Numeros Primos

Dueñas Jiménez Cristian Alexis

November 6, 2022

1 Introducción

El programa debe de preguntar si quiere calcular otra "n" o no y salir hasta que se le especifique. La salida debe ser expresada en notación de conjunto, debe ir a un archivo de texto. En un archivo especificar el conjunto en binario y en otro el conjunto en decimal. Del archivo de salida, graficar el número símbolos y de 1s de cada cadena. El eje de las "x" es la cadena y el eje de las "y" los símbolos que tiene esa cadena. Específicamente en el reporte, calcular y graficar cuando $n = 200^3$. Al mismo tiempo, calcular adicionalmente sus logaritmo en base 10.

2 Marco Teórico

Los números primos son aquellos que solo son divisibles entre ellos mismos y el 1, es decir, que si intentamos dividirlos por cualquier otro número, el resultado no es entero. Dicho de otra forma, si haces la división por cualquier número que no sea 1 o él mismo, se obtiene un resto distinto de cero.

Los números binarios son números que están dentro del sistema binario de numeración que está constituido por dos cifras 1 y 0, un sistema en el cual se escriben cantidades, códigos, mensajes y otros lenguajes con tan solo dos elementos dentro de la numeración, haciendo que el código se simplifique la comprensión de los sistemas informáticos, pues hará que un elemento tenga un valor unitario o nulo. Es decir que se trabaja en un sistema de puertas cerradas o abiertas.

3 Desarrollo

Primero, se hace uso de listas para guardar datos que nos serán de utilidad para posteriormente graficar nuestros datos.

```
1 import numpy as np
2 from functions import *
3 import sys
4
5 def getPrimos(n):
6     primos = []
```

```
    unos = []
    largo =[]
    cont = -1

    f_primo1 = open("../PARCIAL 1/PROGRAMAS/Programa2/outputs/primosbin.txt", "w", encoding="utf-8")
    f_primo2 = open("../PARCIAL 1/PROGRAMAS/Programa2/outputs/primosint.txt", "w", encoding="utf-8")

    f_primo1.write("{  ")
    f_primo2.write("{  ")
    for i in range (2,n+1):
        if Primo(i):
            primos.append(i)
            binario = bin(i)[2:]
            f_primo1.write(','+binario)
            cont+=1
            f_primo2.write(','+str(i))
            unos.append(binario.count('1'))
            largo.append(cont)

    f_primo1.write("}")
    f_primo2.write("}")
    f_primo1.close()
    f_primo2.close()
    graficar(largo, unos, "Cadenas", "Cantidad de 1's")
    graficar(largo, np.log10(unos), "Cadenas", "Cantidad de 1's (Log10)")
    graficar(largo, np.log2(unos), "Cadenas", "Cantidad de 1's (Log2)")

#Generamos nuestro men  de opciones
opc = "1"
print("MENU DE OPCIONES")
while opc == "1":

    opc = input("Seleccione una opci n: \n1.- Ingresar n \n2.- Finalizar\n")

    if opc == "1":
        n = int(input("Ingrese el valor de n: "))
        getPrimos(n)
    else:
        if opc == "2":
            print(f"FIN DE LA EJECUCION")
            sys.exit(0)
        else:
            if opc != "1":
                print("Seleccione una opcion valida")
```

Algoritmo 1: Números primos hasta N

```
import matplotlib.pyplot as plt

def graficar(largo, unos, ejeX, ejeY):

    plt.plot(largo, unos, marker="")
    plt.xlabel(ejeX)
```

```
7       plt.ylabel(ejeY)
8       plt.title("GRAFICA")
9       plt.show()
10
11  def Primo(num):
12      bandera = False
13      if num > 1:
14          for j in range(2, int(num/2) + 1):
15              if (num % j) == 0:
16                  break
17          else:
18              bandera = True
19      return bandera
```

