

**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

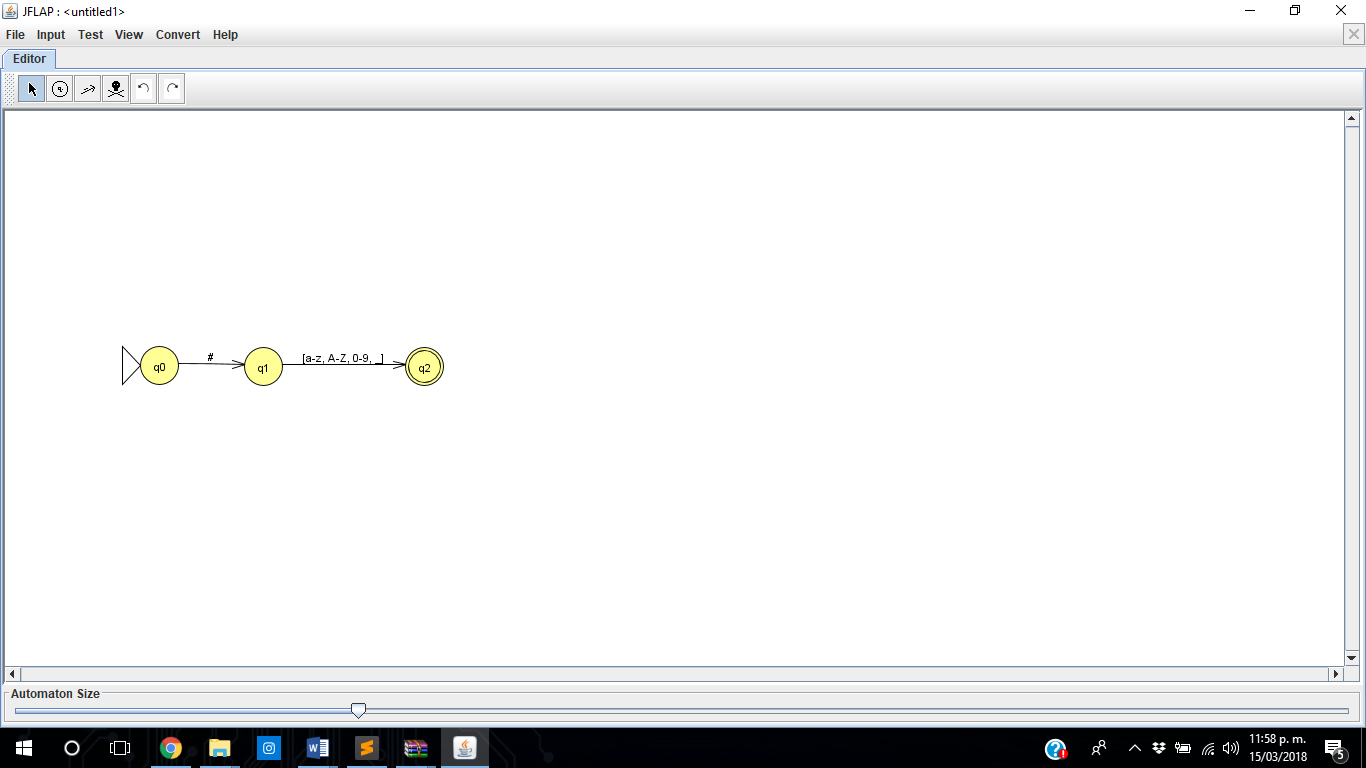
**PLANTEAMIENTO DEL PROBLEMA**

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**

