

Universidad Mariano Gálvez de Guatemala

Facultad de Ingeniería en Sistemas de Información

Licenciatura en Ingeniería en Sistemas de Información y Ciencias de la Computación

Autómatas y Lenguajes Formales

Ing. Nery Antonio Alvizures Melendez

Sección B

Proyecto del Curso

Javier Alejandro Diaz Portillo

9490-19-2647

30 de octubre 2021

Índice

Introducción	3
Problema del Proyecto	4
Código en Java	7
Imágenes de los resultados de Programa	10
Conclusiones	16
Aplicaciones de la Vida Real	16
Bibliografía	16

Introducción

Estaremos viendo sobre el tema de Maquina Turing, tenemos un ejercicio el cual vamos a desarrollar el procedimiento de solucionarlo tal como lo haría una maquina Turing. Luego veremos el código de una MT y probaremos el mismo ejercicio para ver si la maquina nos da el mismo resultado como lo haría una persona.

Esta es nuestro ejercicio para mostrar el tema

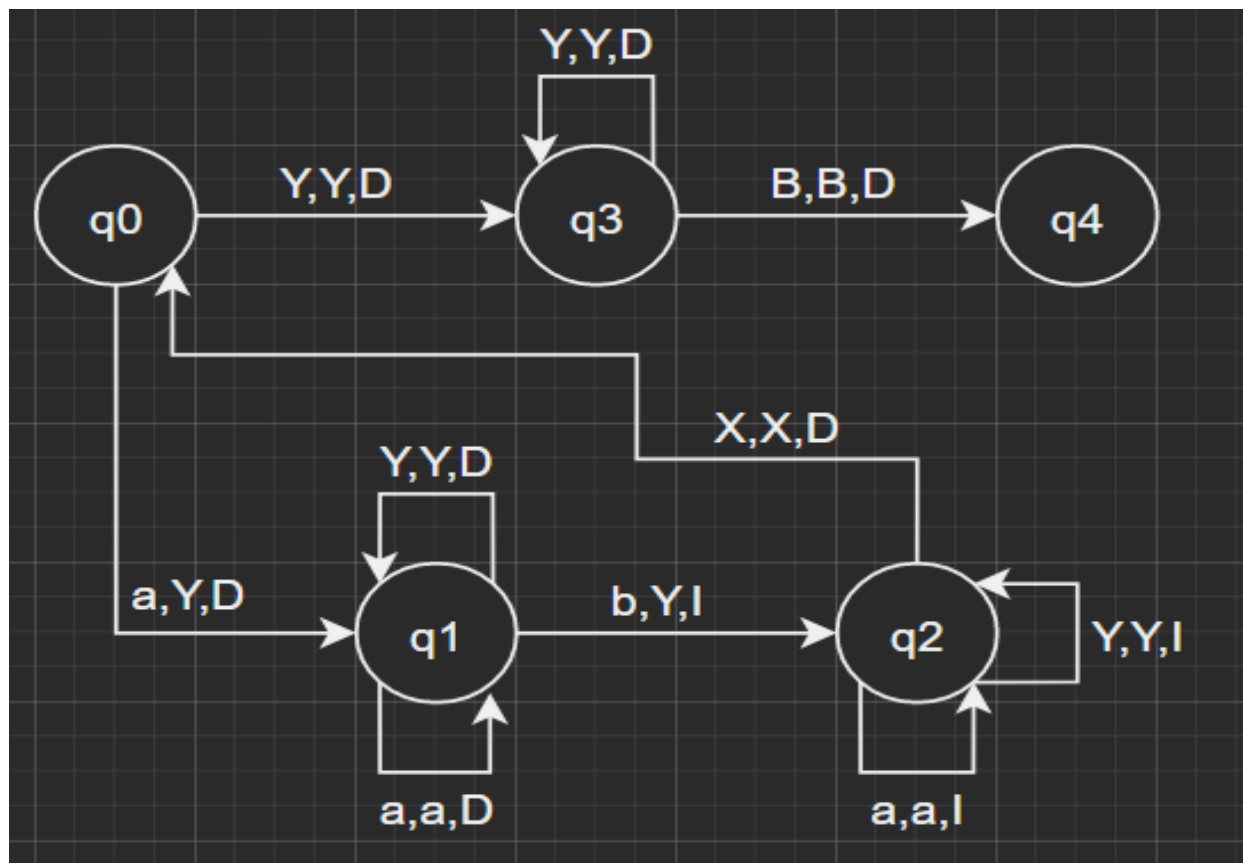
Diseñar una máquina de Turing que acepte el lenguaje $L = \{a^n b^n / n \geq 1\}$.