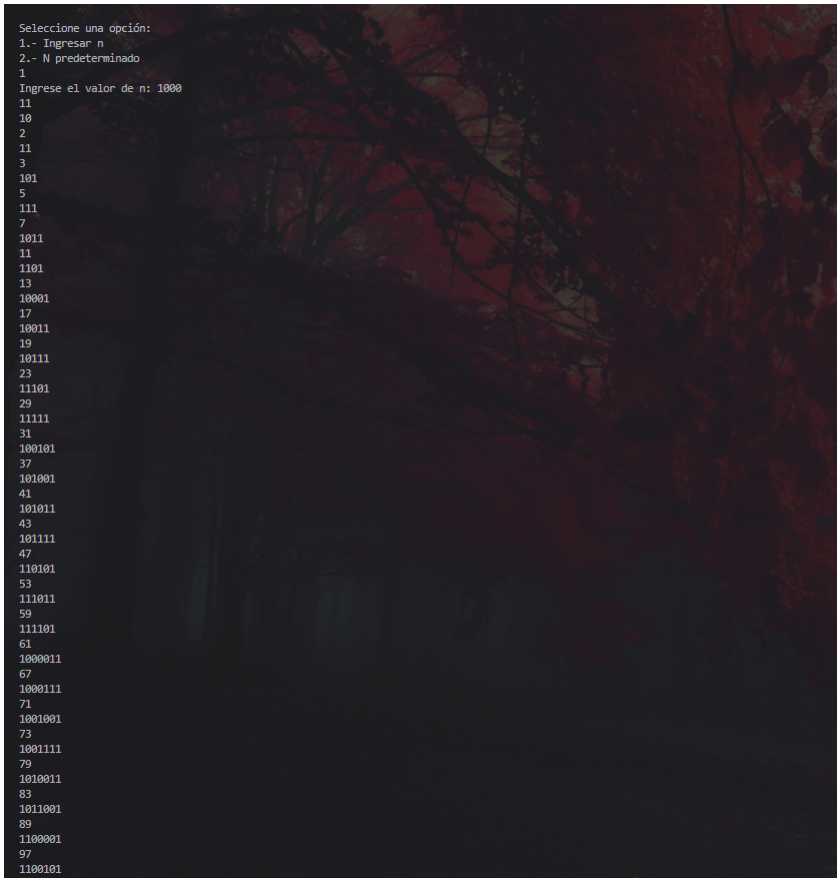
Algoritmo 2: Funciones para la clase main

Lo que hacemos es crear una función que determine cuando un número es Primo o no es primo. Si es el caso de que sí, retorna verdadero, y en caso de que no, retorna falso. Posteriormente utilizamos la función sigma, la cual recibe un valor "n" y este delimita de donde a donde se buscarán números primos.

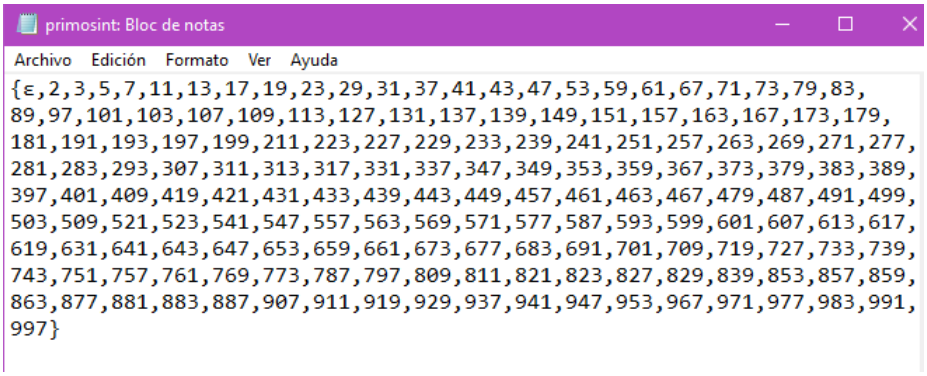
Para esto creamos 2 archivos, uno con el nombre de primosbin y otro con el nombre de primosint, ambos de tipo txt y estos almacenarán según sea el caso, el número primo encontrado. Generamos un bucle que va empieza desde 2, ya que de lo contrario tomaría al 1 como primo y no pertenece a esa familia. Realiza la comparación con cada uno de los valores , llamando a la función Primo y mandandole el valor de la posición en la que va nuestro for. Cada que sea primo, lo convierte a binario y se mandan a escribir a cada uno de los archivos según correspondan.

