



INSTITUTO POLITÉCNICO NACIONAL  
ESCUELA SUPERIOR DE CÓMPUTO

## Práctica 1 - Operaciones con cadenas

Unidad de aprendizaje: Teoría computacional

Grupo: 2CM4

*Alumno(a):*

Nicolás Sayago Abigail

*Profesor(a):*

Sanchez García Luz María

22 de febrero de 2018

# Índice

<b>1</b>	<b>Introducción</b>	<b>2</b>
<b>2</b>	<b>Planteamiento del problema</b>	<b>2</b>
<b>3</b>	<b>Diseño de la solución</b>	<b>5</b>
<b>4</b>	<b>Implementación de la solución</b>	<b>12</b>
<b>5</b>	<b>Funcionamiento</b>	<b>16</b>
<b>6</b>	<b>Conclusiones</b>	<b>22</b>
	<b>Referencias</b>	<b>22</b>

# Operaciones con cadenas

## Reporte 1

22 de febrero de 2018

### 1. Introducción

En la siguiente práctica se desarrollaran las operaciones básicas de las cadenas que pertenecen a cierto lenguaje establecido. El lenguaje de programación en el que se desarrolla la práctica es JAVA. Cuenta con una interfaz amigable para el usuario permitiendo ingresar la cadena que el usuario desee.

A lo largo del documento usaremos la palabra **cadena**, por lo cual tenemos que definir ¿Qué es una cadena? y ¿A qué pertenece?.

Se llama alfabeto a un conjunto finito, no vacío. Los elementos de un alfabeto se llaman símbolos. Un alfabeto se define por la enumeración de los símbolos que contiene. Una palabra o **cadena** es aquella formada con los símbolos de un alfabeto. Dichas cadenas poseen operaciones y propiedades, que en esta práctica se pretenden diseñar e implementar.

Las operaciones básicas de las cadenas son:

- Longitud
- Concatenación
- Potencia
- Inverso o reflejo
- Prefijo
- Sufijo
- Subcadena

### 2. Planteamiento del problema

Diseñar la solución para cada una de las operaciones que se pueden hacer con las cadenas:

- Determinar si es palíndromo

Una palabra es palíndromo si dice lo mismo al derecho y al revés, es algo así como: de ida y vuelta. El problema es determinar si una cadena es palíndromo.

Ejemplo:

Sea  $u = \text{ana}$ , entonces  $u^{-1} = \text{ana}$ . Por lo tanto, la cadena es palíndromo.

La solución al problema, es usar la operación de invertir la cadena, y compararla con la original.

### ■ Cálculo de Longitud

Se llama longitud de una cadena al número de símbolos que la componen. La longitud de la cadena  $x$  se representa con la notación  $|x|$ . La cadena cuya longitud es cero se llama *cadena vacía* y se representa con la letra griega  $\lambda$ .

En el lenguaje de programación a usar, existe un método que proporciona la longitud de una cadena.

### ■ Concatenación de cadenas

Sean  $u$  y  $v$  dos cadenas sobre el mismo alfabeto, la concatenación de  $u$  y  $v$  es una nueva cadena  $w$  que se obtiene yuxtaponiendo primero  $u$  y detrás  $v$ , se escribe:  $w = uv$ .

En el lenguaje de programación a usar, existe un método que permite concatenar las cadenas. Evidentemente la concatenación no es asociativa, y el método permite que esta propiedad se cumpla.

### ■ Potencias

Sea  $u$  una cadena y  $k$  un número entero, definimos:

$$\left. \begin{array}{ll} w \dots^{k)} \dots w & \text{si } k > 0 \\ \lambda & \text{si } k = 0 \\ w^{-1} \dots^{-k)} \dots w^{-1} & \text{si } k < 0 \end{array} \right\} u^k$$

Se concatena la cadena  $k$  veces. Para esto se crean subcadenas, hasta obtener la que queremos.

### ■ Inverso o reflejo

Sea  $u$  una cadena sobre cierto alfabeto. Llamamos inversa o reflejada de la cadena  $u$ , y la representamos por  $u^{-1}$ , a la cadena obtenida al escribir los símbolos que constituyen la cadena  $u$  en orden inverso. Si  $u = a_1, a_2, \dots, a_n$ , su reflejada sería  $u = a_n, a_{n-1}, \dots, a_2, a_1$ .

Para este punto tenemos dos soluciones, la que he usado es que se crea una nueva cadena que es resultado de obtener el último símbolo de la original hasta llegar al primero.

### ■ Prefijos

Un prefijo de la cadena  $u$  es cualquier cadena que se obtiene al eliminar cero o más símbolos del final de  $s$ .

La solución es usar un método que obtendrá cierta posición de la cadena original para ir concatenando con lo que se va teniendo.

### ■ Sufijos

Un prefijo de la cadena  $u$  es cualquier cadena que se obtiene al eliminar cero o más símbolos del principio de  $s$ .

La solución es usar un método que obtendrá cierta posición de la cadena original para ir concatenando con lo que se va teniendo.

- Subcadenas

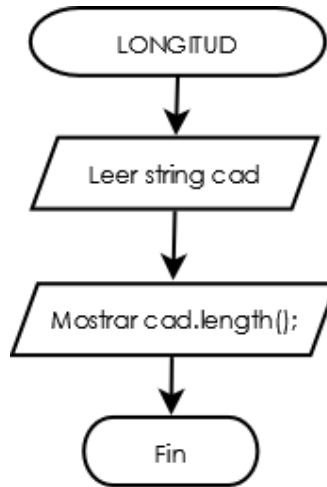
Una subcadena de  $S$  se obtiene al eliminar cualquier prefijo y cualquier sufijo de  $S$ .

La solución es usar un método que obtendrá cierta posición de la cadena original para ir concatenando con lo que se va teniendo. En este caso en particular, se usa la misma técnica que en prefijos.

