



# Escuela Superior de Cómputo

TEORÍA DE LA COMPUTACIÓN

## PRÁCTICA 2: *Primos*

Autor:

Rodrigo Gerardo Trejo Arriaga

Octubre 2023

## Práctica 2:

### *Primos*

En la teoría de la computación, un alfabeto se define como un conjunto finito de símbolos o caracteres. Estos símbolos son la base fundamental para construir cadenas o secuencias de símbolos. Por ejemplo, en el contexto de la teoría de la computación, es común utilizar el alfabeto 0, 1 para representar símbolos binarios.

Una cadena, también conocida como palabra, es una secuencia finita de símbolos tomados de un alfabeto dado. Estas cadenas desempeñan un papel esencial en la representación de datos y forman la base de los lenguajes formales en la teoría de la computación.

El conjunto universo, por otro lado, hace referencia a un conjunto que contiene todos los objetos posibles que son relevantes para un problema o lenguaje específico. Por ejemplo, si se trata de un lenguaje de programación, el conjunto universo podría consistir en todos los programas concebibles que se pueden escribir en ese lenguaje.

En el contexto de la teoría de la computación, un lenguaje formal se define como un conjunto de cadenas (palabras) construidas a partir de un alfabeto dado. Estos lenguajes formales se utilizan para describir propiedades de las cadenas y son fundamentales en la definición de gramáticas formales, autómatas y en la resolución de diversos problemas relacionados con la informática teórica.

En esta práctica presentaremos un programa que conjunta todos estos conceptos.

### I. Descripción del problema

Este programa calcula el lenguaje binario definido por números primos. El valor de "n" puede ser introducido por el usuario o determinado automáticamente por el programa, y está restringido al rango  $[2, 10^7]$ . El valor de "n" determina hasta qué número se calcularán y cuántos números primos existen en ese intervalo.

1. Ejecute el programa y especifique el valor de "n" que desea utilizar para el cálculo.
2. El programa preguntará si desea calcular otro valor de "n". Salir.
3. Los resultados se guardarán en dos archivos de texto: uno en notación binaria y otro en notación decimal.
4. A continuación, graficaremos el número de unos en cada cadena. El eje "x" representa las cadenas y el eje "y" representa el número de unos en cada cadena.
5. En el informe, se explicará, calculará y graficará el caso específico cuando  $n=200^3$ .
6. Además, se calculará una segunda gráfica utilizando el logaritmo en base 10 del número de unos.

## II. Resultados

## III. Código de Implementación

```
1      '''
2      INSTITUTO POLITÉCNICO NACIONAL
3      ESCUELA SUPERIOR DE CÓMPUTO
4
5      INGENIERÍA EN INTELIGENCIA ARTIFICIAL
6
7      TEORÍA DE LA COMPUTACIÓN
8      PARTICIPACIÓN DE NÚMEROS PRIMOS
9
10     GRUPO: 5BM1
11     ALUMNO: TREJO ARRIAGA RODRIGO GERARDO
12
13     ESTE PROGRAMA CREA UNA PARTICIPACIÓN DEL UNIVERSO DE NÚMEROS PRIMOS
14     Y:
15
16     i) ALMACENA TODOS LOS NÚMEROS QUE SON PRIMOS EN UN RANGO (TANTO EN
17        BINARIO COMO DECIMAL)
18     ii) GENERA UNA TABLA CON TODOS LOS NÚMEROS PRIMOS BINARIOS Y LOS
19         NÚMEROS DE UNOS QUE TIENEN
20     iii) GENERA UNA GRÁFICA DEL NÚMERO PRIMO VS SU NÚMERO DE UNOS
21     DEL NÚMERO DE UNOS
22     PARA APRECIAR MEJOR SU CRECIMIENTO
23
24     ÚLTIMA MODIFICACIÓN: 13/10/2023
25     '''
26
27     #
28     -----
29
30     # MÓDULOS Y LIBRERÍAS IMPORTADAS
31
32     import os
33     import time
34     import csv
35     import matplotlib.pyplot as plt
36     import random
37     import math
38
39     #
40     -----
```