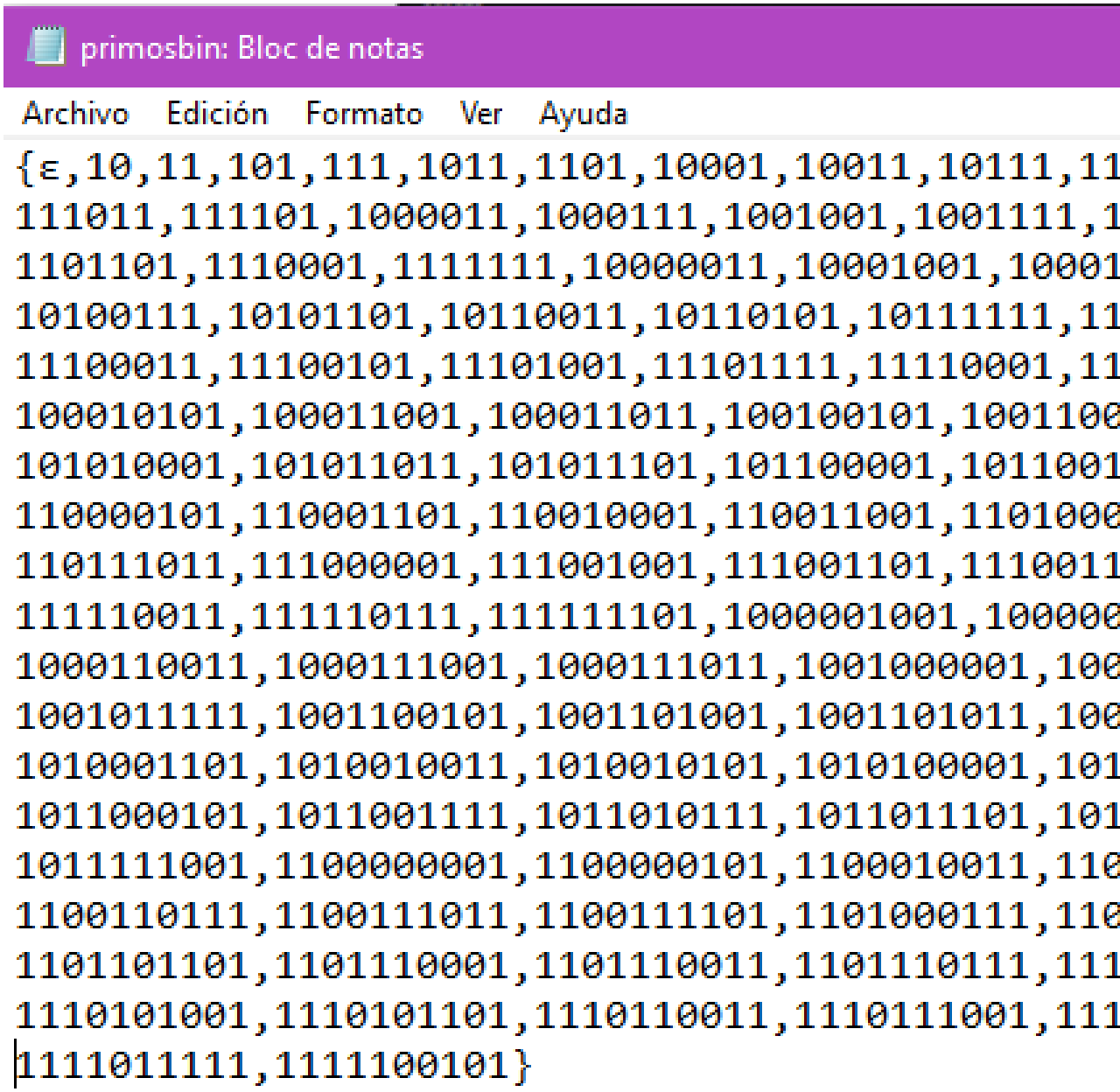
Para graficar, hacemos uso de la librería matplotlib. El programa tiene una función en onde recibe los arreglos llenados en la función sigma, que usa para representarlos en la gráfica.

