

AGUSTÍN TAMBORINI

168012-2



2019
Curso K2055

Trabajo práctico Nº 1

Normativa: la entrega se hace a mi correo roxanaleituz@yahoo.com.ar. Entregar con carátula detallando, nombre, legajo, curso, año. Los desarrollos en C deben estar correctamente implementados, comentados y con una interfaz mínima que permita realizar pruebas.

Fecha de entrega: 9/8

Modalidad: individual

Nota IMPORTANTE: si se registraran dos trabajos iguales/copia se pierde automáticamente la promoción.

1. Dado un alfabeto de 10 símbolos como mínimo.

1.1. Defina formalmente el alfabeto

1.2. Defina un lenguaje formal finito y otro infinito sobre el alfabeto. Por comprensión y/o lenguaje natural.

1.3. Escriba la definición formal de una gramática que los genere.

1.4. Escriba las 5 palabras de menor longitud.

1.5. Escriba reglas en BNF que definan la sintaxis.

1.6. Escriba una ER que los represente usando operadores originales.

1.7. Dibuje el Diagrama de Transiciones de un Autómata Finito que los represente.

• Cuando se pide que describan un AFD se pide su definición formal y el Diagrama de transición o tabla de transición, ambos se considerarán válidos, a menos que se especifique otra cosa.

2. Describa los AFD que aceptan los siguientes lenguajes con el alfabeto $\{0,1\}$:

2.1. Diseñar un AFD que acepte el lenguaje que tiene un número par de unos y ceros sobre el alfabeto 0,1. Haga el diagrama de transición y la tabla de transición completa.

2.2. El conjunto de todas las cadenas tales que cada bloque de cinco símbolos consecutivos contenga al menos dos ceros.

2.3. El conjunto de todas las cadenas cuyo símbolo en la décima posición respecto del extremo derecho sea un 1.

2.4. El conjunto de cadenas que empiecen o terminen (o ambas cosas) con 01.

2.5. El conjunto de las cadenas tales que el número de ceros es divisible por cinco y el número de unos es divisible por 3.

2.6. Dar un AFD que acepte como lenguaje:

a) $L = \{\text{cadenas de ceros y unos tal que no contengan un número de par de ceros si acaban en } 11\}$.

b) $L = \{\text{cadenas de a y b tal que la longitud de la cadena sea par y además no acaben en b}\}$

3. Sean x, y, z, u, v y t, variables que contienen, respectivamente, los valores 2, 3, 4, 5, 6 y 7, ¿qué almacenarán después de ejecutar las siguientes sentencias?

```
x++;  
y = ++z;  
t = --v;  
v = x + (y * 3) / 2;  
u = x + y / 2;
```

4. ¿Qué diferencia hay entre '\a' y '\A'? ¿Y entre 'A' y "A"?

5. Indica cuáles de las expresiones siguientes son verdaderas y cuáles falsas, suponiendo que x, y, z y t, almacenan, respectivamente los valores 20, 10, 5 y 2.

5.1. $x > y \ \&\& \ z > t$ b) $x < y \ \&\& \ z > t$ c) $x < y \ || \ z > t$

5.2. $!0$ e) $!1$ f) $0 != 1$

5.3. $0 != !1$ h) $0 == !1$ i) $200 || x$

5.4. $x * 0$ k) $x * y$ l) $!!0$

5.5. $!(0 == 0)$ n) $10 > 5 \ \&\& \ !(10 < 9) \ || \ 3 <= 4$

5.6. $1 \ \&\& \ !0 \ || \ 1$ o) $1 \ \&\& \ !(0 \ || \ 1)$

6. Sea x una variable entera que almacena el valor 10 ¿Qué almacenará y después de las siguientes sentencias?

a) $y = (x > 9 ? ++x : --x);$

b) $y = (x > 9 ? x++ : x--);$

¿Y si x almacenase el valor 8?

7. En un laboratorio se analizan trozos de códigos de ADN humanos. En dicho código pueden aparecer 4 tipos distintos de bases nitrogenadas : Adenina (A), Timina (T), Citosina (C) y Guanina (G) En un experimento se busca si un determinado individuo tiene una determinada “marca genética”. Se dice que un individuo tiene dicha marca si: i) en su muestra de ADN aparece el patrón CGGC y ii) en dicha muestra no hay un número impar de bases del tipo Timina (T). Diseñar un Autómata Finito Determinista (AFD) que detecte, dada la muestra de un individuo, si éste lleva la marca genética citada.