The screenshot shows the Visual Studio Code interface with a file named `primos.py` open. The code in the editor is as follows:

```
1 """  
2 INSTITUTO POLITÉCNICO NACIONAL  
3 ESCUELA SUPERIOR DE CÓMPUTO  
4 INGENIERÍA EN INTELIGENCIA ARTIFICIAL  
5  
6 TEORÍA DE LA COMPUTACIÓN  
7 PARTICIÓN DE NÚMEROS PRIMOS  
8  
9  
10 GRUPO: 5BM1  
11 ALUMNO: TREJO ARRIAGA RODRIGO GERARDO  
12 """
```

The terminal window at the bottom shows the execution of the script:

```
***UNIVERSO DE NÚMEROS PRIMOS***  
>>Deseas que el rango se elija de manera automática o manual? [a/m]: m  
Ingresa el rango máximo para el cálculo de primos [4,10]: 10000000  
>>Calculando números primos...  
  
Listo! Puedes consultar el conjunto de números primos en los archivos 'univ_binprimos.txt' y 'univ_decprios.txt'  
Se han calculado los números primos en un tiempo de: 44.47 segundos  
rodrigo@rogl3:~/Documentos/5toSemestre/TeoriaComputacion/TeoriaComputacion/Primos$
```

Figura 1: Descripción de la imagen

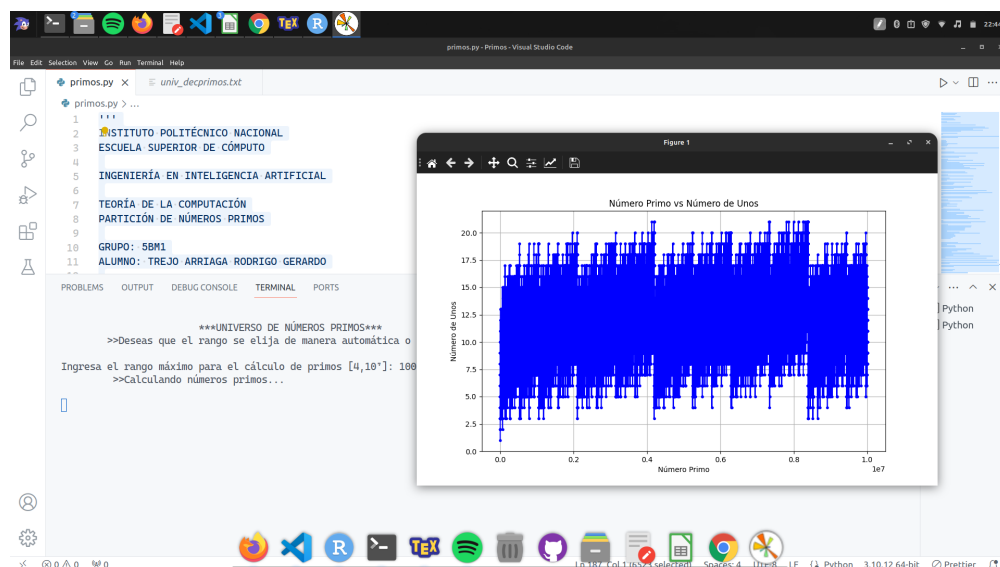


Figura 2: Programa terminado

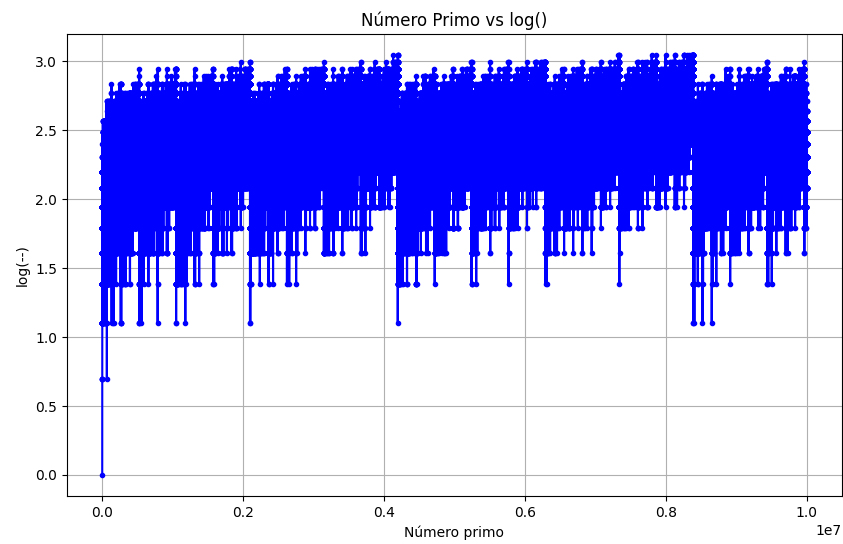


Figura 3: Ejecución con gráfica

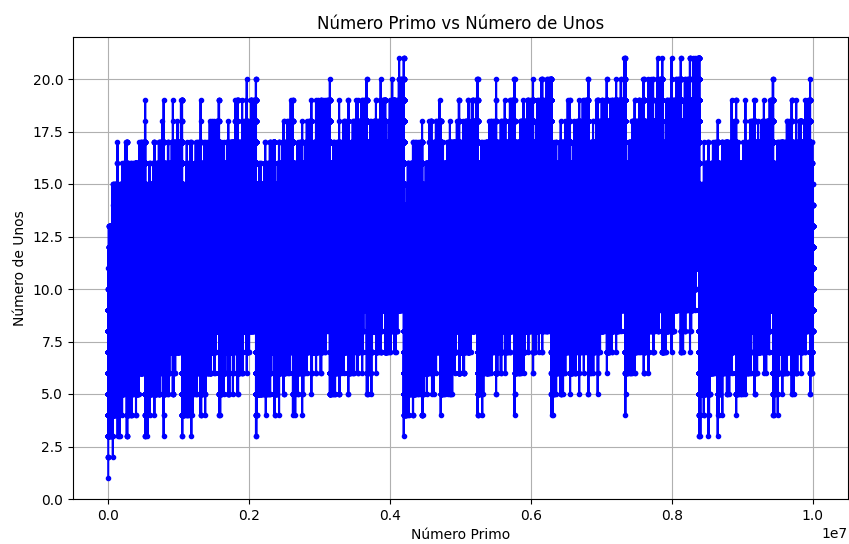


Figura 4: Gráfica Logaritmo

```
36 # CLASES
37
38
39 class ParticionNumPrimos:
40     """Clase que genera la partici n del universo de n meros primos
41         """
42
43     def __init__(self, fin_rango: int) -> None:
44         """Constructor de la clase
45
46     Args:
47     fin_rango (int): L mite del rango de la partici n
48         """
49
50     self.fin_rango = fin_rango
51     self.arch_binprimos = open(f"univ_binprimos.txt", "w+", encoding
52         ="utf-8")
53     self.arch_binprimos.write("    = {")
54     self.arch_decprimos = open(f"univ_decprimos.txt", "w+",
55         encoding="utf-8")
56     self.arch_decprimos.write("    = {")
57     self.arch_tablasUnos = open(f"tabla.csv", "w+")
58     self.arch_tablasUnos.write("Numero, Cantidad de Unos\n" )
59
60
61     def es_primo(self, num_test: int) -> bool:
62         """M todo que indica si un n mero es primo
63
64     Args:
65     num_test (int): N mero que se verifica si es primo
66
67     Returns:
68     bool: Bandera que dice si el n mero es primo o no
69         """
70
71     # Comprobamos si n es igual a 2 ( nico primo par)
72     if num_test == 2:
73         return True
74
75     # Comprobamos si n es menor que 2 o es un n mero par
76     if num_test < 2 or not num_test % 2:
77         return False
78
79     sqrt_n = int(num_test**0.5) + 1 # Calculamos la ra z
80         cuadrada de n + 1 una vez
81