3.1 Implementación



Consola




primosbin: Bloc de notas
Archivo Edición Formato Ver Ayuda
{ε,10,11,101,111,1011,1101,10001,10011,10111,11
111011,111101,1000011,1000111,1001001,1001111,1
1101101,1110001,1111111,10000011,10001001,10001
10100111,10101101,10110011,10110101,10111111,11
11100011,11100101,11101001,11101111,11110001,11
100010101,100011001,100011011,100100101,1001100
101010001,101011011,101011101,101100001,1011001
110000101,110001101,110010001,110011001,1101000
110111011,111000001,111001001,111001101,1110011
111110011,111110111,111111101,1000001001,100000
1000110011,1000111001,1000111011,1001000001,100
1001011111,1001100101,1001101001,1001101011,100
1010001101,1010010011,1010010101,1010100001,101
1011000101,1011001111,1011010111,1011011101,101
1011111001,1100000001,1100000101,1100010011,110
1100110111,1100111011,1100111101,1101000111,110
1101101101,1101110001,1101110011,1101110111,111
1110101001,1110101101,1110110011,1110111001,111
1111011111,1111100101}

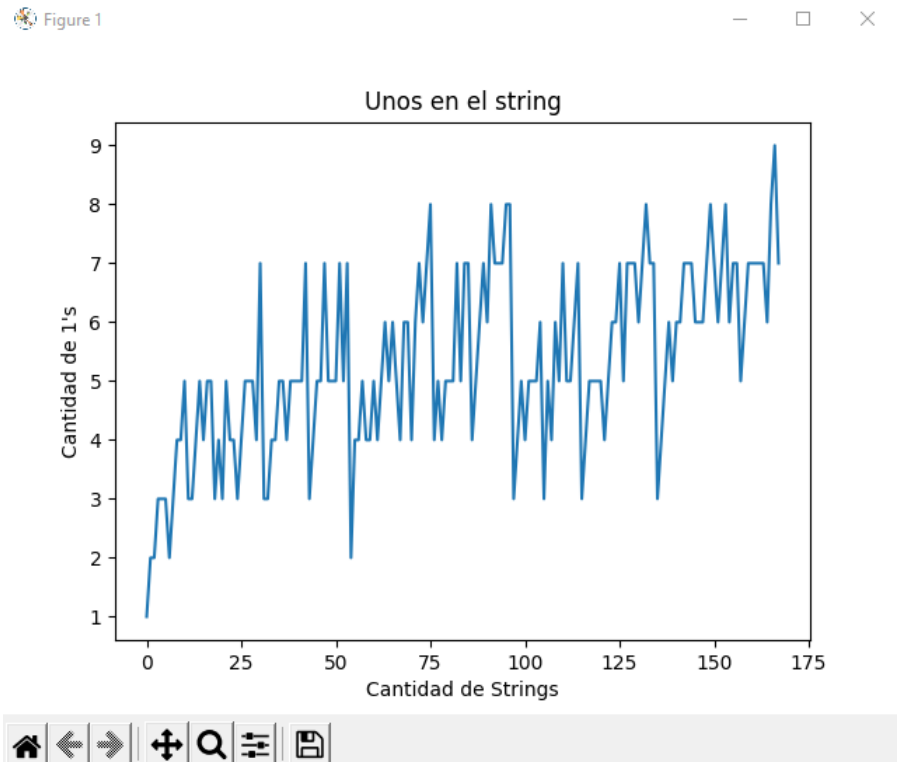


Figure 1
Unos en el string
Cantidad de 1's
Cantidad de Strings
0
25
50
75
100
125
150
175
1
2
3
4
5
6
7
8
9

