

MANUAL DE USUARIO

Se crea un menú

```
SELECCIONA UNA OPCION1
t1 - CARGAR ARCHIVO
t2 - INFORMACION GRAMATICA
t3 - GENERAR AUTOMATA
t4 - REPORTE DE RECORRIDO
t5 - REPORTE TABLA
t6 - SALIR
```

T1: cargar el archivo .glc con sus gramáticas y además muestra que gramática fue descartada por ser regular.

```
INSERTE UNA OPCION >>1
ES UNA EXPRESION REGULAR: Gramatica2
```

T2: da previa información del autómata: se selecciona alguna gramática para evaluarla: Se espera unos 4seg para confirmar si quiere salir o no.

```
Nombre de la gramática Holis
No terminales = {S,A,B,C,E}
terminales = {0,1,2,3}
terminales inicial= S
S->A
A->0
    |0 B 0
    |0 C 0
B->1
    |1 D 1
C->2
    |2 E 2
E->3

Tiempo de espera
esperando_____
QUIERE SALIR: si,no >>no
esperando_____
QUIERE SALIR: si,no >>si
```

T3: Genera un autómata mostrada en html con sus nodos

Nombre: Holis

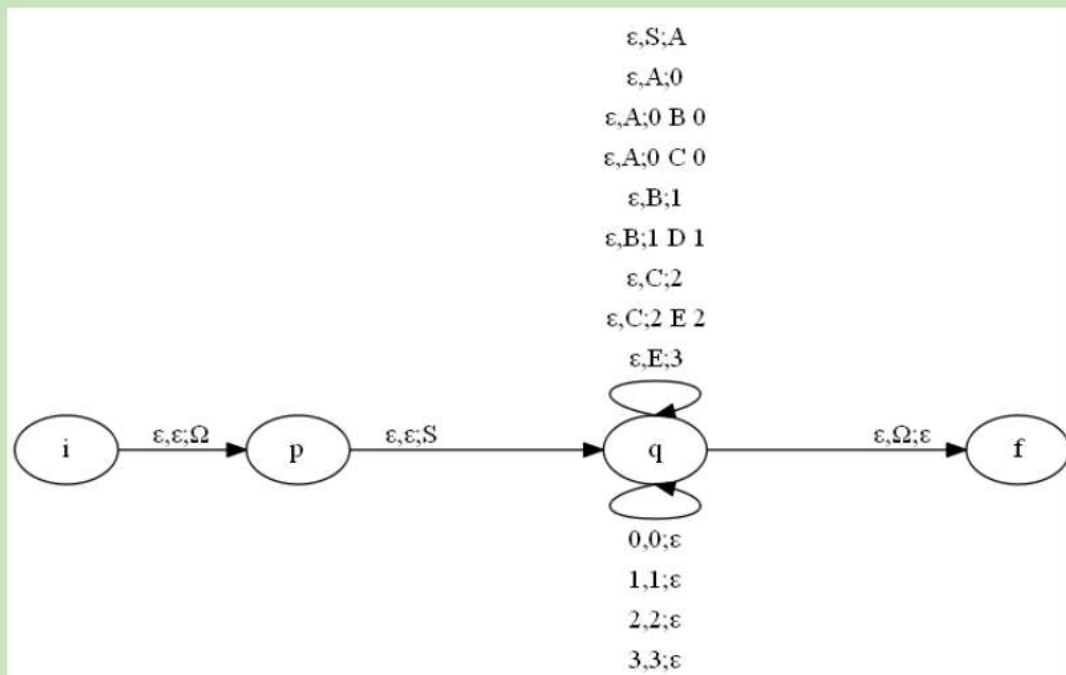
Terminales={0,1,2,3}

Alfabeto de pila={0,1,2,3,S,A,B,C,E, Ω }

Estados={i,p,q,f}

Estado inicial={S}

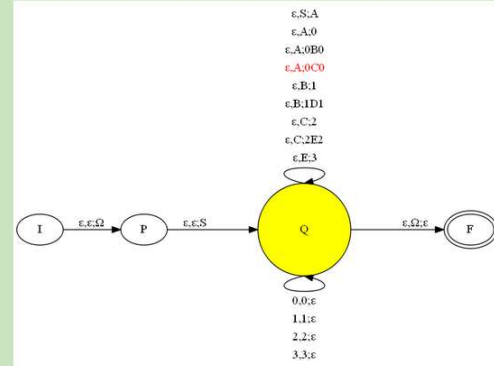
Estado de aceptación={f}



T4: Genero los pasos de esa gramática

+ITERACION:3

Pila	0C0Ω
Entrada	02320



+ITERACION:4

Pila	C0Ω
Entrada	2320

ε,S,A
ε,A,0
ε,A,0B0
ε,A,0C0
ε,B,1
ε,B,1D1
ε,C,2
ε,C,2E2
ε,E,3
0,0,ε
1,1,ε
2,2,ε
3,3,ε

T5: Genera un de una gramática regular antes seleccionada generado en html

Iteracion	Pila	Entrada	Transaccion
0		02320	(i, ε, ε;p,Ω)
1	Ω	02320	(p, ε, ε;q,A)
2	AΩ	02320	q,ε,S,q,A
3	0C0Ω	02320	q,ε,A,q,0C0
4	C0Ω	2320	q,0,0,q,ε
5	2E20Ω	2320	q,ε,C,q,2E2
6	E20Ω	320	q,2,2,q,ε
7	320Ω	320	q,ε,E,q,3
8	20Ω	20	q,3,3,q,ε
9	0Ω	0	q,2,2,q,ε
10	Ω	ε	(q,Ω,ε;f,ε)
11	ε	ε	f

T6:salir

MANUAL TÉCNICO

Analizador

Analizador: analiza si la cadena en el .glc es valida