8. Una máquina de gaseosas se comporta como sigue: • las gaseosas grandes cuestan 125 pesos • sólo acepta monedas de cinco y veinte se denotarán con “C” las monedas de “Cinco” y con “V” las de “Veinte” • no sabe dar cambio: en cuanto ha ingresado más de 125 devuelve todas las monedas introducidas hasta ese momento. Sea L el lenguaje formado por las combinaciones de C’s y V’s por las que la máquina da una botella grande.

- o Construir un AFD que reconozca el lenguaje especificado.
- o Construir la expresión regular que especifica el lenguaje reconocido por el autómata.
- o Programarlo en C.

Ejercicio 1

1.1)

$$\Sigma = \{a, b, 0, 1, 2, 3, 4, 5, 6, 7\}$$

1.2)

$$LFF = \{\epsilon, ab^n0 \mid 0 \leq n \leq 3\}$$

$$LFI = \{a^n b0 \mid n \geq 0\}$$

1.3)

Gramática para el lenguaje formal finito (LFF):

$$G = (\{S, T, R, Q\}, \{a, b, 0, 1, 2, 3, 4, 5, 6, 7\}, \{P\}, S)$$

$$P = S \rightarrow aT0$$

$$T \rightarrow \epsilon \mid bR$$

$$R \rightarrow \epsilon \mid bQ$$

$$Q \rightarrow \epsilon \mid b$$

Gramática para el lenguaje formal infinito (LFI):

$$G = (\{S, R\}, \{a, b, 0, 1, 2, 3, 4, 5, 6, 7\}, \{P\}, S)$$

$$P = S \rightarrow aS \mid bR$$

$$R \rightarrow 0$$

1.4)

5 palabras de menor longitud para el lenguaje formal finito (LFF):

$$\epsilon, a0, ab0, abb0, abbb0$$

5 palabras de menor longitud para el lenguaje formal infinito (LFI):

$$b0, ab0, aab0, aaab0, aaaab0$$

1.5)

Reglas BNF para el lenguaje finito (LFF):

Identificador \rightarrow letraPrincipal identificadorSecundario numeroPrincipal
 identificadorSecundario \rightarrow vacio | letraSecundaria identificadorTerciario
 identificadorTerciario \rightarrow vacio | letraSecundaria identificadorCuaternario
 identificadorCuaternario \rightarrow vacio | letraSecundaria
 letraPrincipal \rightarrow a
 numeroPrincipal \rightarrow 0
 vacio $\rightarrow \epsilon$
 letraSecundaria \rightarrow b

Reglas BNF para el lenguaje infinito (LFI):

Identificador \rightarrow letraPrincipal identificador | letraSecundaria
 identificadorSecundario
 identificadorSecundario \rightarrow numeroPrincipal
 letraPrincipal \rightarrow a
 numeroPrincipal \rightarrow 0
 letraSecundaria \rightarrow b

1.6)

ER-LFF: $(a(\epsilon + b + bb + bbb)0) + \epsilon$

ER-LFI: a^*b0

1.7)

Diagrama de transición de un AF para el lenguaje finito (LFF)

