



Escuela Superior de Cómputo

TEORÍA DE LA COMPUTACIÓN

PRÁCTICA 7:

LENGUAJES LIBRES DE CONTEXTO - PALÍNDROMOS

Autor:

Rodrigo Gerardo Trejo Arriaga

Diciembre 2023

Práctica 7:

LENGUAJES LIBRES DE CONTEXTO - PALÍNDROMOS

I. Introducción

Un **lenguaje libre de contexto** es un tipo de lenguaje formal que tiene gran importancia en la teoría de la computación y la teoría de autómatas. Los lenguajes libres de contexto son generados por gramáticas libres de contexto y son reconocidos por autómatas de pila [1].

II. Definición [1]

Una gramática libre de contexto (GLC) se define como una cuádrupla $G = (V, \Sigma, R, S)$, donde:

- V es un conjunto finito de variables (también llamadas símbolos no terminales).
- Σ es un conjunto finito de símbolos terminales, disjunto de V .
- R es un conjunto finito de reglas de producción, cada una de la forma $A \rightarrow \beta$, donde $A \in V$ y $\beta \in (V \cup \Sigma)^*$.
- $S \in V$ es el símbolo inicial.

III. Características [2]

1. **Independencia del Contexto:** En una GLC, la producción a aplicar depende solo del símbolo no terminal a reemplazar, y no del contexto (símbolos adyacentes) en el que aparece.
2. **Jerarquía de Chomsky:** Los lenguajes libres de contexto se sitúan en el segundo nivel de la Jerarquía de Chomsky, entre los lenguajes regulares y los lenguajes sensibles al contexto.
3. **Autómatas de Pila:** Estos lenguajes pueden ser reconocidos por autómatas de pila, que son máquinas abstractas con capacidad de almacenamiento adicional en forma de pila.

IV. Ejemplos [2]

Un ejemplo clásico de lenguaje libre de contexto es el lenguaje de paréntesis balanceados, definido por la gramática $G = (\{S\}, \{ (,) \}, R, S)$ con $R = \{ S \rightarrow SS, S \rightarrow (S), S \rightarrow \epsilon \}$.

Los lenguajes libres de contexto son fundamentales en el diseño de lenguajes de programación y en el análisis sintáctico en compiladores.

V. Palíndromos y Lenguajes Libres de Contexto

Un palíndromo es una secuencia de símbolos que se lee igual hacia adelante y hacia atrás. Los lenguajes libres de contexto pueden describir palíndromos de manera eficiente mediante reglas de producción.

La generación de palíndromos utilizando una gramática libre de contexto se basa en la construcción simétrica de la cadena. Por ejemplo, consideremos la gramática $G = (V, \Sigma, R, S)$ con:

- $V = \{P\}$
- $\Sigma = \{0, 1\}$
- $R = \{P \rightarrow \varepsilon, P \rightarrow 0, P \rightarrow 1, P \rightarrow 0P0, P \rightarrow 1P1\}$
- $S = P$

Ventajas

1. **Estructura Simétrica:** Las reglas de producción permiten construir la cadena de manera simétrica, garantizando la propiedad palindrómica.
2. **Flexibilidad:** Los lenguajes libres de contexto pueden generar palíndromos de diferentes longitudes y complejidades.
3. **Automatización:** Facilitan la automatización en la generación de palíndromos, especialmente útil en aplicaciones de procesamiento de lenguajes y criptografía.