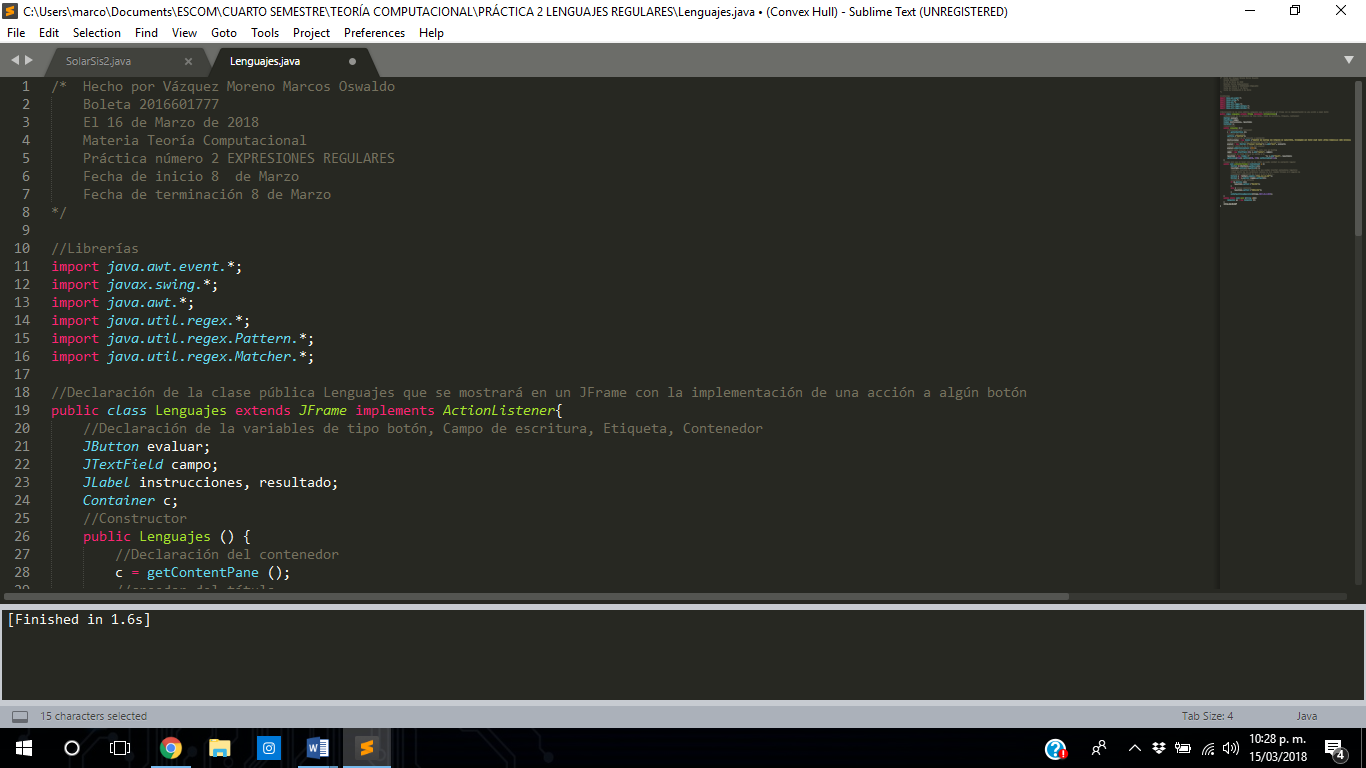
Siendo el autómata finito:

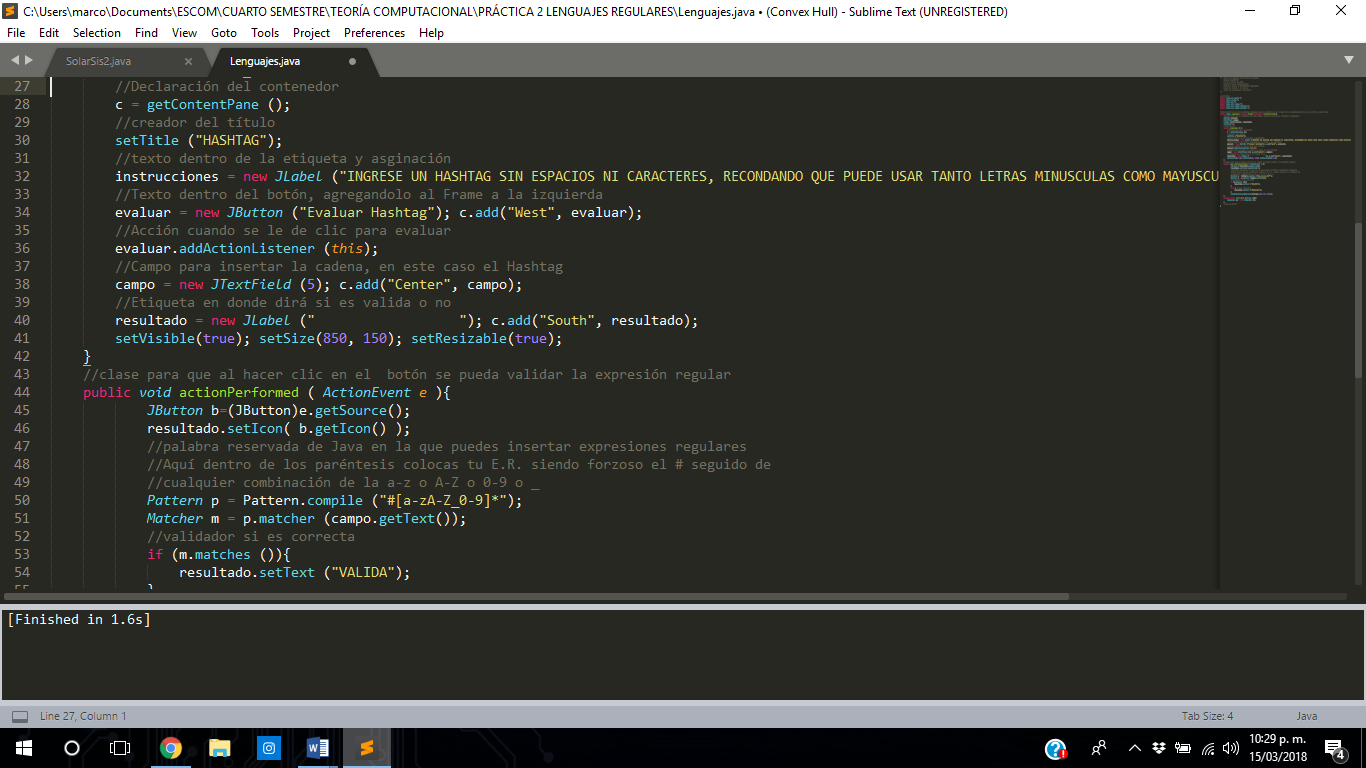


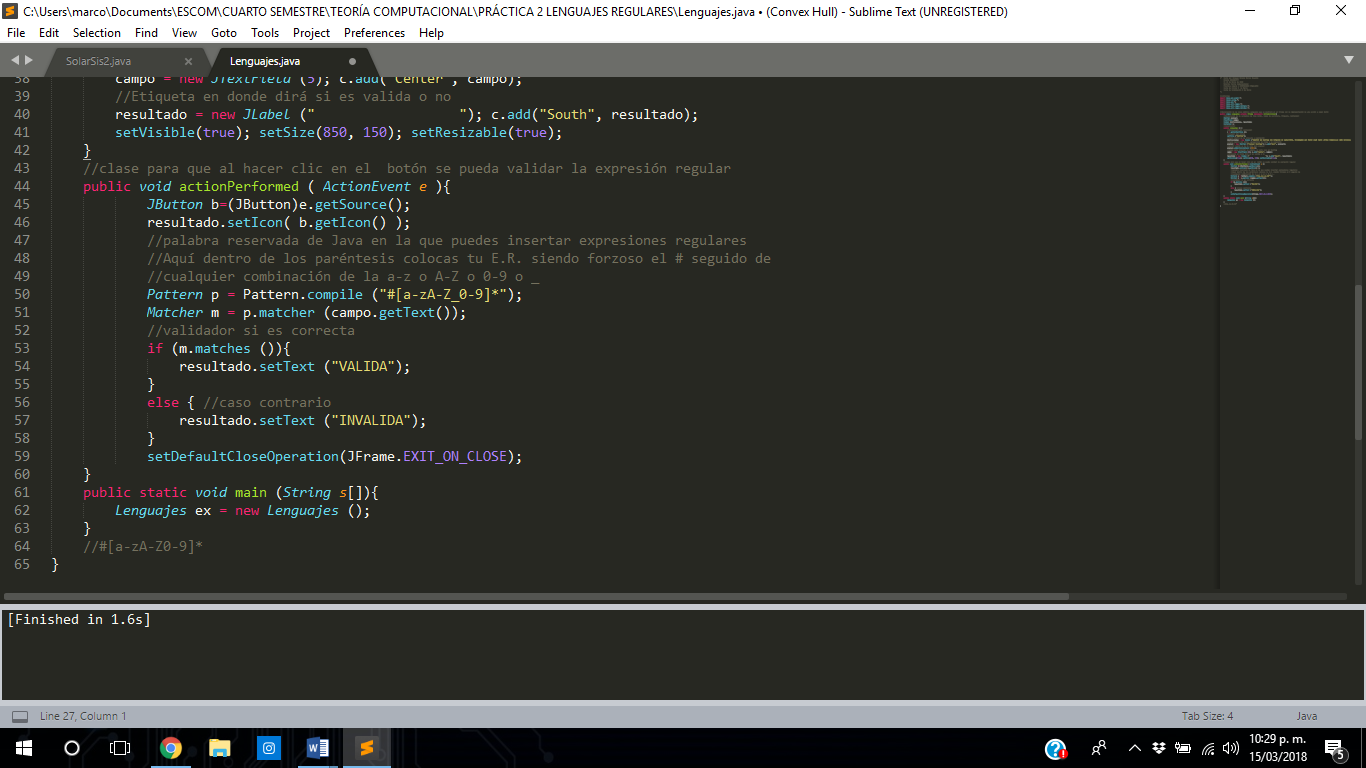
Una vez con la problemática planteada se necesitaba de