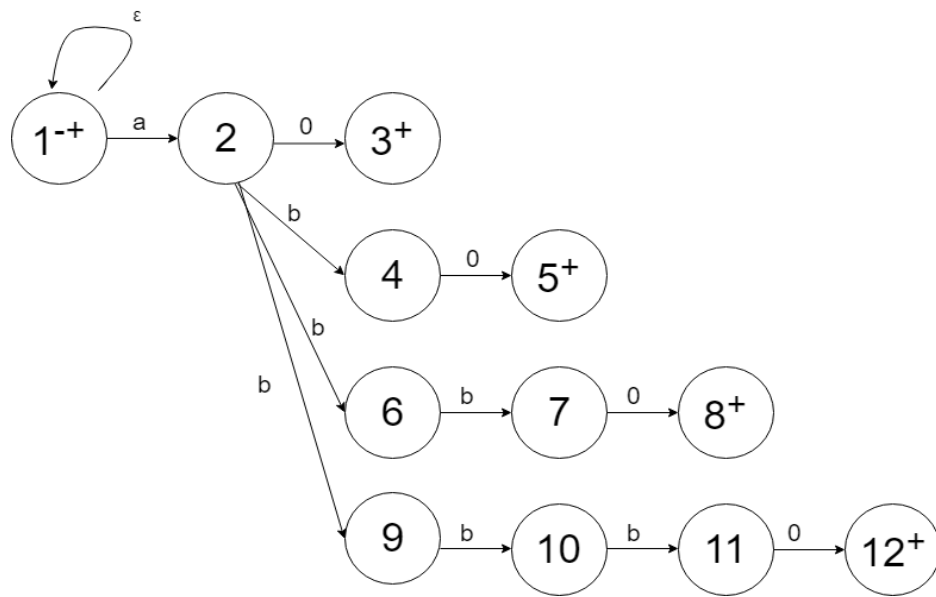
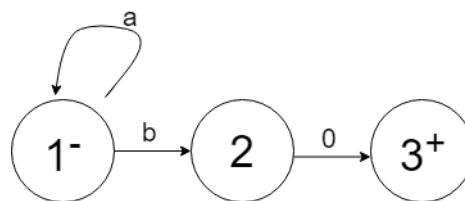
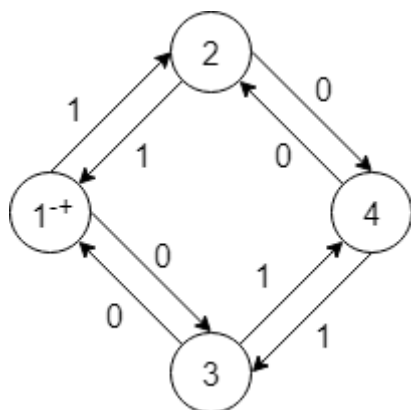


Diagrama de transición de un AF para el lenguaje infinito (LFI)



Ejercicio 2

2.1)



T	0	1
1 ⁺	3	2
2	4	1
3	1	4
4	2	3

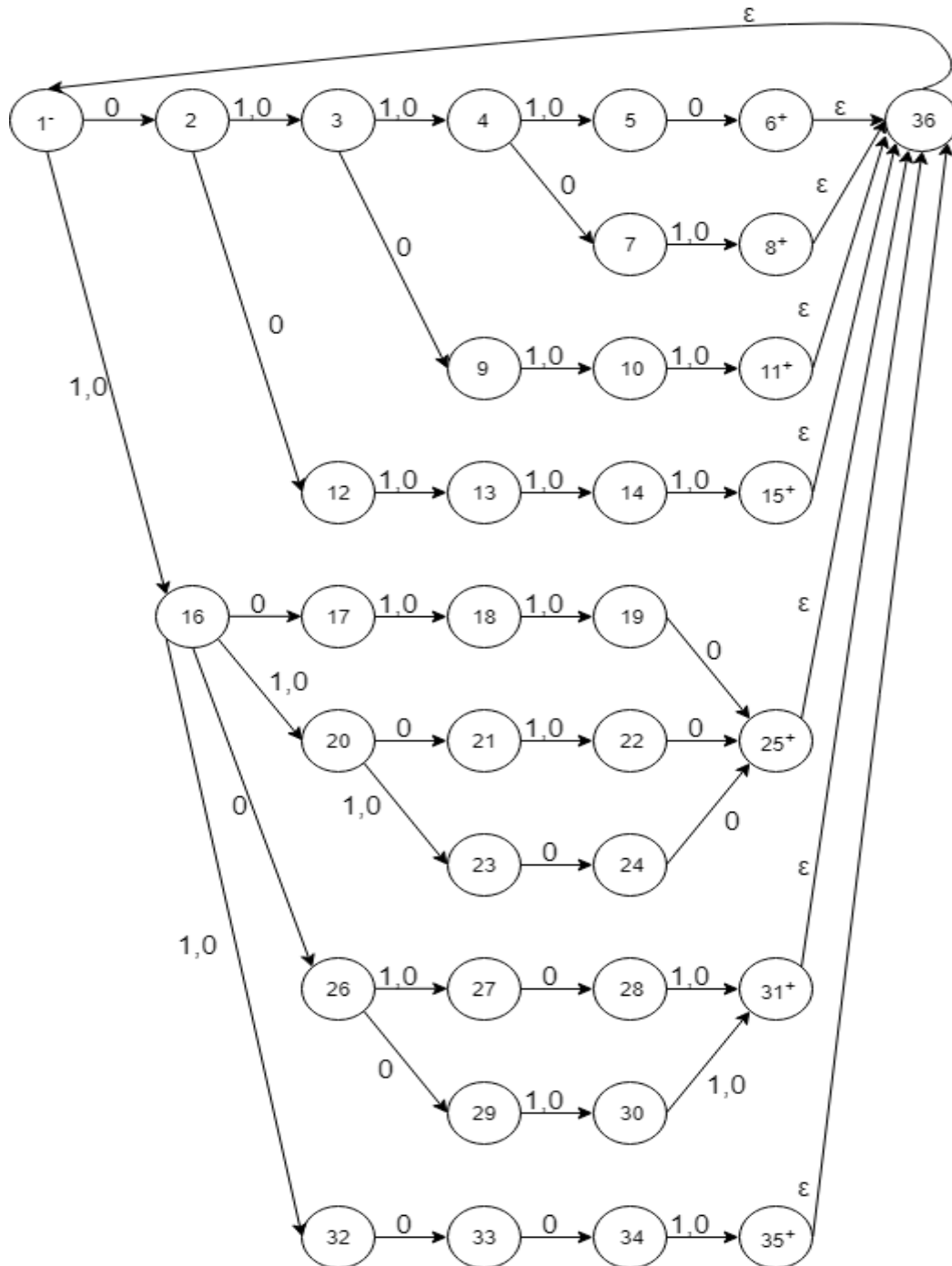
AFD = ({1,2,3,4}, {1,0}, T, 1, {1})