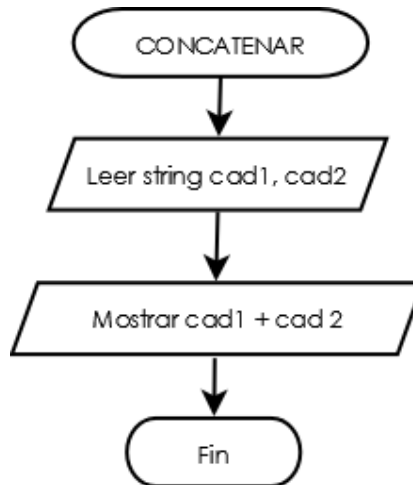
### 3. Diseño de la solución

A continuación mostraré el diseño que se tiene para cada una de las operaciones a realizar.

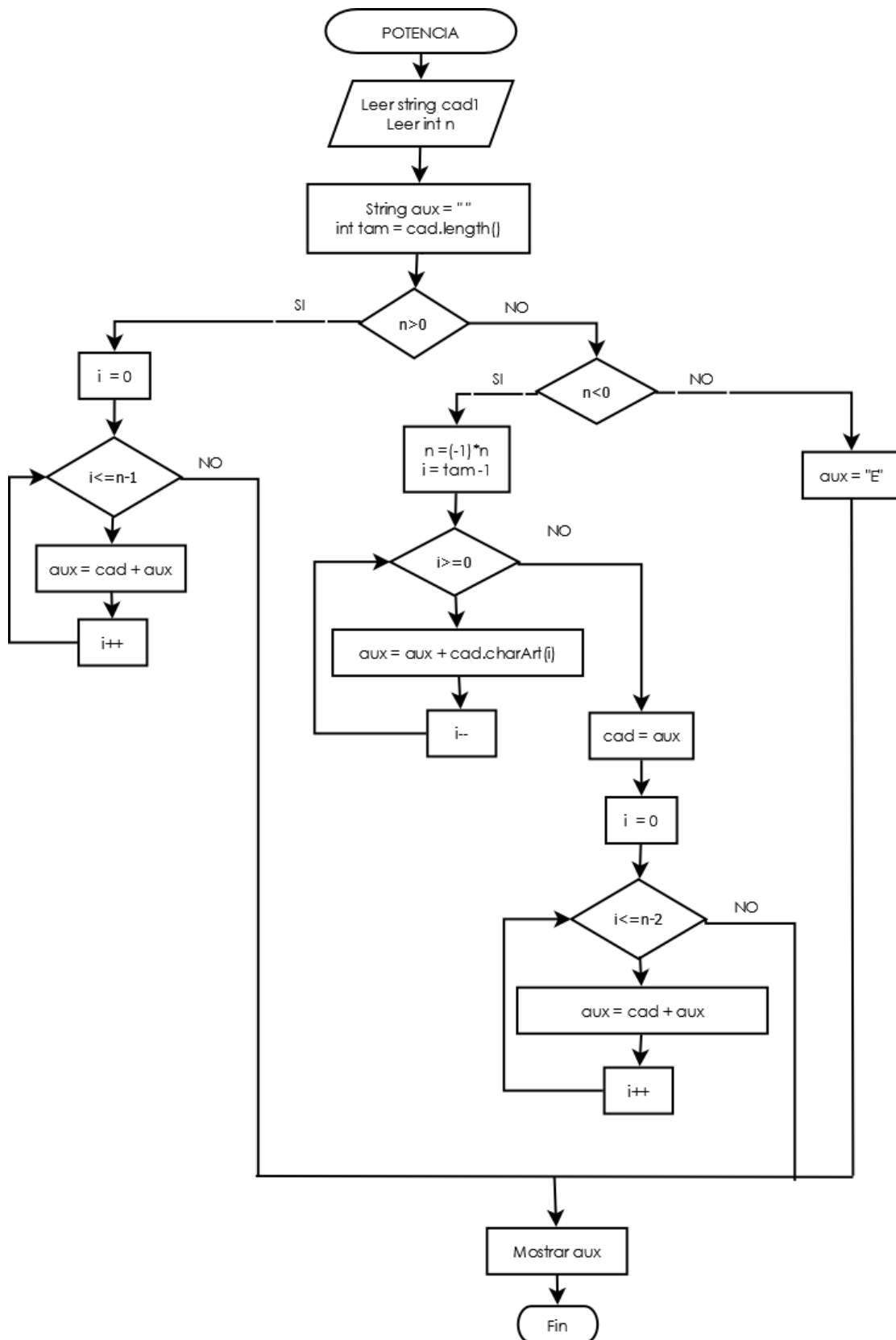
- Cálculo de Longitud



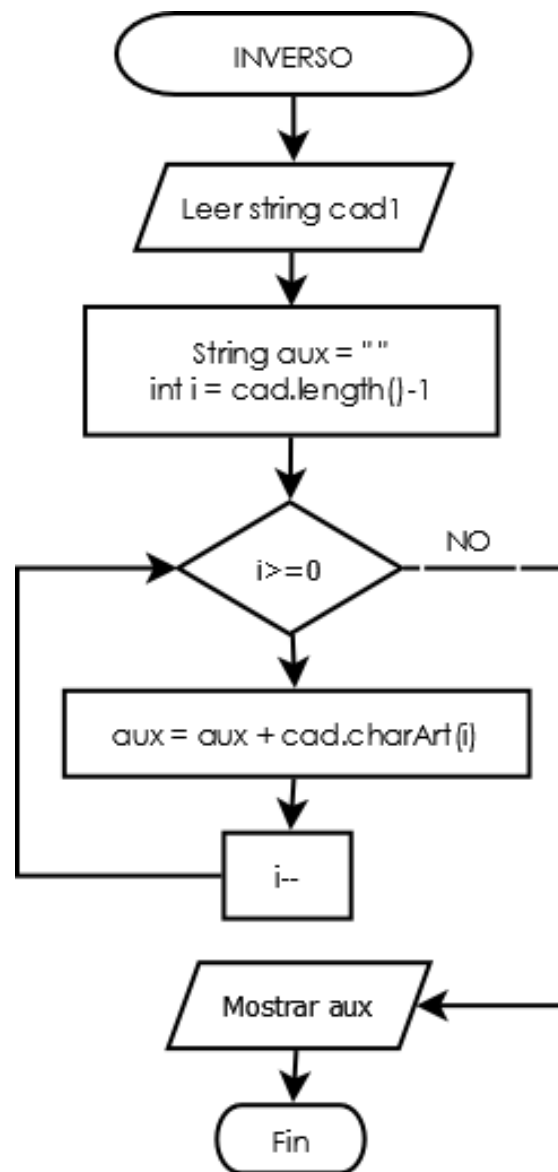
- Concatenación de cadenas



## ■ Potencias

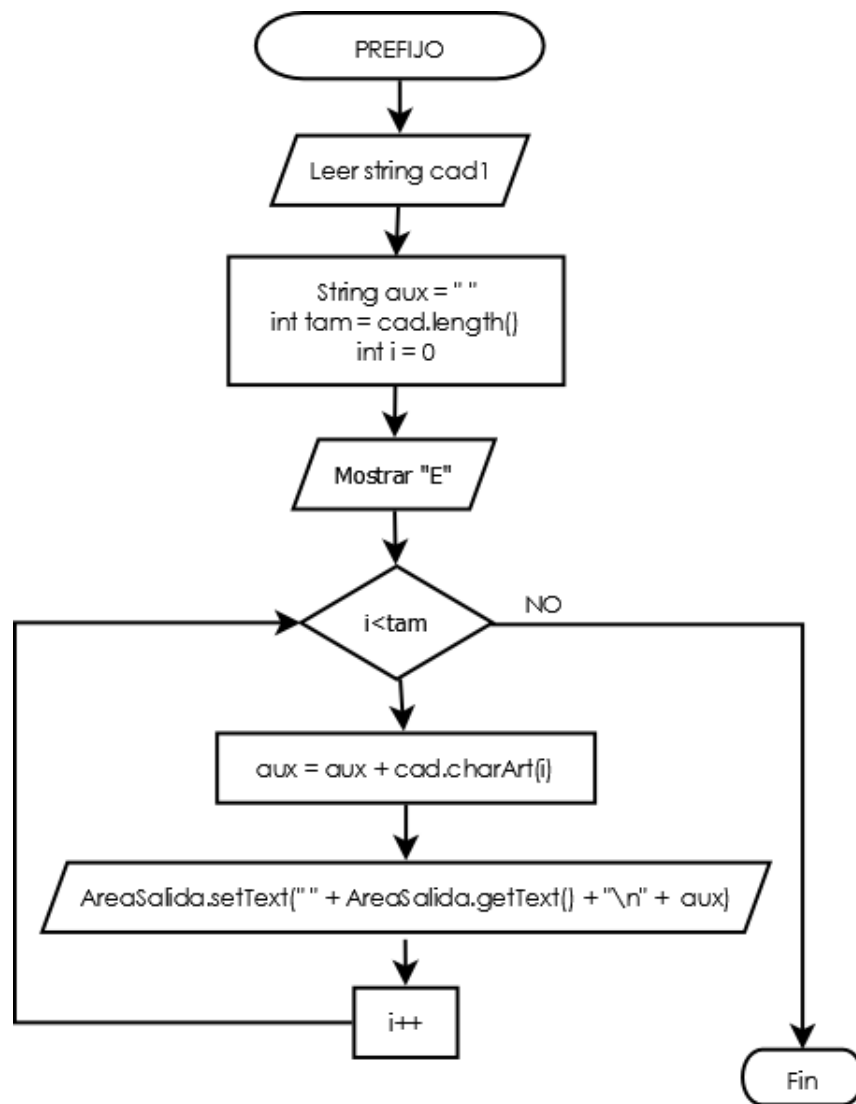


## ■ Inverso o reflejo

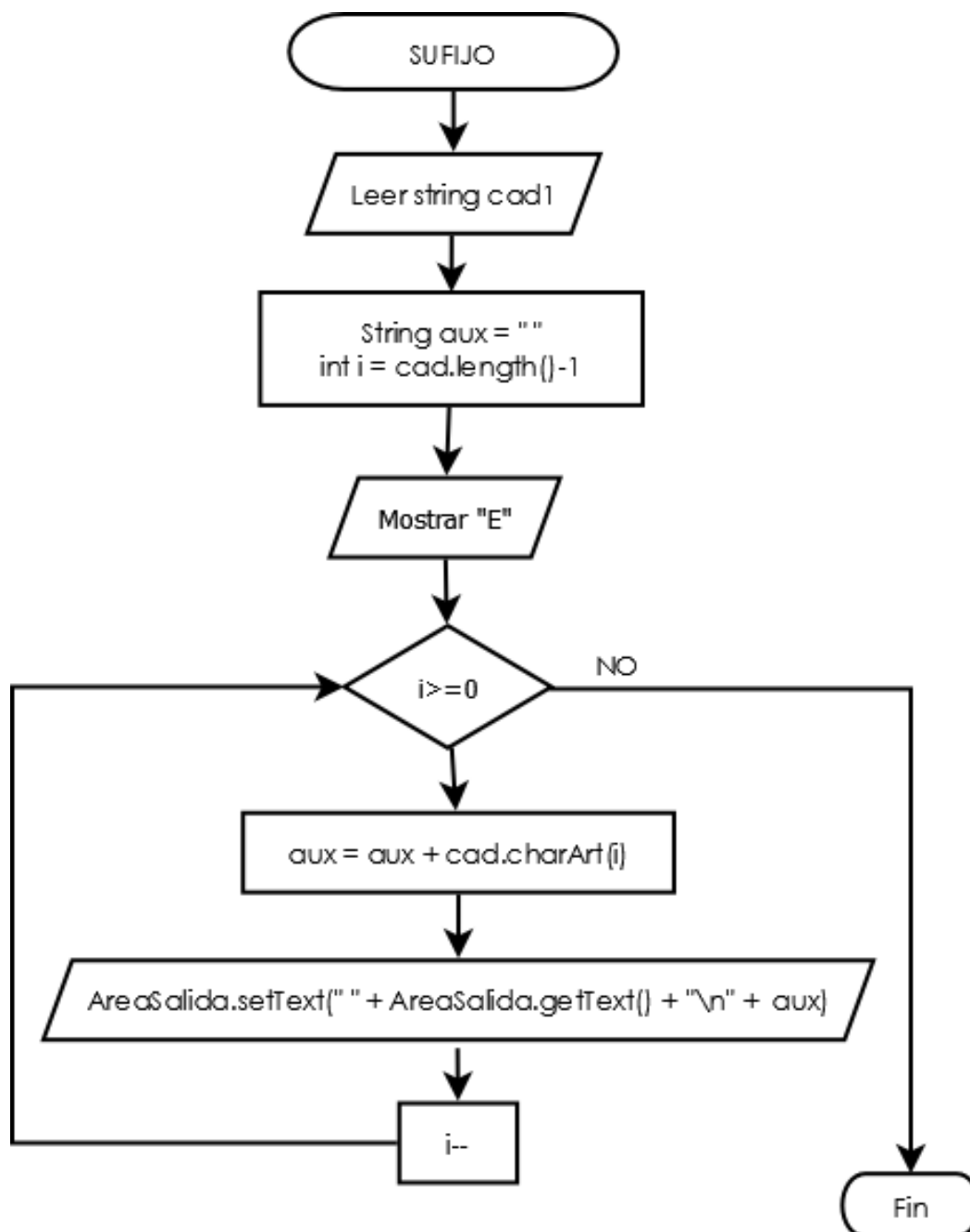




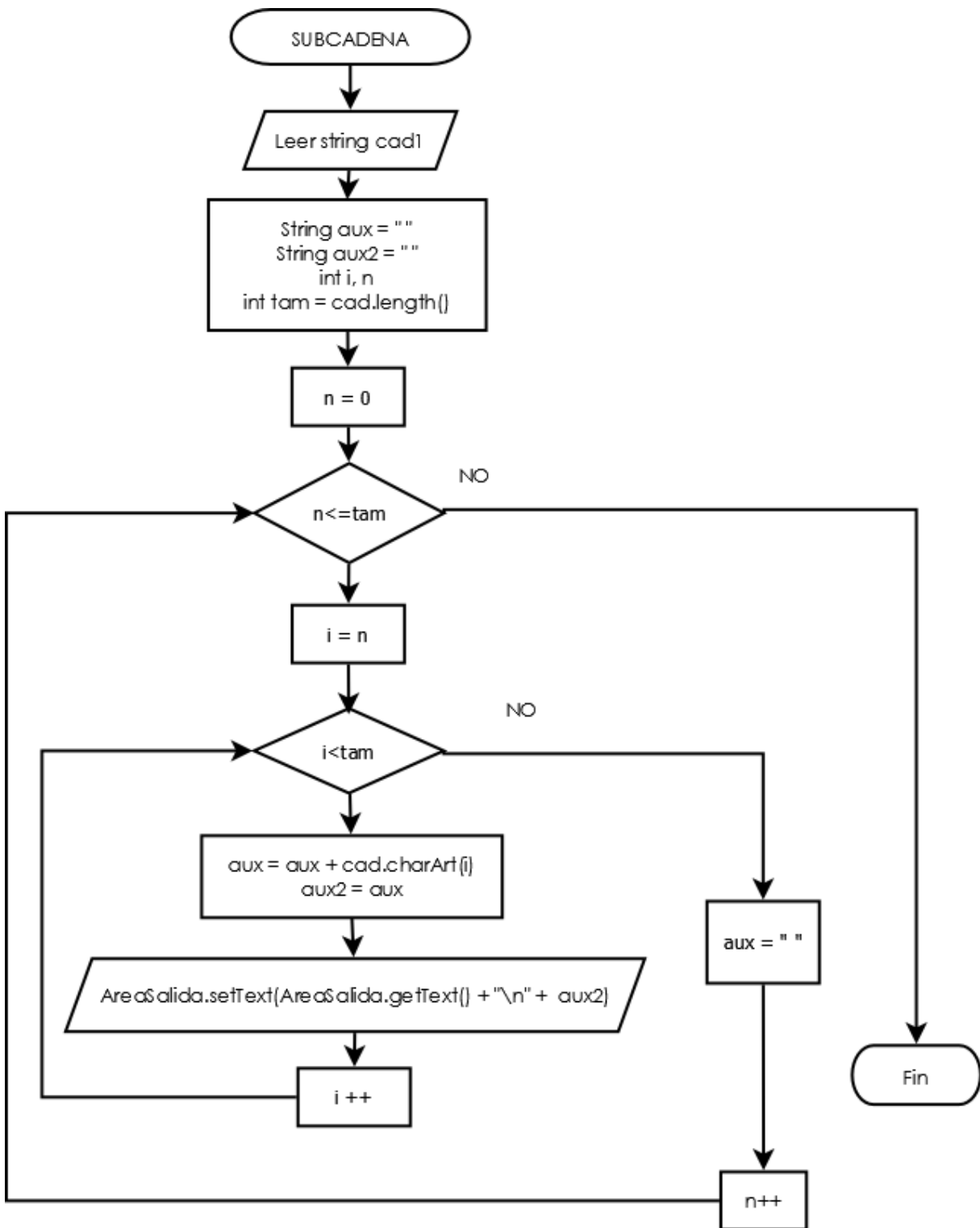
## ■ Prefijos



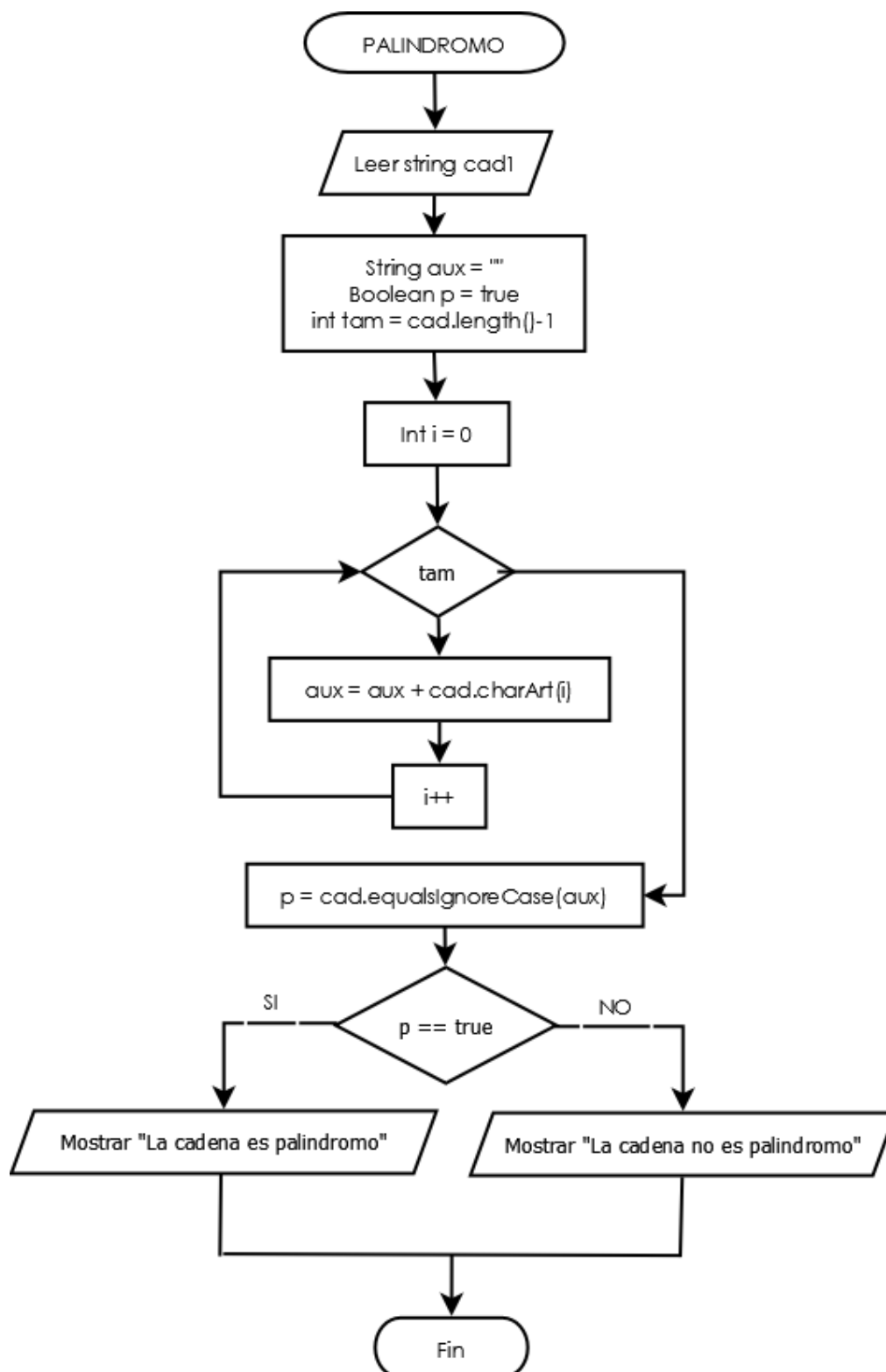
## ■ Sufijos



## ■ Subcadenas



- Determinar si es palíndromo