VI. Instrucciones

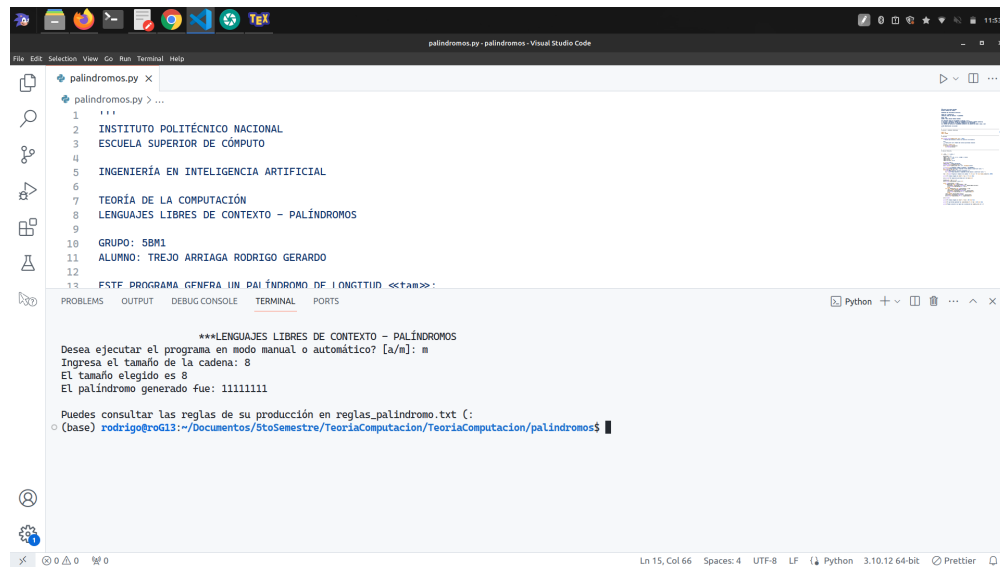
Este programa en Python genera palíndromos en un lenguaje binario según una gramática libre de contexto. El usuario puede elegir la longitud del palíndromo o permitir que el programa lo genere automáticamente. La longitud máxima es de 10,000 caracteres.

Gramática Libre de Contexto

La gramática se define con las siguientes reglas de producción:

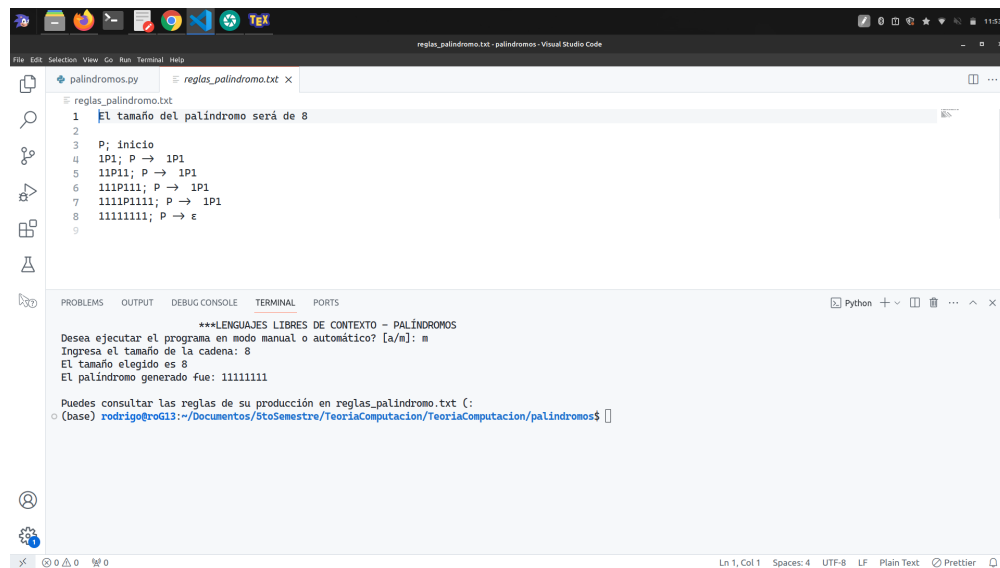
1. $P \rightarrow \varepsilon$
2. $P \rightarrow 0$
3. $P \rightarrow 1$
4. $P \rightarrow 0P0$
5. $P \rightarrow 1P1$