2.2)

✓ Planteo una ER equivalente al lenguaje:

$$ER = (0(1+0)(1+0)(1+0)0 + 0(1+0)(1+0)0(1+0) + 0(1+0)0(1+0)(1+0) + 00(1+0)(1+0)(1+0) + (1+0)0(1+0)(1+0)0 + (1+0)(1+0)0(1+0)0 + (1+0)(1+0)(1+0)00 + (1+0)0(1+0)0(1+0) + (1+0)00(1+0)(1+0) + (1+0)(1+0)00(1+0))^+$$

✓ Construyo un AFN que reconozca dicha ER:



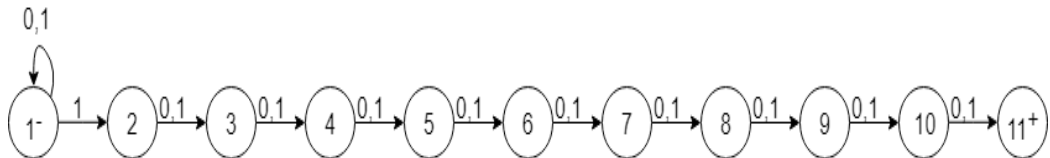
AFN = ({n / 1 ≤ n ≤ 36}, {1,0}, TT, 1, {6,8,11,15,25,31,35})

2.3)

- ✓ Obtengo una expresión regular equivalente al lenguaje:

$$ER = (1+0)^*1(1+0)^9$$

- ✓ Genero un AFN que pueda interpretar dicha ER:



TT	0	1
1 ⁻	{1}	{1,2}
2	{3}	{3}
3	{4}	{4}
4	{5}	{5}
5	{6}	{6}
6	{7}	{7}
7	{8}	{8}
8	{9}	{9}
9	{10}	{10}
10	{11}	{11}
11 ⁺	-	-

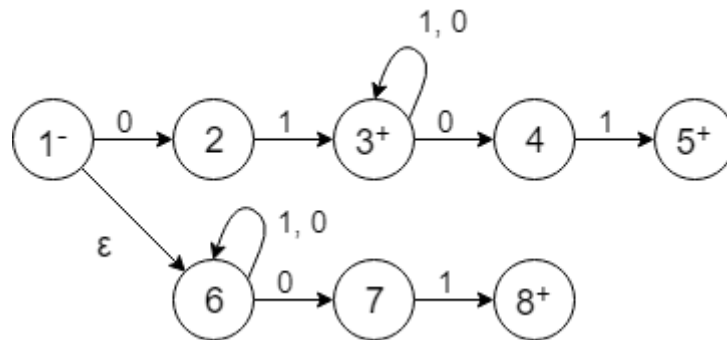
$$AFN = (\{1,2,3,4,5,6,7,8,9,10,11\}, \{1,0\}, TT, 1, \{11\})$$

