



**INSTITUTO POLITÉCNICO NACIONAL  
ESCUELA SUPERIOR DE CÓMPUTO**

**TEORÍA COMPUTACIONAL**

**2CM4**

**PROFESOR: LUZ MARÍA SÁNCHEZ GARCÍA**

**PRÁCTICA 2 EXPRESIONES REGULARES**

**VÁZQUEZ MORENO MARCOS OSWALDO**

**2016601777**



**FECHA DE ENTREGA: 16 DE MARZO DE 2018**

## INTRODUCCIÓN

En la siguiente práctica se pretende realizar un programa en lenguaje de programación Java, la cual valide expresiones regulares, en este caso la expresión es un *Hashtag*, en la cual se pretende que no pueda aceptar espacios, siendo rigorista en hacer uso forzosamente del “#” al inicio, seguido de cualquier combinación entre a-z (mayúsculas o minúsculas), 0-9 o “\_”.

Quedando de la siguiente manera la Expresión Regular `//#[a-zA-Z0-9]*`.

La implementación como el desarrollo de la idea es sencilla pero eficiente

## PLANTEAMIENTO DEL PROBLEMA

Implementar el algoritmo de codificación de un programa el cual acepte una cadena de caracteres de tipo *Hashtag* haciendo una correcta validación respecto a las siguientes restricciones:

- ✓ Iniciar siempre con el símbolo de almohadilla “#”.
- ✓ Usar cualquier letra del abecedario sin importar si son mayúsculas o minúsculas, además en cualquier combinación.
- ✓ Usar cualquier dígito del 0 al 9 en cualquier combinación.
- ✓ Es permitido más no obligatorio el uso del guion bajo “\_”.
- ✓ No es permitido usar cualquier otro tipo de caracteres que no sea guion bajo y almohadilla.
- ✓ No es permitido usar espacios dentro de la cadena.
- ✓ No es permitido usar acentos.
- ✓ La longitud de la cadena no importa.

## DISEÑO Y FUNCIONAMIENTO DE LA SOLUCIÓN

Siendo el autómata finito:



Una vez con la problemática planteada se necesitaba de poder darle sentido a la solución y es por eso por lo que se decidió crear un programa desarrollado en

lenguaje Java utilizando unas librerías y unas palabras reservadas capaz de admitir una expresión regular y llevarla a su validación.

Se llevará a cabo dentro del entorno gráfico de Java en un JFrame en el cual habrá una Etiqueta con las instrucciones, un JButton el cual hará la validación de que el Hashtag insertado en el campo de texto sea correcto, de ser valida la cadena arrojará un mensaje a otra etiqueta diciendo “VALIDA” y de lo contrario diciendo “NO VALIDA”.