VII. Resultados



```
palindromos.py
1 '''
2 INSTITUTO POLITÉCNICO NACIONAL
3 ESCUELA SUPERIOR DE CÓMPUTO
4 INGENIERÍA EN INTELIGENCIA ARTIFICIAL
5
6 TEORÍA DE LA COMPUTACIÓN
7 LENGUAJES LIBRES DE CONTEXTO - PALÍNDROMOS
8
9 GRUPO: 5BM1
10 ALUMNO: TREJO ARRIAGA RODRIGO GERARDO
11
12 ESTE PROGRAMA GENERA UN PALÍNDROMO DE LONGITUD < tam >:
13
14 ***LENGUAJES LIBRES DE CONTEXTO - PALÍNDROMOS
15 Desea ejecutar el programa en modo manual o automático? [a/m]: m
16 Ingrese el tamaño de la cadena: 8
17 El tamaño elegido es 8
18 El palíndromo generado fue: 11111111
19
20 Puedes consultar las reglas de su producción en reglas_palindromo.txt (:
21 (base) rodrigo@rogi3:~/Documentos/5toSemestre/TeoriaComputacion/TeoriaComputacion/palindromos$
```

Figura 1: Modo manual – Longitud par



```
reglas_palindromo.txt
1 El tamaño del palíndromo será de 8
2
3 P; inicio
4 1P1; P → 1P1
5 11P11; P → 1P1
6 111P111; P → 1P1
7 1111P1111; P → 1P1
8 11111111; P → ε
9
10
11
12
13
14
15
16
17
18
19
20
21
22
23
24
25
26
27
28
29
30
31
32
33
34
35
36
37
38
39
40
41
42
43
44
45
46
47
48
49
50
51
52
53
54
55
56
57
58
59
60
61
62
63
64
65
66
67
68
69
70
71
72
73
74
75
76
77
78
79
80
81
82
83
84
85
86
87
88
89
90
91
92
93
94
95
96
97
98
99
100
```

```
palindromos.py
1 '''
2 INSTITUTO POLITÉCNICO NACIONAL
3 ESCUELA SUPERIOR DE CÓMPUTO
4 INGENIERÍA EN INTELIGENCIA ARTIFICIAL
5
6 TEORÍA DE LA COMPUTACIÓN
7 LENGUAJES LIBRES DE CONTEXTO - PALÍNDROMOS
8
9 GRUPO: 5BM1
10 ALUMNO: TREJO ARRIAGA RODRIGO GERARDO
11
12 ESTE PROGRAMA GENERA UN PALÍNDROMO DE LONGITUD < tam >:
13
14 ***LENGUAJES LIBRES DE CONTEXTO - PALÍNDROMOS
15 Desea ejecutar el programa en modo manual o automático? [a/m]: m
16 Ingrese el tamaño de la cadena: 8
17 El tamaño elegido es 8
18 El palíndromo generado fue: 11111111
19
20 Puedes consultar las reglas de su producción en reglas_palindromo.txt (:
21 (base) rodrigo@rogi3:~/Documentos/5toSemestre/TeoriaComputacion/TeoriaComputacion/palindromos$
```

Figura 2: Modo manual – Longitud par – Archivo

The screenshot shows a Visual Studio Code editor with a Python file named 'palindromos.py'. The code defines a list of institutions and a function to generate a palindrome. The terminal output shows the execution of the script, which prompts for a size and displays the generated palindrome '1000000001'. The status bar at the bottom indicates the file is at line 15, column 66.

```

1  ...
2  INSTITUTO POLITÉCNICO NACIONAL
3  ESCUELA SUPERIOR DE CÓMPUTO
4
5  INGENIERÍA EN INTELIGENCIA ARTIFICIAL
6
7  TEORÍA DE LA COMPUTACIÓN
8  LENGUAJES LIBRES DE CONTEXTO - PALÍNDROMOS
9
10 GRUPO: 5BM1
11 ALUMNO: TREJO ARRIAGA RODRIGO GERARDO
12
13 ESTE PROGRAMA GENERA UN PALÍNDROMO DE LONGITUD <TAM>
14
15 ***LENGUAJES LIBRES DE CONTEXTO - PALÍNDROMOS
16
17 Desea ejecutar el programa en modo manual o automático? [a/m]: m
18 Ingresa el tamaño de la cadena: 11
19 El tamaño elegido es 11
20 El palíndromo generado fue: 1000000001
21
22 Puedes consultar las reglas de su producción en reglas_palindromo.txt (:
23
24 (base) rodrigo13:~/Documentos/9toSemestre/TeoriaComputacion/TeoriaComputacion/palindromos$