## 4. Implementación de la solución

El programa consta de 9 métodos de los cuales 8 corresponden a cada una de las operaciones que se requieren.

Primero muestro una tabla con los métodos con los que ya cuenta el lenguaje de programación y que fueron utilizados en la realización de las operaciones.

Método	Función
<code>cadena.charAt(indice)</code>	Devuelve el caracter que encuentra en una posición específica.
<code>cadena.length()</code>	Devuelve la longitud de la cadena tomando en cuenta los espacios.
<code>string.equalsIgnoreCase(string2)</code>	Devuelve true si una cadena es igual a la otra y false en caso contrario.

### ■ Determinar si es palíndromo

```

1  public void Palindromo(int sel)
2  {
3      int i;
4      Boolean p = true; // Bandera
5      String cad = campos[sel].getText(); //Cadena elegida
6      String aux=""; //Cadena auxiliar para guardar el inverso
7      //Invertimos la cadena
8      for (i = cad.length()-1; i>=0; i--)
9          aux = aux + cad.charAt(i);
10     // Comparamos la original con la invertida
11     p = cad.equalsIgnoreCase(aux);
12     if(p == true) // Si la bandera no ha sido modificada
13         AreaSalida.setText("LA CADENA ES PALINDROMO");
14     else
15         AreaSalida.setText("LA CADENA NO ES PALINDROMO");
16 }

```

### ■ Calculo de Longitud

```

1  public void longi(int sel)
2  {
3      String cad = campos[sel].getText(); //Obtenemos la cadena
4      String mensaje = "LA LONGITUD DE LA CADENA ES: \n\n";
5      if (cad.equalsIgnoreCase("E")) // Para la cadena vac a
6          AreaSalida.setText(mensaje + "|" + cad + "|" + " = " + "0" + "\n");
7      else // Para cualquier otra cadena
8          AreaSalida.setText(mensaje + "|" + cad + "|" + " = " + cad.length() + "\n");
9  }