Grafica 1

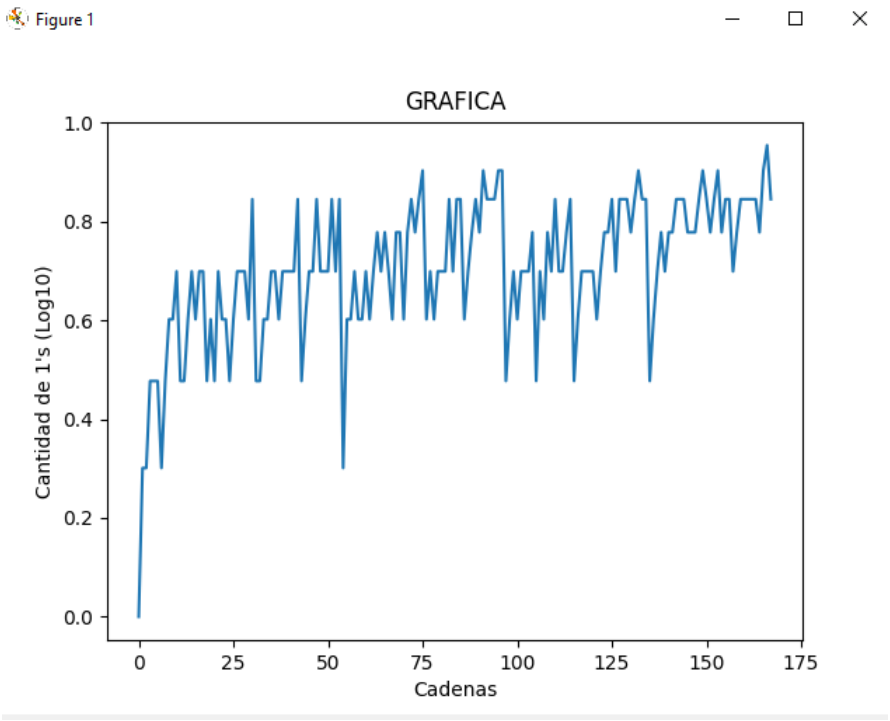
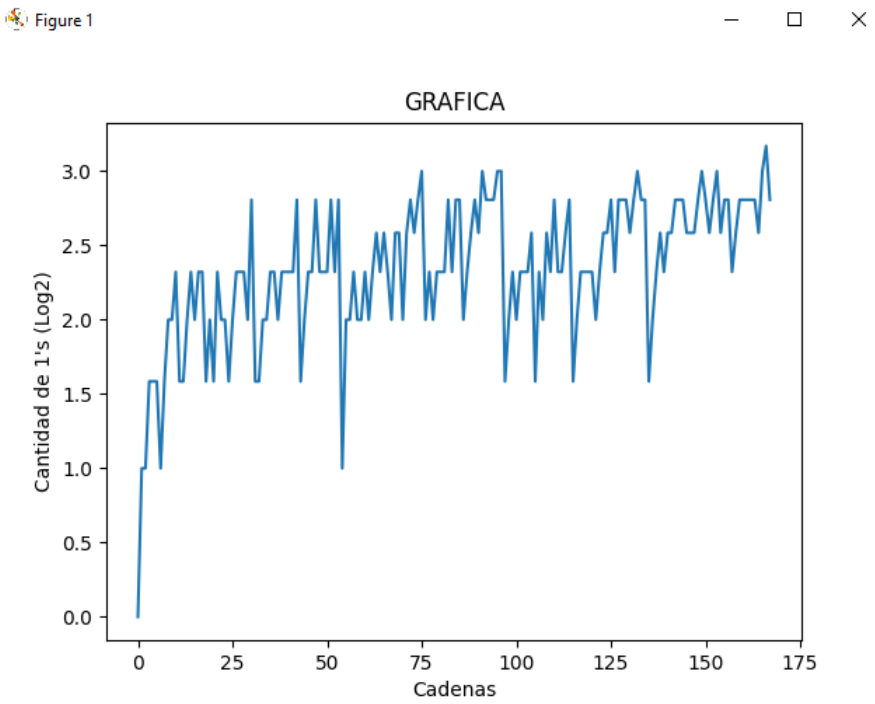


Figure 1
GRAFICA
1.0
0.8
0.6
0.4
0.2
0.0
Cantidad de 1's (Log10)
0
25
50
75
100
125
150
175
Cadenas

Grafica 2



Grafica 3

4 Conclusión

Con la realización de este ejercicio, podemos ver que un trabajo que parece bastante complejo se puede resumir a breves líneas de código. Uno pensaría que para calcular la cantidad de números primos que hay de 1 a una cantidad inmensa es bastante tedioso, claramente si nos ponemos a hacerlo a mano sí, pero utilizando nuestro conocimientos podemos generar cosas como este ejercicio, que bien es cierto que no va a calcular los números primos hasta infinito, pero sí a una cantidad que para nosotros sería de mucho tiempo.

References

[1] Rivera, C. (2022, 14 marzo). Números primos y compuestos: qué son + ejemplos. Smartick. Recuperado 19 de junio de 2022, de https://www.smartick.es/blog/matematicas/numeros-enteros/numeros-primos-y-numeros-compuestos/: