, y todas las formas de escoger 3 elementos son $\{1, 2, 3\}$, $\{1, 2, 4\}$, $\{1, 2, 5\}$, $\{1, 2, 6\}$, $\{1, 3, 4\}$, $\{1, 3, 5\}$, $\{1, 3, 6\}$, $\{1, 4, 5\}$, $\{1, 4, 6\}$, $\{1, 5, 6\}$, $\{2, 3, 4\}$, $\{2, 3, 5\}$, $\{2, 3, 6\}$, $\{2, 4, 5\}$, $\{2, 4, 6\}$, $\{2, 5, 6\}$, $\{3, 4, 5\}$, $\{3, 4, 6\}$, $\{3, 5, 6\}$ y $\{4, 5, 6\}$, son 20 conjuntos distintos, y el dígito menos significativo no cero de 20 es 2.

Hint

La criba de Eratóstenes puede ser útil.