De_or: sustituye -> por un | si viene mas de una vez una producción con la misma letra

Automata_pila: genera las cadenas para los nodo en graphviz

Html: genera el html mostrando la img de grahpviz

AutomataHtml

Gramática_inf

Inf: muestra la información de la gramática en consola

Lector

Lee los archivos .glc

Menú

Genera el menu

ordenarP:

ordenar: ordena las gramticas por terminales y no terminales dando le prioridad los terminales

pasos:

genera el html de las transacciones

tabla

complemento de pasos

transacciones

genera las tabla de transacciones [q,e,S,A]

EXPLICACIÓN AFD: Elimina los caminos que interrumpen la cadena, para generar el afd.

T NT		a[0]	a[1]	a[2]	a[3]	(S,02320,[],{})		(S,02320,[],{})		(S,02320,[],{})
NT	1	ε	S	A		(A,02320,[A],{})	IF []!=E	(A,02320,[A],{})	IF []!=E	(A,02320,[A],{})
T	2	ε	A	O		(0,02320,[O],{})	IF []==E & TXT[0]=[O]	(0,02320,[O B O],{})	IF []==E & TXT[0]=[O]	(0,02320,[O B O],{})
T;NT;T	3	ε	A	O B O		(0,2320,[],{0})	IF TXT=0 & IF[]=0	(B,2320,[B O],{0})	IF []!=E	(B,2320,[B O],{0})
T;NT;T	4	ε	A	O C O		FIN:		(1,2320,[1 O],{})	IF TXT=0 & IF[]=0	(1,2320,[1 D 1 O],{0})
NT	5	ε	B	1				FIN:		FIN:
T;NT;T	6	ε	B	1 D 1						
T	7	ε	C	2						
T;NT;T	8	ε	C	2 E 2						
T	9	ε	E	3						
	10	O	O	ε		S1-A2	IF []==NT	S1-A3-B5	IF []==NT	S1-A3-B6
	11	1	1	ε		BUSCAR(A2)		BUSCAR(B5)		BUSCAR(B6)
	12	2	2	ε		ELIMINAR(A2)		ELIMINAR(B5)		ELIMINAR(B6)
	13	3	3	ε		S1		S1-A3		S1-A3
					PILA=0					(S,02320,[],{})
					TXT[0]=PILA[0]					(A,02320,[A],{})
						PILA=0				(0,02320,[O B O],{})
						PILA=0				(B,2320,[B O],{0})
					TXT[0]=PILA[0]					FIN:
										BUSCAR(A3)
										ELIMINAR(A3)
										S1

EXPLICACIÓN GRAMTICA: Con una condición se verifico si los terminales no fuera<2 y que si los no terminales fueran<2, para filtrar las gramaticas