82     # Comprobamos si n es divisible por cualquier entero
83         impar entre 3 y la ra z cuadrada de n
```

```
79         return not any(num_test % i == 0 for i in range(3, sqrt_n
80                        , 2))
81
82     def convertir_a_binario(self, numero: int) -> str:
83         """M todo que convierte un n mero decimal a binario
84
85         Args:
86         numero (int): N mero decimal
87
88         Returns:
89         str: N mero binario
90         """
91         return bin(numero)[2:]
92
93
94     def calcular_rango_primos(self):
95         """M todo que escribe en archivos los n meros primos
96             decimales y binarios
97         """
98         for i in range(2, self.fin_rango+1):
99             bandera_primo = self.es_primo(i)
100             if bandera_primo:
101                 binario = self.convertir_a_binario(i)
102                 self.arch_binprimos.write(f"{binario}, ")
103                 self.arch_decprimos.write(f"{i}, ")
104                 self.arch_tablasUnos.write(f"{i}, {binario.count('1')}\n"
105                                           )
106
107
108     def plotear_grafico(self, x: list, y: list, titulo: str,
109                        label_x: str, label_y: str):
110         """M todo que plotea una gr fica seg n dos listas
111
112         Args:
113         x (list): Lista de datos que ir n en el eje x
114         y (list): Lista de datos que ir n en el eje y
115         titulo (str): T tulo de la gr fica
116         label_x (str): Etiqueta del eje x
117         label_y (str): Etiqueta del eje y
118         """
119         plt.figure(figsize=(10, 6))
120         plt.plot(x, y, marker='o', linestyle='-', color='b',
121                markersize=3)
122         plt.title(titulo)
123         plt.xlabel(label_x)
124         plt.ylabel(label_y)
125         plt.grid(True)
```

```

122     plt.show()
123
124
125
126     def graficar_num_unos(self):
127         """M todo que obtiene la gr fica del n mero de unos
128             por n mero primo
129         """
130         numeros_primos = []
131         numeros_de_unos = []
132
133         with open("tabla.csv", newline='') as archivo_csv:
134             reader = csv.DictReader(archivo_csv)
135             for fila in reader:
136                 try:
137                     numeros_primos.append(int(fila["Numero"]))
138                     numeros_de_unos.append(int(fila[" Cantidad de Unos"]))
139                 except:
140                     print(f"numero = {fila['Numero']} Cantidad = {int(fila['
141                         Cantidad de Unos'])}")
142