poder darle sentido a la solución y es por eso por lo que se decidió crear un programa desarrollado en lenguaje Java utilizando unas librerías y unas palabras reservadas capaz de admitir una expresión regular y llevarla a su validación.

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”.

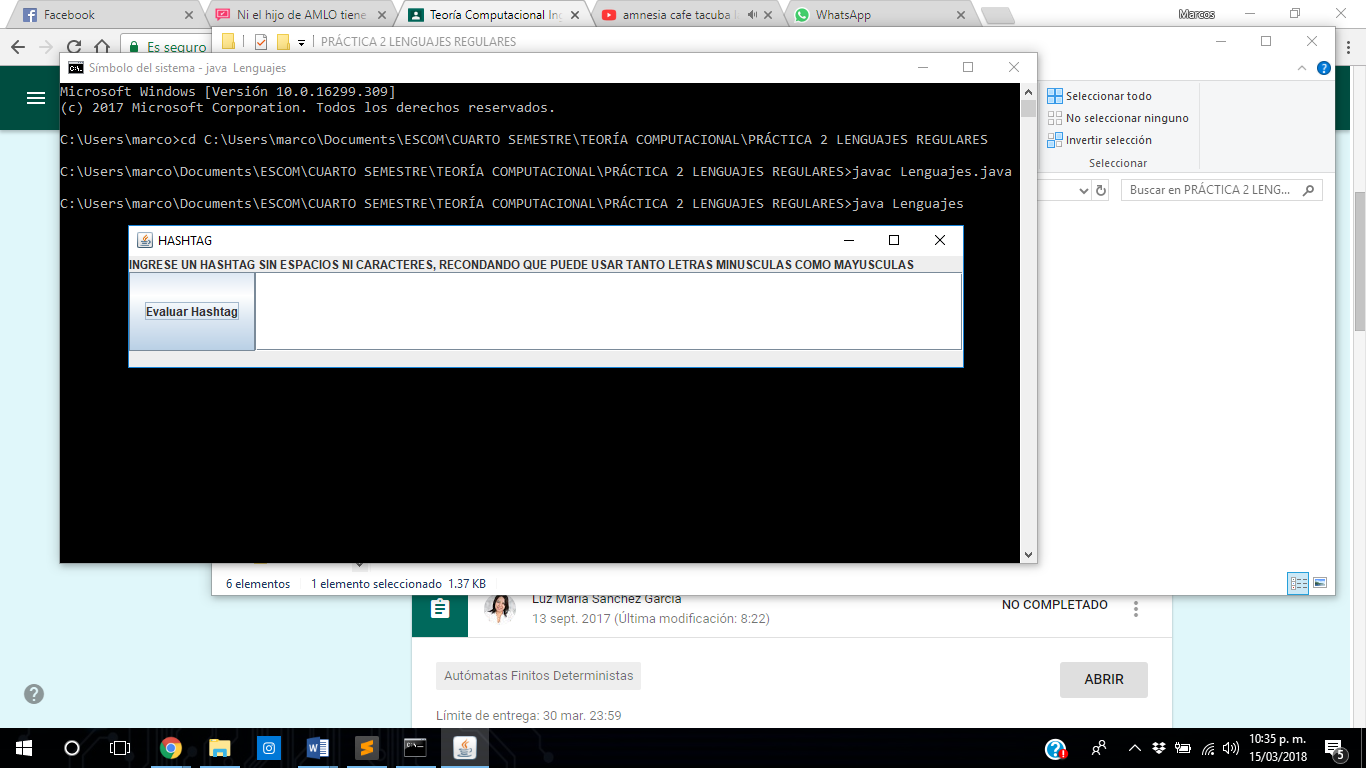
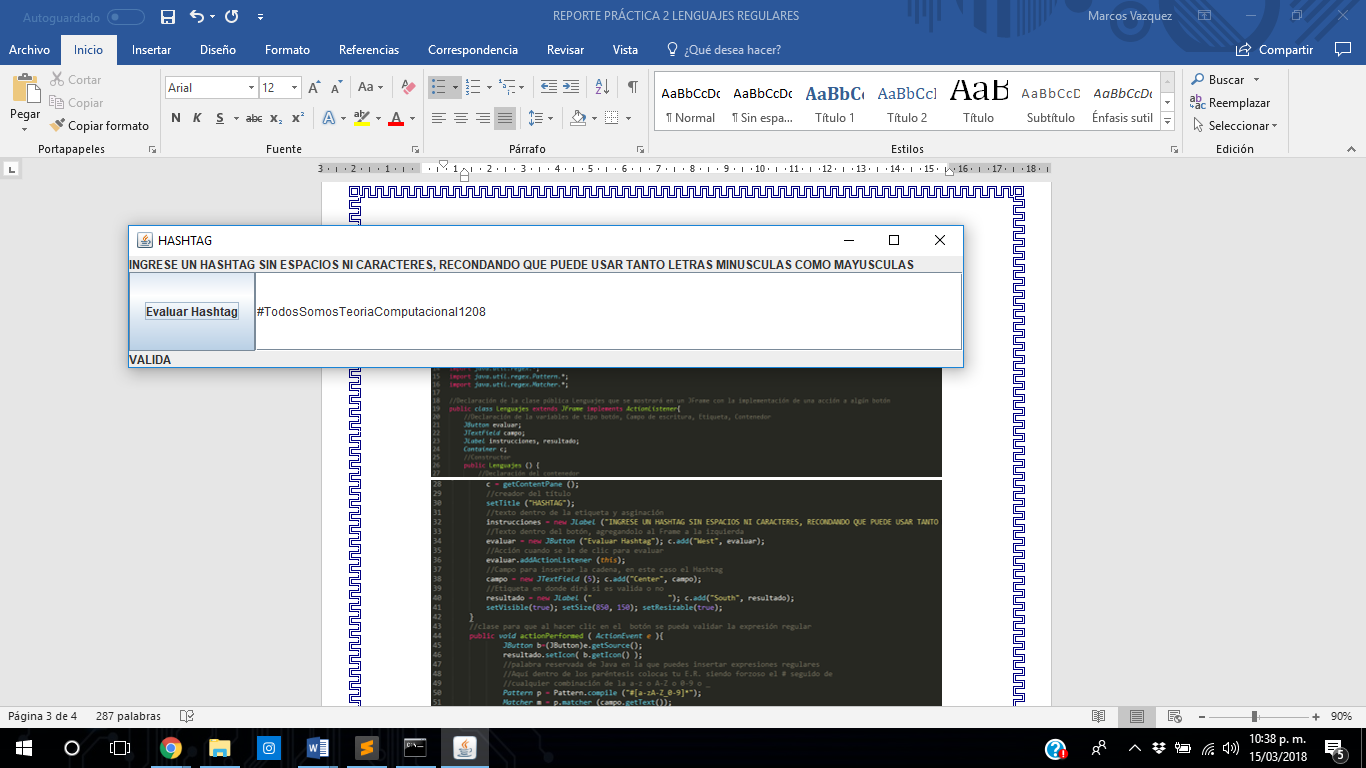
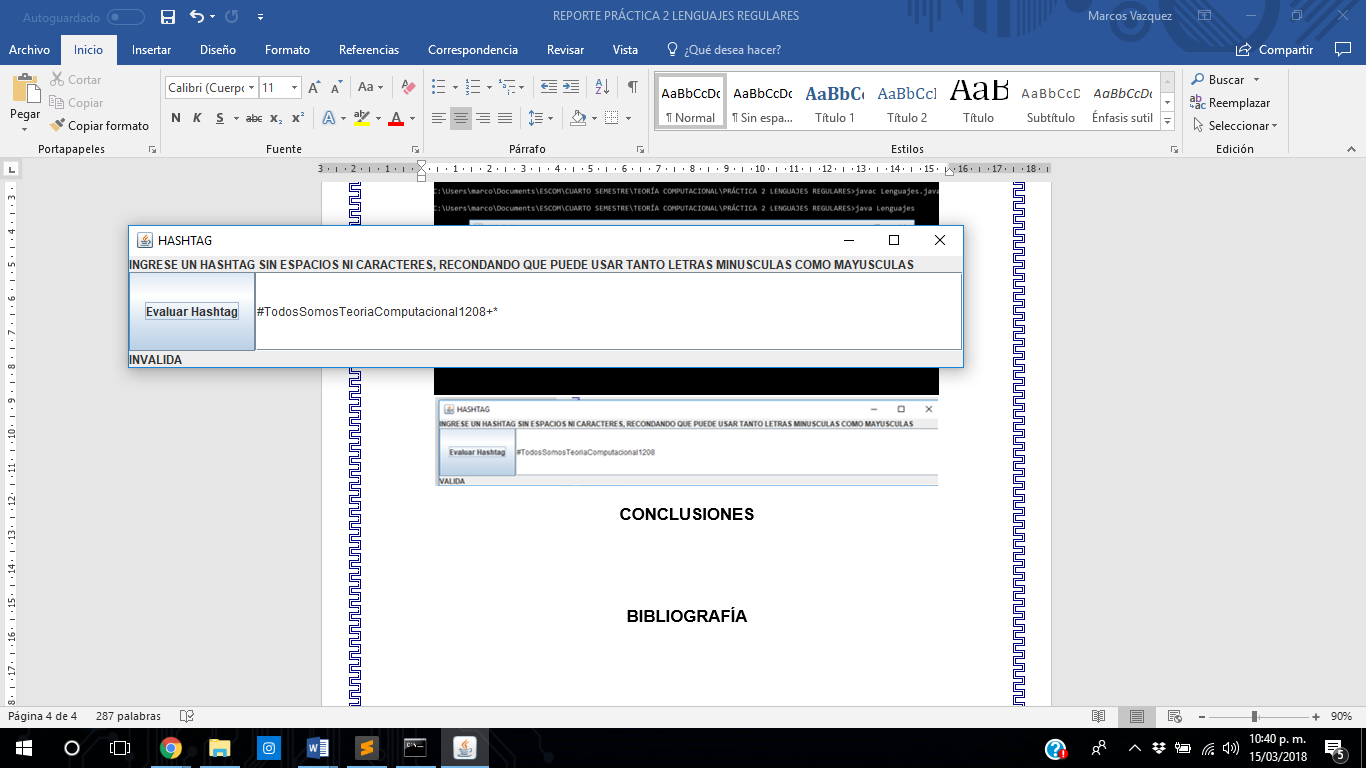
**IMPLEMENTACIÓN DE LA SOLUCIÓN**







**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](https://docs.oracle.com/javase/7/docs/legal/cpyr.html) © 1993, 2017, Oracle and/or its affiliates. All rights reserved. Use is subject to [license terms](http://download.oracle.com/otndocs/jcp/java_se-7-mrel-spec/license.html).

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