```

### ■ Concatenación de cadenas

```
1 public void Concatenar()
2 {
3     String cad1 = campos[0].getText(); //Obtenemos la primera cadena
4     String cad2 = campos[1].getText(); //Obtenemos la segunda cadena
5     String mensaje = "CONCATENACION DE LA CADENA 1 Y CADENA 2: \n\n";
6     AreaSalida.setText( mensaje + "\n\n" + cad1 + cad2 ); // Se concatenan
7 }
```

### ■ Potencias

```
1 public void Potencia(int sel)
2 {
3     int tam, i;
4     String cad = campos[sel].getText(); // Se obtiene la cadena
5     int n = Integer.parseInt(campos[3].getText()); // Se obtiene la potencia
6     String aux=""; // Cadena auxiliar para guardar la nueva cadena
7     tam = cad.length(); // Tama o de la cadena
8     // CASO: La potencia es positiva
9     if(n > 0)
10    {
11        // Concatenamos la cadena consigo misma n veces
12        for (i=0; i<=n-1; i++)
13            aux = cad + aux;
14    }
15    // CASO: La potencia es negativa
16    else if(n<0)
17    {
18        // Invertimos la cadena
19        n=-1*n;
20        for (i = tam-1; i>=0; i--)
21            aux = aux + cad.charAt(i);
22        cad = aux;
23        // Concatenamos la cadena consigo misma n veces
24        for (i=0; i<=n-2; i++)
25            aux = cad + aux;
26    }
27    // CASO: La potencia es 0
28    else if(n == 0)
29        aux = "E"; // Cadena vac a
30    // Se muestra la nueva cadena
31    AreaSalida.setText("LA POTENCIA ES:\n\n" + aux + "\n");
32 }
```

### ■ Inverso o reflejo

```
1 public void Inverso(int sel)
2 {
3     String cad = campos[sel].getText(); //Se obtiene la cadena
4     String aux=""; // Cadena auxiliar para guardar el inverso
5     int i, tam;
6     tam = cad.length(); // Tama o de la cadena
7     //Invertimos la cadena
8     for (i = tam-1; i>=0; i--)
9         aux = aux + cad.charAt(i);
10    // Se muestra la cadena auxiliar
11    AreaSalida.setText("LA CADENA INVERTIDA ES:" + "\n\n" + aux);
12 }
```

### ■ Prefijos

```
1 public void Prefijo(int sel)
2 {
3     int tam, i;
4     String cad = campos[sel].getText(); // Obtenemos a cadena
5     String aux = ""; // Cadena auxiliar para guardar la nueva cadena
6     tam = cad.length(); // Tama o de la cadena
7     AreaSalida.setText("E"); // Se imprime la cadena vac a
8     // Se imprimen los prefijos, recorriendo cada simbolo de la cadena
9     for (i=0; i<tam; i++) // Iniciamos del primer simbolo
10    {
11        aux = aux + cad.charAt(i); //Nueva cadena
12        AreaSalida.setText(" " + AreaSalida.getText() + "\n" + aux);
13    }
14 }
```

### ■ Sufijos

```
1 public void Sufijo(int sel)
2 {
3     int tam, i;
4     String cad = campos[sel].getText(); // Se obtiene la cadena
5     String aux = ""; // Cadena auxiliar para guardar la nueva cadena
6     tam = cad.length(); // Tama o de la cadena
7     AreaSalida.setText("E"); // Se imprime la cadena vac a
8     // Se imprimen los sufijos, recorriendo cada simbolo de la cadena
9     for(i=tam-1; i>=0; i--) // Iniciamos de ltimo simbolo
10    {
11        aux = cad.charAt(i) + aux; // Nueva cadena
12        AreaSalida.setText(" " + AreaSalida.getText() + "\n" + aux);
13    }
14 }
```

## ■ Subcadenas

```
1 public void Subcadena(int sel)
2 {
3     String cad = campos[sel].getText(); // Obtenemos la cadena
4     String aux = ""; // Cadena auxiliar para guardar cada subcadena
5     String aux2 = ""; // Cadena auxiliar para imprimir cada subcadena
6     int i, tam, n;
7     tam = cad.length(); // Tama o de la cadena
8     n = 0; // Variable para iniciar desde cierto simbolo
9     while(n<=tam) // Para que no se rebase el limite
10    {
11        for (i=n; i<tam; i++) // Se inicia en el simbolo indicado
12        {
13            aux = aux + cad.charAt(i); // Generamos la nueva cadena
14            aux2 = aux; // Guardamos en otra cadena para imprimir
15            // Mostramos todo lo que ya ten amos y la nueva cadena
16            AreaSalida.setText(AreaSalida.getText() + "\n" + aux2);
17        }
18        aux = " "; // Limpiamos la cadena auxiliar
19        n++; // Se recorre el indice de inicio
20    }
21 }
```



## 5. Funcionamiento

Primero que nada, mostramos la interfaz inicial. Observamos que el usuario tiene la oportunidad de ingresar dos cadenas. Y tiene que elegir si quiere ver los resultados de la cadena 1 o 2, unicamente poniendo esos números en el campo correspondiente. Y existe un campo especial para escribir la potencia. Cada botón permite realizar una operación.

The screenshot shows a window titled "Operaciones con cadenas". It contains two input fields for "CADENA 1" and "CADENA 2". Below "CADENA 1" is a label "ELIGE LA CADENA 1 o 2" and a corresponding input field. Below "CADENA 2" is a label "ESCRIBE UNA POTENCIA" and a corresponding input field. In the center, there is a large rectangular area labeled "BIENVENIDO". At the bottom, there are eight buttons arranged in two columns: Palindromo, Longitud, Concatenar, Potencia, Inverso, Prefijo, Sufijo, and Subcadena.