Que permita el ingreso de cualquier sentencia o cadena de caracteres e indique si es una cadena valida o no

Vamos a utilizar como ejemplos el uso de la cadena ab y aabb, ya que el ejercicio requiere que acepte $a^n b^n$, es decir, la misma cantidad de a como b. Si ponemos abb debe salir que la cadena **no** fue aceptada, entonces nos sale esta tabla de movimientos:

δ	a	b	X	Y	B
q0	q1,X,D	-	-	q3,Y,D	-
q1	q1,a,D	q2,Y,I	-	q1,Y,D	-
q2	q2,a,I	-	q0,X,D	q2,Y,I	-
q3	-	-	-	q3,Y,D	q4,B,D
q4	S	S	S	S	S

Nuestro Diagrama de Moore sería el siguiente:



Iteracion	Cadena	ab	Estado Actual		Nuevo estado		
0	a	b	B	q0		q1	
1	X	b	B	q1		q2	
2	X	Y	B	q2		q0	
3	X	Y	B	q0		q3	
4	X	Y	B	q3		q4	
5	S			q4			

Codigo en Java

```
package proyecto;
```

```
import java.io.BufferedReader;
import java.io.InputStreamReader;
import java.util.Arrays;
import java.util.Collections;
```

```
public class maquinaTuring {
    public static void main(String[] args) throws Exception {
        BufferedReader ingreso = new BufferedReader(new InputStreamReader(System.in));

        int estados;
        String cadenachar;
        char[] caracter;
        char[] car;
        String qInicial, q, qFinal;

        estados = 5; //Numero de estados
        qInicial = "q0"; //Estado Inicial
        q = qInicial; //Estado Actual
        cadenachar = "abXYB"; //Los caracteres a utilizar
        caracter = cadenachar.toCharArray(); //caracteres a utilizar, convertido en un array
        qFinal = "q4"; //Estado Final

        String Q[] = {"q0", "q1", "q2", "q3", "q4"}; //Nombre de nuestros estados
        //Movimientos en cada estado,posicion
        String estado[][] = {
            //          a      b      X      Y      B
            /*q0*/ { "q1,X,D", " - ", " - ", "q3,Y,D", " - " },
            /*q1*/ { "q1,a,D", "q2,Y,I", " - ", "q1,Y,D", " - " },
            /*q2*/ { "q2,a,I", " - ", "q0,X,D", "q2,Y,I", " - " },
            /*q3*/ { " - ", " - ", " - ", "q3,Y,D", "q4,B,D" },
            /*q4*/ { " S ", " S ", " S ", " S ", " S " };

        //Nos indica que contiene la máquina de Turing
        System.out.println("Q={ " + Arrays.toString(Q).replace("[", "").replace("]", "") + " }");
        System.out.println("Σ={ " + caracter[0] + ", " + caracter[1] + " }");
        System.out.println("Γ={ " + Arrays.toString(caracter).replace("[", "").replace("]", "") +
        " }");
        System.out.println("F={ " + qFinal + " } Estado Inicial: " + qInicial);
```

```

//Imprime nuestra tabla de funciones
String repeat = "----+---";
System.out.println("La función  $\delta$  viene dada por la tabla siguiente:\n  $\delta$ " +
Arrays.toString(caracter).replace("[", " | ").replace(",", " | ").replace("]", " |"));
for (int i = 0; i < estados; i++) {
    System.out.println(String.join("", Collections.nCopies(caracter.length + 1, repeat)));
    System.out.println(Q[i] + Arrays.toString(estados[i]).replace("[", " | ").replace(" ", " |").replace("]", " |"));
}
System.out.println(String.join("", Collections.nCopies(caracter.length + 1, repeat)));

String cadena;
System.out.println("Ingrese una cadena:");
cadena = "B" + ingreso.readLine() + "B"; //Cadena que ingreso el usuario, se agrega "B"
por su significado: en blanco
car = cadena.toCharArray(); //cadena convertido a Array

int i = 1;
boolean izq = false, der = false, stop = false, aceptado = false; //Condiciones para mover
izquierda o derecha, también si se debe detener o no y por ultimo si la cadena es aceptada
while (!stop) { //Si es falso, el ciclo no se detiene y repetirá el resultado
    if (i < car.length) {
        System.out.println("-----Caracter Asignado " + car[i] + "-----");
        //Se puede decir que el carácter asignado es la cabeza de la cinta
        for (int j = 0; j < estados; j++) {
            if (q.equals(Q[j])) {
                for (int k = 0; k < caracter.length; k++) {
                    if (car[i] == caracter[k] && !estado[j][k].contains("-")) {
                        //Si el carácter de la cadena es igual a una de los caracteres de nuestro programa, y que su celda
                        no tiene "-", entonces seguimos esta parte del código
                        System.out.println("Inicial:" + q);
                        String[] instr = estado[j][k].split(",");
                        System.out.println(estado[j][k]);
                        System.out.println(""" + car[i] + "", tiene el estado: " + instr[0] + ",
reemplazaremos el valor:" + instr[1] + ", y su Direccion:" + instr[2]);
                        q = instr[0];
                        car[i] = instr[1].charAt(0);
                        if (instr[2].equals("D")) {
                            der = true;
                            izq = false;