```

Figura 3: Modo manual - Longitud impar

The screenshot shows the Visual Studio Code interface with a file named `reglas_palindromo.txt` open. The file contains a context-free grammar for palindromes. The terminal output shows the program running in manual mode, accepting the input '11' and generating the palindrome '1000000001'.

```

1 El tamaño del palindromo será de 11
2
3 P: inicio
4 IP1: P → 1P1
5 10P01: P → 0P0
6 100P001: P → 0P0
7 1000P0001: P → 0P0
8 10000P00001: P → 0P0
9 100000P000001: P → 0
10

```

Terminal Output:

```

***LENGUAJES LIBRES DE CONTEXTO - PALÍNDROMOS
Desea ejecutar el programa en modo manual o automático? [a/m]: m
Ingresa el tamaño de la cadena: 11
El tamaño elegido es 11
El palíndromo generado fue: 1000000001

```

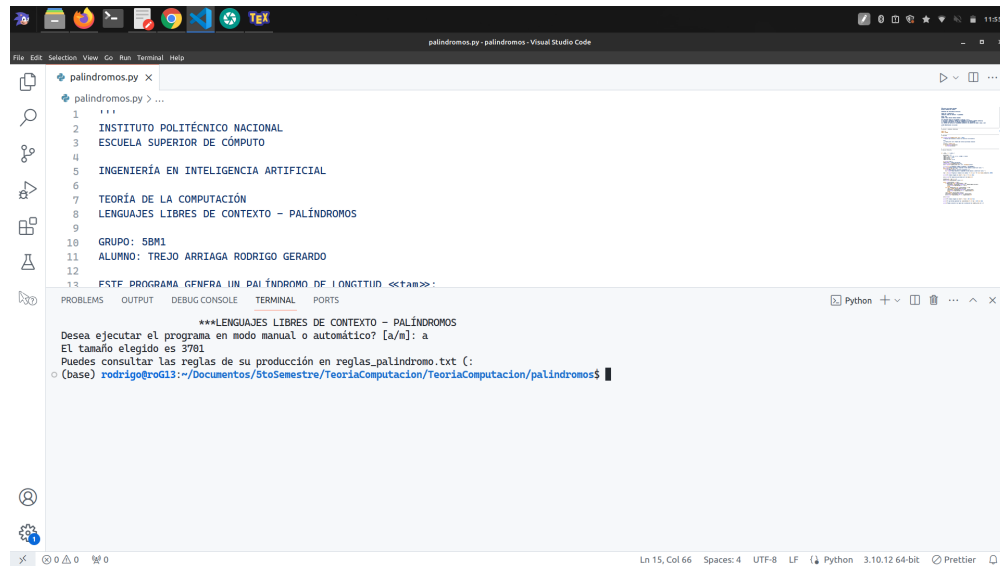
Below the terminal, there is a prompt to consult the rules of the production in `reglas_palindromo.txt`:

```

Puedes consultar las reglas de su producción en reglas_palindromo.txt (:
(base) rodrigo@roG13:~/Documentos/5toSemestre/TeoriaComputacion/TeoriaComputacion/palindromos$

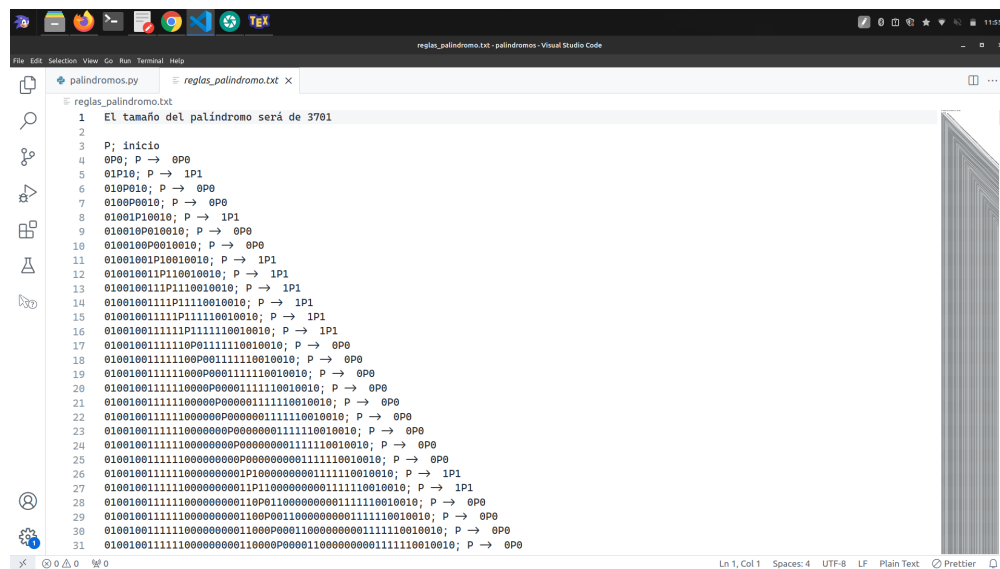
```