2.4)

- ✓ Obtengo una expresión regular equivalente al lenguaje:

$$ER = 01(1+0)^* + (1+0)^*01 + 01(1+0)^*01$$

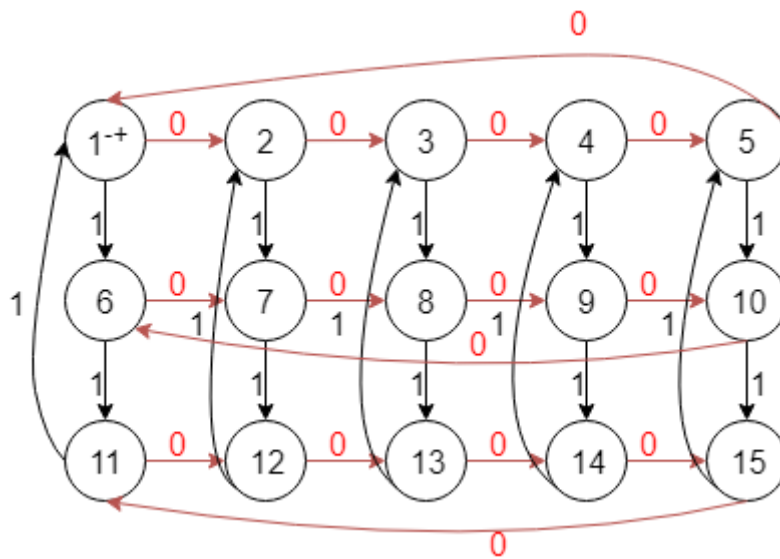
- ✓ Convierto la expresión regular en un AFN:



TT	0	1	ϵ
1 ⁻	{2}	-	{6}
2	-	{3}	-
3 ⁺	{3, 4}	{3}	-
4	-	{5}	-
5 ⁺	-	-	-
6	{6, 7}	{6}	-
7	-	{8}	-
8 ⁺	-	-	-

$$AFN = (\{1,2,3,4,5,6,7,8\}, \{1,0\}, TT, 1, \{3,5,8\})$$

2.5)



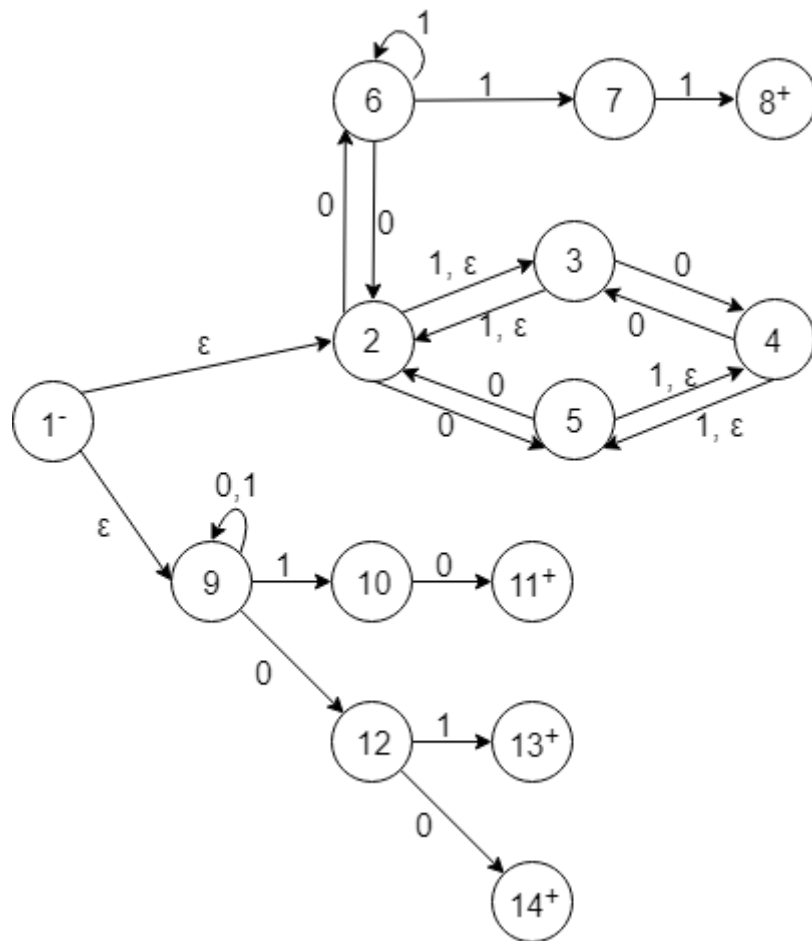
T	0	1
1 ⁺	2	6
2	3	7
3	4	8
4	5	9
5	1	10
6	7	11
7	8	12
8	9	13
9	10	14
10	6	15
11	12	1
12	13	16
13	14	16
14	15	16
15	11	16
16	16	16