```



```

        } else if (instr[2].equals("I")) {
            der = false;
            izq = true;
        }
        break;
    } else if (car[i] == caracter[k] && estado[j][k].contains("-")) {
//Si la celda si contiene el signo "-", entonces correra este código que nos indica que es un
"estado imposible" por ende, termina el programa y ponemos que la cadena no se acepto
        System.out.println("Inicial:" + q);
        System.out.println(estado[j][k]);
        stop = true;
        aceptado = false;
        break;
    }
}
break;
}
}

if (der) {
    i++;
}
if (izq) {
    i--;
}
} else {
    if (qFinal.equals(q)) {
        //Si la iteración se pasa de nuestra cadena, entonces aquí verificamos si nuestro
estado actual es igual al estado final, si es asi, entonces nuestra maquina aceptara la cadena del
usuario y detiene la cinta
        System.out.println("-----Estado Final " + q + "-----");
        System.out.println(q + estado[estados - 1][caracter.length - 1]);
        stop = true;
        aceptado = true;
        break;
    }
    else{
        stop = true;
        aceptado = false;
        break;
    }
}
}
System.out.println(Arrays.toString(car).replace("[", " ").replace(", ", "").replace("]", ""));

```

```
    }  
  
    if(aceptado){  
        System.out.println("La cadena fue aceptada");  
    }  
    else{  
        System.out.println("La cadena no fue aceptada");  
    }  
  
    }  
}
```

Resultado del programa:

Usando las cadenas: abab, aabb, ab

$Q = \{q_0, q_1, q_2, q_3, q_4\}$

$\Sigma = \{a, b\}$

$\Gamma = \{a, b, X, Y, B\}$

$F = \{q_4\}$ Estado Inicial: q_0

La función δ viene dada por la tabla siguiente:

δ	a	b	X	Y	B
q_0	$ q_1, X, D $	-	-	$ q_3, Y, D $	-
q_1	$ q_1, a, D q_2, Y, I $	-	$ q_1, Y, D $	-	
q_2	$ q_2, a, I $	-	$ q_0, X, D q_2, Y, I $	-	
q_3	-	-	-	$ q_3, Y, D q_4, B, D $	
q_4	S	S	S	S	S

Ingrese una cadena:

ab

```
-----Caracter Asignado a-----  
Inicial:q0  
q1,X,D  
'a', tiene el estado: q1, reemplazaremos el valor:X, y su Direccion:D  
  BXbB  
-----Caracter Asignado b-----  
Inicial:q1  
q2,Y,I  
'b', tiene el estado: q2, reemplazaremos el valor:Y, y su Direccion:I  
  BXYB  
-----Caracter Asignado X-----  
Inicial:q2  
q0,X,D  
'X', tiene el estado: q0, reemplazaremos el valor:X, y su Direccion:D  
  BXYB  
-----Caracter Asignado Y-----  
Inicial:q0  
q3,Y,D  
'Y', tiene el estado: q3, reemplazaremos el valor:Y, y su Direccion:D  
  BXYB  
-----Caracter Asignado B-----  
Inicial:q3  
q4,B,D  
'B', tiene el estado: q4, reemplazaremos el valor:B, y su Direccion:D  
  BXYB  
-----Estado Final q4-----  
q4    S  
La cadena fue aceptada  
BUILD SUCCESSFUL (total time: 25 seconds)
```

$Q = \{q_0, q_1, q_2, q_3, q_4\}$

$\Sigma = \{a, b\}$

$\Gamma = \{a, b, X, Y, B\}$

$F = \{q_4\}$ Estado Inicial: q_0

La función δ viene dada por la tabla siguiente:

δ	a	b	X	Y	B
q_0	$ q_1, X, D $	-	-	$ q_3, Y, D $	-
q_1	$ q_1, a, D q_2, Y, I $	-	$ q_1, Y, D $	-	
q_2	$ q_2, a, I $	-	$ q_0, X, D q_2, Y, I $	-	
q_3	-	-	-	$ q_3, Y, D q_4, B, D $	
q_4	S	S	S	S	S