Figura 4: Modo manual - Longitud impar - Archivo



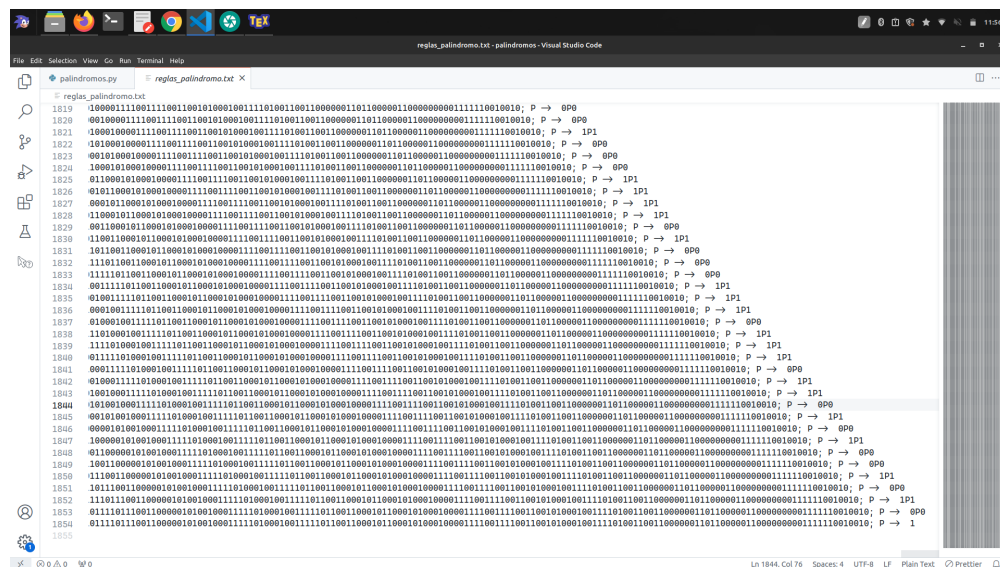
```
palindromos.py
1 '''
2 INSTITUTO POLITÉCNICO NACIONAL
3 ESCUELA SUPERIOR DE CÓMPUTO
4
5 INGENIERÍA EN INTELIGENCIA ARTIFICIAL
6
7 TEORÍA DE LA COMPUTACIÓN
8 LENGUAJES LIBRES DE CONTEXTO - PALÍNDROMOS
9
10 GRUPO: 5BM1
11 ALUMNO: TREJO ARRIAGA RODRIGO GERARDO
12
13 ESTE PROGRAMA GENERA UN PALÍNDROMO DE LONGITUD < tam >:
14
15 ***LENGUAJES LIBRES DE CONTEXTO - PALÍNDROMOS
16 Desea ejecutar el programa en modo manual o automático? [a/n]: a
17 El tamaño elegido es 3701
18 Puedes consultar las reglas de su producción en reglas_palindromo.txt (:
19 (base) rodrigo@rogi3:~/Documentos/StoSemestre/TeoriaComputacion/TeoriaComputacion/palindromos$
```