### ■ Calculo de Longitud

The screenshot shows the same window as before, but with data entered. "CADENA 1" contains "HolaMundo" and "CADENA 2" contains "TeoriaComputacional". The "ELIGE LA CADENA 1 o 2" field contains "1". The "ESCRIBE UNA POTENCIA" field is empty. The central area now displays "LA LONGITUD DE LA CADENA ES:" followed by " $|HolaMundo| = 9$ ".

A Java Swing window titled "Operaciones con cadenas" with a standard Mac OS X title bar. It contains two columns of input fields. The first column, labeled "CADENA 1", has two fields: the top one contains "HolaMundo" and the bottom one contains "2". The second column, labeled "CADENA 2", has two fields: the top one contains "TeoriaComputacional" and the bottom one contains "ESCRIBE UNA POTENCIA". Below these fields is a large rectangular text area. Inside this area, the text "LA LONGITUD DE LA CADENA ES:" is followed by the result "[TeoriaComputacional] = 19".

#### ■ Concatenación de cadenas

A Java Swing window titled "Operaciones con cadenas" with a standard Mac OS X title bar. It contains two columns of input fields. The first column, labeled "CADENA 1", has two fields: the top one contains "HolaMundo" and the bottom one contains "2". The second column, labeled "CADENA 2", has two fields: the top one contains "TeoriaComputacional" and the bottom one contains "ESCRIBE UNA POTENCIA". Below these fields is a large rectangular text area. Inside this area, the text "CONCATENACION DE LA CADENA 1 Y CADENA 2:" is followed by the result "HolaMundoTeoriaComputacional".

#### ■ Potencias

A Java Swing window titled "Operaciones con cadenas" with a standard Mac OS X title bar. It contains two columns of input fields. The first column, labeled "CADENA 1", has two fields: the top one contains "HolaMundo" and the bottom one contains "2". The second column, labeled "CADENA 2", has two fields: the top one contains "TeoriaComputacional" and the bottom one contains "3". Below these fields is a large rectangular text area. Inside this area, the text "LA POTENCIA SEGUN EL NUMERO DADO ES:" is followed by the result "TeoriaComputacionalTeoriaComputacionalTeoriaComputacional".

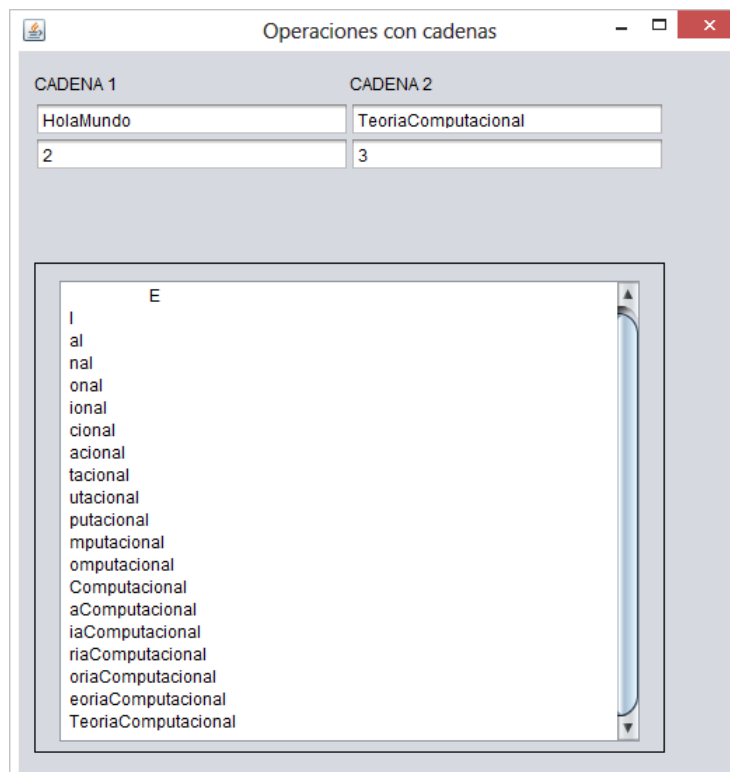
- Inverso o reflejo

The screenshot shows a Java Swing window titled "Operaciones con cadenas". It contains two input fields labeled "CADENA 1" and "CADENA 2". "CADENA 1" contains the text "HolaMundo" and "1". "CADENA 2" contains the text "TeoriaComputacional" and "3". Below the input fields, there is a text area labeled "LA CADENA INVERTIDA ES:" which displays the reversed string "odnuMaloH".