Ingrese una cadena:

aabb

```
-----Caracter Asignado a-----  
Inicial:q0  
q1,X,D  
'a', tiene el estado: q1, reemplazaremos el valor:X, y su Direccion:D  
BXabbB  
-----Caracter Asignado a-----  
Inicial:q1  
q1,a,D  
'a', tiene el estado: q1, reemplazaremos el valor:a, y su Direccion:D  
BXabbB  
-----Caracter Asignado b-----  
Inicial:q1  
q2,Y,I  
'b', tiene el estado: q2, reemplazaremos el valor:Y, y su Direccion:I  
BXaYbB  
-----Caracter Asignado a-----  
Inicial:q2  
q2,a,I  
'a', tiene el estado: q2, reemplazaremos el valor:a, y su Direccion:I  
BXaYbB  
-----Caracter Asignado X-----  
Inicial:q2  
q0,X,D  
'X', tiene el estado: q0, reemplazaremos el valor:X, y su Direccion:D  
BXaYbB  
-----Caracter Asignado a-----  
Inicial:q0  
q1,X,D  
'a', tiene el estado: q1, reemplazaremos el valor:X, y su Direccion:D  
BXXYbB
```

```
-----Caracter Asignado Y-----
Inicial:q1
q1,Y,D
'Y', tiene el estado: q1, reemplazaremos el valor:Y, y su Direccion:D
  BXXYbB
-----Caracter Asignado b-----
Inicial:q1
q2,Y,I
'b', tiene el estado: q2, reemplazaremos el valor:Y, y su Direccion:I
  BXXYYB
-----Caracter Asignado Y-----
Inicial:q2
q2,Y,I
'Y', tiene el estado: q2, reemplazaremos el valor:Y, y su Direccion:I
  BXXYYB
-----Caracter Asignado X-----
Inicial:q2
q0,X,D
'X', tiene el estado: q0, reemplazaremos el valor:X, y su Direccion:D
  BXXYYB
-----Caracter Asignado Y-----
Inicial:q0
q3,Y,D
'Y', tiene el estado: q3, reemplazaremos el valor:Y, y su Direccion:D
  BXXYYB
-----Caracter Asignado Y-----
Inicial:q3
q3,Y,D
'Y', tiene el estado: q3, reemplazaremos el valor:Y, y su Direccion:D
  BXXYYB
-----Caracter Asignado B-----
Inicial:q3
q4,B,D
'B', tiene el estado: q4, reemplazaremos el valor:B, y su Direccion:D
```

-----Estado Final q4-----

q4 S

La cadena fue aceptada

Si no se ingresa la cadena de una forma correcta, o si no cumple las condiciones del lenguaje, la maquina Turing nos dira que hay un error

```
Ingresa una cadena:
abab
-----Caracter Asignado a-----
Inicial:q0
q1,X,D
'a', tiene el estado: q1, reemplazaremos el valor:X, y su Direccion:D
BXbabB
-----Caracter Asignado b-----
Inicial:q1
q2,Y,I
'b', tiene el estado: q2, reemplazaremos el valor:Y, y su Direccion:I
BXYabB
-----Caracter Asignado X-----
Inicial:q2
q0,X,D
'X', tiene el estado: q0, reemplazaremos el valor:X, y su Direccion:D
BXYabB
-----Caracter Asignado Y-----
Inicial:q0
q3,Y,D
'Y', tiene el estado: q3, reemplazaremos el valor:Y, y su Direccion:D
BXYabB
-----Caracter Asignado a-----
Inicial:q3
-
BXYabB
La cadena no fue aceptada
```


Conclusiones

Aquí tenemos el desarrollo de nuestra máquina Turing, hemos ingresado varios ejemplos en el programa para demostrar su función, su proceso, las iteraciones, que al final, nos dice si nos lo acepta o no

Aplicación a la Vida Real

Las máquinas Turing pueden ser utilizadas en varias áreas de nuestro día a día, por ejemplo, se puede utilizar como una Máquina Oráculo, que es una máquina capaz de contestar preguntas sobre la pertenencia a un conjunto específico de números naturales, generadores de lenguaje, calculadora de funciones, etc.

Bibliografías

Programa para hacer la Diagrama de Moore <https://app.diagrams.net/>

Aplicaciones <http://maquinasdeturing.blogspot.com/2010/08/9-ejemplos-de-aplicacion-de-las-mt.html>