

Teoría de la computación - Práctica 2

Grupo Lunes 8:00-10:00 semanas B

— Práctica 2 —

Autor: Fabrizio Duarte

NIP:736857

Autor: Fernando Pastor Peralta

NIP:897113

Ejercicio 1:

1. Resumen:

El objetivo es implementar un analizador léxico, que dado un fichero en HTML, se muestren las URL's del atributo href de las etiquetas <a> (una URL por línea) y el total de cabeceras que contenga el fichero:

Todas las condiciones de arranque serán exclusivas.

En cuanto al total de cabeceras, declaramos contadorH, variable dedicada a contar la cantidad de etiquetas de cabecera que hay en el texto. El contador se incrementará según una regla, en la que se comprueba que el inicio de la etiqueta <h> es válido (< junto a h o H y un número del 0 al 6), es resto de la instrucción se omite con la condición de arranque CODIGO H. Al terminar el escaner del texto, se imprimirá por pantalla contadorH.

En INITIAL, cuando el analizador se encuentra con el símbolo "<a" se interpreta como el comienzo de una etiqueta HTML "<a>" y cambia al estado BUSCARHREF.

Para imprimir las URLs, definimos dos condiciones de arranque, para buscar y procesar href:

-BUSCARHREF, donde se busca href (en cualquiera combinación de mayúsculas o minúsculas seguida de un símbolo "=" y comillas dobles) y una vez encontrado llama a HREFENCONTRADO.

-HREFENCONTRADO, dentro de este, se obtiene la URL del atributo href y se imprime en el texto de salida.

Cada vez que se encuentre un comentario, su contenido será omitido mediante la condición de arranque COMENTARIO.

Destacamos que durante todas las secciones de código eliminamos los caracteres de entrada que no usa ninguna regla para que no sean imprimidos en el texto de salida.

2.Pruebas:

ENTRADA:

```
1 <html> <body>
2 <h3 style = "color : red">Tema 1</h3>
3 <p>Empezamos el tema 1 con un enlace </p>
4 <a lang = "es"
5 Href = "http://www.unizar.es">
6 http://algo.com </A>
7 <!-- <a Href = "http://www.uniclo.es">
8 compra </a>
9 -->
10 Es necesario un
11 <abbr title="Personal Computer">PC </abbr>
12 para resolver los problemas.
13
14 <a href="http://s.com">http://s.com</a>
15 <a href="http://www.unizar.es">La uni</a>
16
17 <H2 title = "<a>"> Ejemplo de cabecera que contiene " </h2>
18
19 </body></html>
20
```

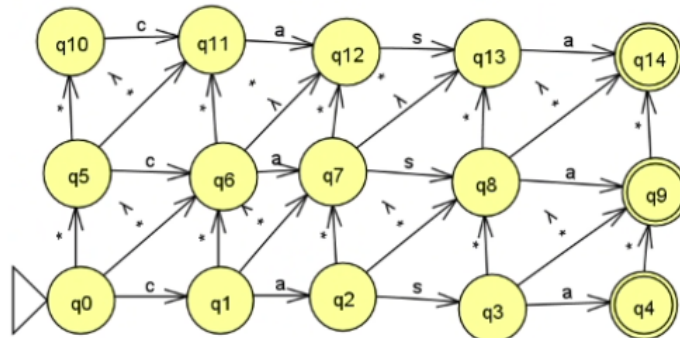
SALIDA:

```
http://www.unizar.es
http://s.com
http://www.unizar.es
H: 2
hendrix02:~/ProgramasP2-TComp/ej1_p2_Tcomp/ pwd
/home/a897113/ProgramasP2-TComp/ej1_p2_Tcomp
```