Figura 5: Modo automático



```
reglas_palindromo.txt
1 El tamaño del palíndromo será de 3701
2
3 P: inicio
4 0P0; P → 0P0
5 01P10; P → 1P1
6 010P010; P → 0P0
7 0100P0010; P → 0P0
8 01001P10010; P → 1P1
9 010010P010010; P → 0P0
10 0100100P010010; P → 0P0
11 01001001P10010010; P → 1P1
12 010010011P10010010; P → 1P1
13 0100100111P1110010010; P → 1P1
14 01001001111P11110010010; P → 1P1
15 010010011111P111110010010; P → 1P1
16 0100100111111P1111110010010; P → 1P1
17 01001001111110P01111110010010; P → 0P0
18 010010011111100P001111110010010; P → 0P0
19 0100100111111000P0001111110010010; P → 0P0
20 01001001111110000P0000111110010010; P → 0P0
21 010010011111100000P00000111110010010; P → 0P0
22 0100100111111000000P000000111110010010; P → 0P0
23 01001001111110000000P0000000111110010010; P → 0P0
24 010010011111100000000P00000000111110010010; P → 0P0
25 0100100111111000000000P000000000111110010010; P → 0P0
26 01001001111110000000001P1000000001111110010010; P → 1P1
27 010010011111100000000011P1000000001111110010010; P → 1P1
28 010010011111100000000010P0100000000111110010010; P → 0P0
29 0100100111111000000000100P00100000000111110010010; P → 0P0
30 01001001111110000000001000P000100000000111110010010; P → 0P0
31 010010011111100000000010000P0000100000000111110010010; P → 0P0
```

Figura 6: Modo automático - Archivo



The image shows a screenshot of the Visual Studio Code editor. The file explorer on the left shows a folder named 'palindromos.py' containing a file 'reglas_palindromo.txt'. The main editor window displays the content of 'reglas_palindromo.txt', which consists of a large number of lines of text. Each line contains a binary string (e.g., '1000011100011100010010100010011101001001100000011011000001100000000111110010010;') followed by a command (e.g., 'P -> 0P0'). The text is wrapped, and the editor shows line numbers on the left margin. The status bar at the bottom indicates 'Ln 1844, Col 76' and 'UTF-8 LF Plain Text'.

Figura 7: Modo automático - Archivo

Conclusión

Esta práctica ha demostrado la eficacia de los lenguajes libres de contexto en la generación de estructuras simétricas como los palíndromos. Hemos explorado cómo una gramática simple puede ser utilizada para generar cadenas palindrómicas de forma sistemática y flexible, lo que demuestra la versatilidad de los lenguajes libres de contexto en la teoría de la computación.

La implementación de un programa en Python para generar palíndromos binarios según la gramática definida ha proporcionado una comprensión práctica y aplicada de estos conceptos. Este enfoque nos ha permitido observar de primera mano la relación entre las reglas de producción de la gramática y la estructura final de los palíndromos generados.

Además, la capacidad de generar palíndromos de manera aleatoria o con una longitud definida por el usuario muestra la flexibilidad y el potencial de este enfoque para diversas aplicaciones, desde el análisis de datos hasta la criptografía.

Así, esta práctica ha reforzado la importancia de los lenguajes libres de contexto en el ámbito de la teoría de autómatas y la computación, destacando su utilidad para resolver problemas complejos de manera estructurada y eficiente.