143             logaritmo = [0 if valor == 0 else math.log(valor) for
144                 valor in numeros_de_unos]
145
146             self.plotear_grafico(numeros_primos, numeros_de_unos, '
147                 N mero Primo vs N mero de Unos', 'N mero Primo', '
148                 N mero de Unos', )
149
150             self.plotear_grafico(numeros_primos, logaritmo, 'N mero
151                 Primo vs log()', 'N mero primo', 'log(--)' )
152
153             #
154             -----
155
156             # FUNCION PRINCIPAL MAIN
157
158             if __name__ == "__main__":
159                 os.system('clear')
160
161                 print("\t\t\t***UNIVERSO DE N MEROS PRIMOS***")
162                 modo = input("\t>>Deseas que el rango se elija de manera
163                     autom tica o manual? [a/m]: ")
164                 while modo.lower() != "a" and modo.lower() != "m":
165                     print("Modo inv lido , ingres selo nuevamente para
166                         continuar")

```



```
159         modo = input("\t>>Deseas que 'n' se elija de manera
160             autom tica o manual? [a/m]: ")
161
162         match modo:
163             case "m":
164                 n = int(input("\nIngresa el rango m ximo para el
165                     c lculo de primos [4,10   ]: "))
166                 while n > 10000000 and n < 4:
167                     print("Rango incorrecto! Int ntalo nuevamente.")
168                     n = int(input("\nIngresa el rango m ximo para el
169                         c lculo de primos [4,10   ]: "))
170             case _:
171                 n = random.randint(0, 10000000)
172                 print(f"\nSe ha elegido un n={n}")
173
174                 print("\t >>Calculando n meros primos...\n")
175                 inicio = time.perf_counter()
176                 t = ParticionNumPrimos(n)
177                 t.calcular_rango_primos()
178                 fin = time.perf_counter()
179                 t.arch_binprimos.write("{}")
180                 t.arch_binprimos.close()
181                 t.arch_decprimos.write("{}")
182                 t.arch_decprimos.close()
183                 t.arch_tablasUnos.close()
184                 t.graficar_num_unos()
185                 tiempo_transcurrido = fin - inicio
186
187                 print("Listo! Puedes consultar el conjunto de n meros primos en
188                     los archivos 'univ_binprimos.txt' y 'univ_decprimos.txt'")
189
190                 print(f"\nSe han calculado los n meros primos en un tiempo de: {
191                     tiempo_transcurrido:.2f} segundos")
```