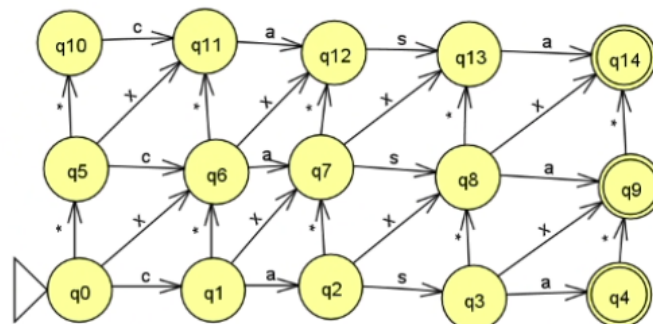
Ejercicio 2:

1.Resumen:

Para la resolución de este ejercicio hemos creado el autómata de Levenshtein:



Debido a que la expresión regular de este autómata es complicada de traducir en Flex, hemos optado por poner una letra aleatoria (x) para simplificar el proceso y conseguir una expresión regular más sencilla.



Posteriormente modificaremos esta expresión con Visual Studio Code (haciendo uso de la herramienta de seleccionar y cambiar los mismos caracteres a la vez) para que sea una expresión regular comprensible para Flex. Los Cambio de la expresión regular que nos da JFLAP son: los "+" como "|", los "*" como "." y las "x" (letra aleatoria) como "?".

Expresión regular aportada por JFLAP (no x):

```
casa+cas(*+λ)+casa*+(ca(*+λ)+cas*+(c(*+λ)+ca*+(*+λ+c*+*c)a)s)a+(ca(*+λ)+cas*+(c(*+λ)+ca*+(*+λ+c*+*c)a)s)(*+λ)+(c
as(*+λ)+casa*+(ca(*+λ)+cas*+(c(*+λ)+ca*+(*+λ+c*+*c)a)s)a
)*+((c(*+λ)+ca*+(*+λ+c*+*c)a)(λ+*)+(ca(*+λ)+cas*+(c(*+λ)+c
a*+(*+λ+c*+*c)a)s)*+(((*+λ+c*+*c)(λ+*)+(c(*+λ)+ca*+(*+λ+c*
+*c)a)*+(*+λ)+(*+λ+c*+*c)*+*c)a)s)a
```

Expresión regular aportada por JFLAP (con x):

`casa+casx+casa*+(cax+cas*+(cx+ca*+(x+c*+*c)a)s)a+(cax+cas*+(cx+ca*+(x+c*+*c)a)s)x+(casx+casa*+(cax+cas*+(cx+ca*+(x+c*+*c)a)s)a)*+((cx+ca*+(x+c*+*c)a)x+(cax+cas*+(cx+ca*+(x+c*+*c)a)s)*+((x+c*+*c)x+(cx+ca*+(x+c*+*c)a)*+(*x+(x+c*+*c)*+*c)a)s)a`

Expresión regular modificada para uso en Flex:

`^(casa|cas.|(ca.|cas.|(c.|ca.|(.|c.|c)a)s)a|(ca.|cas.|(c.|ca.|(.|c.|c)a)s).|(cas.|(ca.|cas.|(c.|ca.|(.|c.|c)a)s)a).|((c.|ca.|(.|c.|c)a).|(ca.|cas.|(c.|ca.|(.|c.|c)a)s).|((.|c.|c).|(c.|ca.|(.|c.|c)a).|(..|(.|c.|c).|c)a)s)a)$`

Los caracteres que no son utilizados por la regla, son eliminados.

Finalmente, cuando una palabra es aceptada por el autómata, la imprime por el texto de salida.

2.Pruebas:

ENTRADA:

```
1 casas
2 costa
3 costas
4 asa
5 asado
6 cosa
7 posa
8 rusa
9 chocolate
10 cals
11 cssa
12 csa
13 sa
14 sula
15
```

SALIDA:

```
casas
costa
asa
cosa
posa
rusa
cals
cssa
csa
sa
hendrix02:~/ProgramasP2-TComp/ej2_p2_Tcomp/ pwd
/home/a897113/ProgramasP2-TComp/ej2_p2_Tcomp
```