Bibliografía

[1] AcademiaLab. "Gramática libre de contexto _ AcademiaLab". Home | AcademiaLab. Accedido el 24 de diciembre de 2023. [En línea]. Disponible: <https://academia-lab.com/enciclopedia/gramatica-libre-de-contexto/>

[2] Wikiwand. "Wikiwand – Gramática libre de contexto". Wikiwand. Accedido el 24 de diciembre de 2023. [En línea]. Disponible: https://www.wikiwand.com/es/Gramática_libre_de_contexto

VIII. Anexo - Código de Implementación

```

1      '''
2
3      INSTITUTO POLITECNICO NACIONAL
4      ESCUELA SUPERIOR DE COMPUTO
5
6      INGENIERIA EN INTELIGENCIA ARTIFICIAL
7
8      TEORIA DE LA COMPUTACION
9      LENGUAJES LIBRES DE CONTEXTO - PALINDROMOS
10
11     GRUPO: 5BM1
12     ALUMNO: TREJO ARRIAGA RODRIGO GERARDO
13
14     ESTE PROGRAMA GENERA UN PALINDROMO DE LONGITUD <<tam>>:
15     i) SOLICITA AL USUARIO EL TAMANO DEL PALINDROMO O LO GENERA DE
16         MANERA AUTOMATICA
17     ii) ESCRIBE EN UN ARCHIVO LAS REGLAS DE PRODUCCION DEL
18         PALINDROMO
19     iii) IMPRIME EN PANTALLA EL PALINDROMO OBTENIDO SI SU LONGITUD
20         ES MENOR O IGUAL A 100
21
22     ULTIMA MODIFICACION: 24/12/2023
23     '''
24
25     # -----
26     # MODULOS Y LIBRERIAS IMPORTADAS
27
28     import os
29     import random
30
31     # -----
32     # FUNCIONES
33
34     def eliminar_archs(nombre_arch: str) -> None:
35         """Funcion que elimina un archivo si existe en el
36             directorio
37
38             Args:
39             nombre_arch (str): Nombre del archivo que deseas eliminar
40             """
41
42             archivo1 = nombre_arch
43             if os.path.exists(archivo1):
44                 os.remove(archivo1)
45
46     # -----
47     # FUNCION PPRINCIPAL
48
49     if __name__ == "__main__":
50
51         palindromo = "P"

```

```
48 reglas = {1:"", 2:"0", 3:"1", 4:"OP0", 5:"1P1"}
49 reglas_constr = [4,5]
50 regla_fin_par = 1
51 reglas_fin_imp = [2,3]
52
53 os.system('clear')
54 nombre_arch = "reglas_palindromo"
55 eliminar_archs(f"{nombre_arch}.txt")
56 arch = open(f"{nombre_arch}.txt", "a+", encoding="utf-8")
57
58 print("\t\t\t\t\t***LENGUAJES LIBRES DE CONTEXTO - PALINDROMOS"
59       )
60 op = input("Desea ejecutar el programa en modo manual o
61           automatico? [a/m]: ")
62 while op!="a" and op != "m":
63     print("Modo invalido, intentelo nuevamente :")
64     op = input("Desea ejecutar el programa en modo manual o
65               automatico? [a/m]: ")
66
67 tam = int(input("Ingresa el tamano de la cadena: ")) if op
68     == "m" else random.randint(0, 10000)
69
70 print(f"El tamano elegido es {tam}") if op == "a" else None
71
72 arch.write(f"El tamano del palindromo sera de {tam}\n\n")
73
74 bandera_par = tam % 2 == 0
75 arch.write(f"{palindromo}; inicio \n")
76
77 while len(palindromo) <= tam+1:
78     if bandera_par and len(palindromo) > tam:
79         palindromo = palindromo.replace("P", reglas[regla_fin_par])
80         arch.write(f"{palindromo}; P -> e\n")
81         break
82     elif not bandera_par and len(palindromo) == tam:
83         rand = random.choice(reglas_fin_imp)
84         palindromo = palindromo.replace("P", reglas[rand])
85         arch.write(f"{palindromo}; P -> {reglas[rand]}\n")
86         break
87     rand = random.choice(reglas_constr)
88     palindromo = palindromo.replace("P", reglas[rand])
89     arch.write(f"{palindromo}; P -> {reglas[rand]}\n")
90
91 arch.close()
92
93 print(f"El tamano elegido es {tam}") if tam <= 100 else
94     None
95
96 print(f"El palindromo generado fue: {palindromo}\n") if tam
97     <= 100 else None
98
99 print(f"Puedes consultar las reglas de su produccion en {
100       nombre_arch}.txt (:")
```