TEORÍA DE NÚMEROS

Practicas SAGE $\mathbf{n^o\ 1}$

Implementar en `SAGE` los siguientes algoritmos.

(1) Lista de primos:
INPUT: $n \in \mathbb{N}$.
OUPUT: $\mathbb{P}(n) = \{\text{primos } p \leq n\}$.

Crear una lista con los primos menores a 10^6.

(2) Conjetura de Goldbach:
INPUT: $n \in 2\mathbb{N}$, $n > 2$.
OUPUT: Dos primos p, q tales que $n = p + q$.

Hacer una tabla en la que en la primera columna aparezcan los números positivos pares menores que 1000 y en la segunda y tercera la descomposición obtenida de aplicar el algoritmo anterior.

(3) Conjetura de Goldbach II:
INPUT: $n \in \mathbb{N}$, $n > 17$.
OUPUT: Tres primos p_1, p_2, p_3 distintos tales que $n = p_1 + p_2 + p_3$.

Hacer una tabla en la que en la primera columna aparezcan los números positivos pares menores que 1000 y en la segunda, tercera y cuarta la descomposición obtenida de aplicar el algoritmo anterior.

(4) Teorema de los números primos:
INPUT: $n \in \mathbb{N}$.
OUPUT: $\pi(n) = \#\mathbb{P}(n)$.

Hacer una tabla en la que en la primera columna aparezcan distintos valores de n, en la segunda $\pi(n)$, en la tercera $n/\log n$ y en la cuarta el error cometido.

(5) Primos gemelos:
INPUT: $n \in \mathbb{N}$.
OUPUT 1: $\mathbb{P}_2(n) = \{\text{primos } p \leq n \text{ tal que } p + 2 \text{ es primo}\}$.
OUPUT 2: $\pi_2(n) = \#\mathbb{P}_2(n)$.

− Hacer una tabla de los primos gemelos menores que 1000.
− Hacer una tabla en la que en la primera columna aparezcan distintos valores de n, en la segunda $\pi_2(n)$ y en la tercera $n/(\log n)^2$. Si suponemos

$$\lim_{n\to\infty} \frac{\pi_2(n)}{n/(\log n)^2} = C \in \mathbb{R}\,,$$

calcular C con al menos 4 cifras decimales.

(6) Conjeturando:
INPUT: $n, k \in \mathbb{N}$.
OUPUT 1: $\mathbb{P}_{2k}(n) = \{\text{primos } p \leq n \text{ tal que } p + 2k \text{ es primo}\}$.
OUPUT 2: $\pi_{2k}(n) = \#\mathbb{P}_{2k}(n)$.

− Calcular $\mathbb{P}_4(10^6)$, $\mathbb{P}_6(10^6)$ y $\mathbb{P}_8(10^6)$.
− Hacer una tabla en la que en la primera columna aparezcan distintos valores de n, en la segunda $\pi_4(n)$, en la tercera $\pi_6(n)$ y en la cuarta $\pi_8(n)$.
− De manera análoga al ejercicio (5) buscar funciones que aproximen a $\pi_{2k}(n)$ para $k = 2, 3, 4$.
− Para $k \in \mathbb{N}$ ¿Qué se puede decir del conjunto $\mathbb{P}_{2k} = \mathbb{P}_{2k}(\infty)$?

(7) Teorema Dirichlet:
INPUT: $a, b, n \in \mathbb{N}$ tal que $(a, b) = 1$.
OUPUT 1: $\mathbb{P}_{a,b}(n) = \{m \leq n \text{ tal que } am + b \text{ es primo}\}$.
OUPUT 2: $\pi_{a,b}(n) = \#\mathbb{P}_{a,b}(n)$.

Tomar distintos valores de $a, b \in \mathbb{N}$ tal que $(a, b) = 1$ y hacer lo siguiente:
− Calcular $\mathbb{P}_{a,b}(1000)$.
− Hacer una tabla en la que en la primera columna aparezcan distintos valores de n, y en la segunda $\pi_{a,b}(n)$.
− ¿Qué se puede decir del conjunto $\mathbb{P}_{a,b} = \mathbb{P}_{a,b}(\infty)$?

(8) Una conjetura de Hardy-Littlewood:
INPUT: $n \in \mathbb{N}$ y $G \in \mathbb{Z}[x]$ irreducible.
OUPUT 1: $\mathbb{P}_G(n) = \{m \leq n \text{ tal que } G(m) \text{ es primo}\}$.
OUPUT 2: $\pi_G(n) = \#\mathbb{P}_G(n)$.

Tomar un polinomio $G \in \mathbb{Z}[x]$ irreducible cuadrático y hacer lo siguiente:
− Calcular $\mathbb{P}_G(1000)$.
− Hacer una tabla en la que en la primera columna aparezcan distintos valores de n, y en la segunda $\pi_G(n)$.
− Si suponemos

$$\lim_{n\to\infty} \frac{\pi_G(n)}{n/\log n} = C_G \in \mathbb{R}\,,$$

calcular C con al menos 4 cifras decimales.
− ¿Qué se puede decir del conjunto $\mathbb{P}_G = \mathbb{P}_G(\infty)$?

(9) Primos en progresiones aritméticas:
INPUT: $n \in \mathbb{N}$.
OUPUT: $p_1, \ldots, p_n$ primos en progresión aritmética.

Calcular progresiones aritméticas de primos de distintas longitudes.

(10) Teorema de los cuatro cuadrados de Lagrange:
INPUT: $n \in \mathbb{N}$.
OUPUT: $n_1, n_2, n_3, n_4 \in \mathbb{N}$ tal que $n = n_1^2 + n_2^2 + n_3^2 + n_4^2$.

Hacer una tabla en la que en la primera columna aparezcan los números positivos menores que 1000 y en la segunda la descomposición obtenida de aplicar el algoritmo anterior.

(11) Conjeturando:
INPUT: $n \in \mathbb{N}$.
OUPUT: `true` si existe un primo p tal que $n^2 < p < (n+1)^2$ y el primo p, `false` en otro caso.

Para distintos valores de $n \in \mathbb{N}$ hacer ejecutar el algoritmo anterior ¿Qué ocurre?.

(12) ¡¡¡Conjetura!!!!

Nota: En todos los algoritmos anteriores se tendrá en cuenta para la calificación la velocidad del algoritmo.