## IMPLEMENTACIÓN DE LA SOLUCIÓN

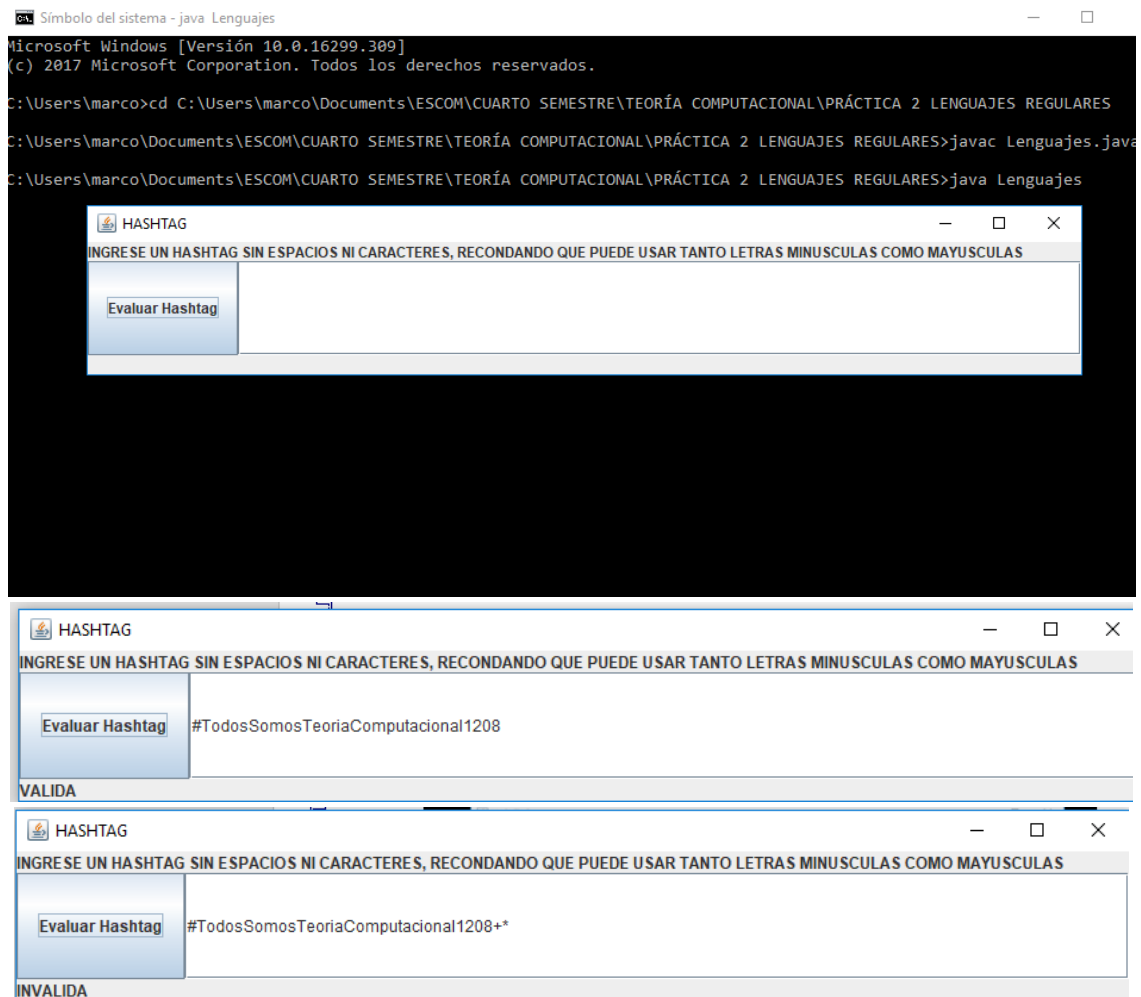
```
1  /* Hecho por Vázquez Moreno Marcos Oswaldo
2  Boleta 2016601777
3  El 16 de Marzo de 2018
4  Materia Teoría Computacional
5  Práctica número 2 EXPRESIONES REGULARES
6  Fecha de inicio 8 de Marzo
7  Fecha de terminación 8 de Marzo
8  */
9
10 //Librerías
11 import java.awt.event.*;
12 import javax.swing.*;
13 import java.awt.*;
14 import java.util.regex.*;
15 import java.util.regex.Pattern.*;
16 import java.util.regex.Matcher.*;
17
18 //Declaración de la clase pública Lenguajes que se mostrará en un JFrame con la implementación de una acción a algún botón
19 public class Lenguajes extends JFrame implements ActionListener{
20 //Declaración de la variables de tipo botón, Campo de escritura, Etiqueta, Contenedor
21 JButton evaluar;
22 JTextField campo;
23 JLabel instrucciones, resultado;
24 Container c;
25 //Constructor
26 public Lenguajes () {
27 //Declaración del contenedor
28
29 c = getContentPane ();
30 //creador del titulo
31 setTitle ("HASHTAG");
32 //texto dentro de la etiqueta y asignación
33 instrucciones = new JLabel ("INGRESE UN HASHTAG SIN ESPACIOS NI CARACTERES, RECONDANDO QUE PUEDE USAR TANTO
34 //Texto dentro del botón, agregandolo al Frame a la izquierda
35 evaluar = new JButton ("Evaluar Hashtag"); c.add("West", evaluar);
36 //Acción cuando se le de clic para evaluar
37 evaluar.addActionListener (this);
38 //Campo para insertar la cadena, en este caso el Hashtag
39 campo = new JTextField (5); c.add("Center", campo);
40 //Etiqueta en donde dirá si es valida o no
41 resultado = new JLabel (""); c.add("South", resultado);
42 setVisible(true); setSize(850, 150); setResizable(true);
43 }
44 //clase para que al hacer clic en el botón se pueda validar la expresión regular
45 public void actionPerformed ( ActionEvent e ){
46 JButton b=(JButton)e.getSource();
47 resultado.setIcon( b.getIcon() );
48 //palabra reservada de Java en la que puedes insertar expresiones regulares
49 //Aquí dentro de los paréntesis colocas tu E.R. siendo forzoso el # seguido de
50 //cualquier combinación de la a-z o A-Z o 0-9 o _
51 Pattern p = Pattern.compile ("#[a-zA-Z_0-9]*");
52 Matcher m = p.matcher (campo.getText());
53 //validador si es correcta
54 if (m.matches ()){
55 resultado.setText ("VALIDA");
56 }
```

```

43 //clase para que al hacer clic en el botón se pueda validar la expresión regular
44 public void actionPerformed ( ActionEvent e ){
45     JButton b=(JButton)e.getSource();
46     resultado.setIcon( b.getIcon() );
47     //palabra reservada de Java en la que puedes insertar expresiones regulares
48     //Aquí dentro de los paréntesis colocas tu E.R. siendo forzoso el # seguido de
49     //cualquier combinación de la a-z o A-Z o 0-9 o _
50     Pattern p = Pattern.compile ("#[a-zA-Z_0-9]*");
51     Matcher m = p.matcher (campo.getText());
52     //validador si es correcta
53     if (m.matches ()){
54         resultado.setText ("VALIDA");
55     }
56     else { //caso contrario
57         resultado.setText ("INVALIDA");
58     }
59     setDefaultCloseOperation(JFrame.EXIT_ON_CLOSE);
60 }
61 public static void main (String s[]){
62     Lenguajes ex = new Lenguajes ();
63 }
64 //#[a-zA-Z_0-9]*
65 }

```

## FUNCIONAMIENTO



## CONCLUSIONES

En conclusión, la práctica número 2 me agradó bastante, además me ayudó más que nada a practicar de nuevo mi destreza en la programación ahora en un lenguaje nuevo de programación para mí el cual es Java y tengo poco de usarlo

Por otro lado, es una práctica bastante útil ya que la finalidad es comprender cómo funcionan las expresiones regulares y su correcta implementación, se cree que no se necesita el problema más difícil para comprender lo que se ve en la clase, lo cual ha sido bastante bueno y este tipo de códigos e implementación retroalimentan de una muy buena forma lo que aprendemos en el salón de clases, con una excelente profesora.

Sin duda estoy contento por lo que se ha logrado en esta práctica ya que lo que me tocó fue divertido para mí, así como espero que haya sido para todos mis compañeros, el uso de un nuevo Software como JFLAP facilita sin duda el diseño de los autómatas, por lo que estoy agradecido con la profesora por proporcionarnos las herramientas para una mejor implementación.

## BIBLIOGRAFÍA

Software de aplicación JFLAP: <http://www.jflap.org/>

Copyright © 1993, 2017, Oracle and/or its affiliates. All rights reserved. Use is subject to license terms.

<https://docs.oracle.com/javase/7/docs/api/java/util/regex/Pattern.html>