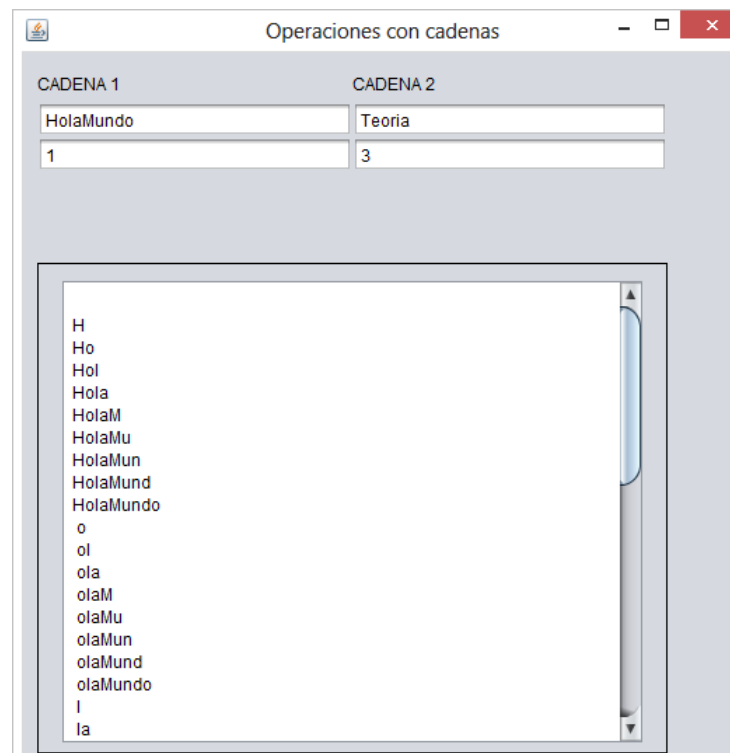
- Prefijos

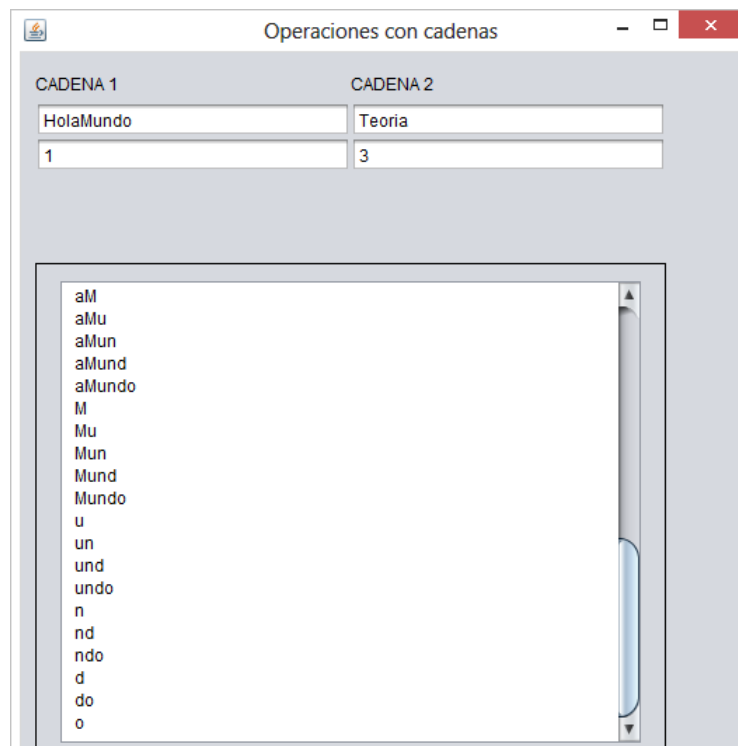
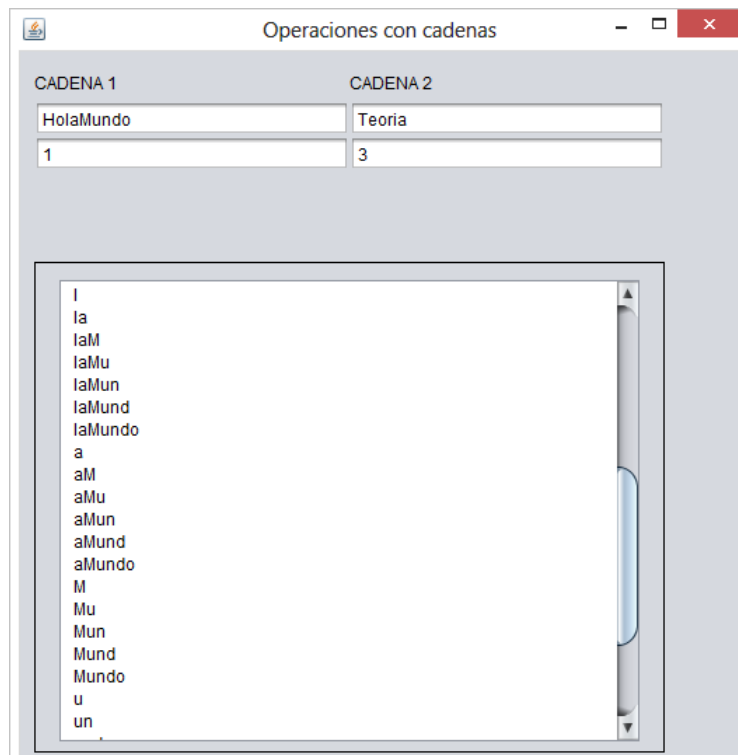
The screenshot shows a Java Swing window titled "Operaciones con cadenas". It contains two input fields labeled "CADENA 1" and "CADENA 2". "CADENA 1" contains the text "HolaMundo" and "2". "CADENA 2" contains the text "TeoriaComputacional" and "3". Below the input fields, there is a text area displaying a list of prefixes for the string "TeoriaComputacional". The prefixes are listed vertically, starting with "T" and ending with "TeoriaComputacional".

## ■ Sufijos



## ■ Subcadenas





- Determinar si es palíndromo

Operaciones con cadenas

CADENA 1 CADENA 2

Oso

1

LA CADENA ES PALINDROMO

Operaciones con cadenas

CADENA 1 CADENA 2

Anitalavalatina

1

LA CADENA ES PALINDROMO

Operaciones con cadenas

CADENA 1 CADENA 2

Hola

1

LA CADENA NO ES PALINDROMO

## 6. Conclusiones

Al terminar esta práctica pude manejar, diseñar, reafirmar e implementar en un lenguaje de programación las operaciones que se pueden hacer con las cadenas. Con respecto al lenguaje de programación, en mi caso personal jamás había usado estos métodos para trabajar cadenas, pero pude ver que es realmente sencillo y práctico usarlos, además de que facilitan mucho el trabajo.

Finalmente las operaciones fueron implementadas de manera correcta y en una interfaz amigable de tal forma que al usuario le fuera fácil trabajar con ellas.

## Referencias

- [1] (2017) Palíndromos. [Online]. Available: <http://www.letraslibres.com/sites/Convivio-zaid-esp.pdf>
- [2] E. A. Martinez, *Alfabetos, símbolos y cadenas*. [Online]. Available: <http://eafranco.com/docencia/teoriacomputacional/files/03/Clase03.pdf>