*16 es el estado de rechazo. Lo utilizamos con el fin de hacer al autómata completo.

AFD = ({1,2,3,4,5,6,7,8,9,10,11,12,13,14,15,16}, {1,0}, T, 1, {1})

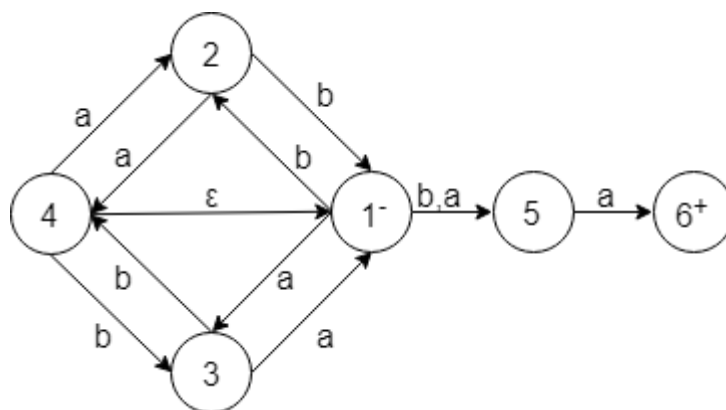
2.6)

a_



AFN = ({1,2,3,4,5,6,7,8,9,10,11,12,13,14}, {1,0}, T, 1, {8,11,13,14})

b_



AFN = ({1,2,3,4,5,6}, {a,b}, T, 1, {6})

Ejercicio 3

✓ $X = 2$

✓ $Y = 3$

✓ $Z = 4$

✓ $U = 5$

✓ $V = 6$

✓ $T = 7$

✓ Luego de ejecutar la sentencia ' $X++$;' la variable X valdrá 3.

✓ Luego de ejecutar la sentencia ' $Y = ++Z$;' la variable Y valdrá 5.

✓ Luego de ejecutar la sentencia ' $T = --V$;' la variable T valdrá 5.

✓ Luego de ejecutar la sentencia ' $V = X + (Y *= 3) / 2$;' la variable V valdrá 6 (si trabajamos con números enteros o int) o 6,5 (si trabajamos con números decimales o float).

✓ Luego de ejecutar la sentencia ' $U = X + Y / 2$;' la variable U valdrá 3 (si trabajamos con números enteros o int) o 3,5 (si trabajamos con números decimales o float).

Ejercicio 4

La operación de control '\a' permite al usuario ejecutar un sonido de alerta mientras que '\A' no es reconocido como ninguna secuencia de escape (simplemente se imprime el carácter 'A'). La diferencia entre 'A' y "A" es que, si bien ambas tienen la misma salida al imprimir, a 'A' lo estamos tratando como un único carácter mientras que "A" está actuando como una cadena de texto, que es una concatenación de caracteres.

Ejercicio 5

✓ $X = 20$

✓ $Y = 10$

✓ $Z = 5$

✓ $T = 2$

5.1) a_ 'x > y && z > t'. Es **verdadero**.

b_ 'x < y && z > t'. Es **falso**.

c_ 'x < y || z > t'. Es **verdadero**.

5.2) d_ '!0'. Es **verdadero**.

e_ '!1'. Es **falso**.

f_ '0 != 1'. Es **verdadero**.

5.3) g_ '0 != !1'. Es **falso**.

h_ '0 == !1'. Es **verdadero**.

i_ '200 || x'. Es **verdadero**.

5.4) j_ 'x * 0'. Es **falso**.

k_ 'x * y'. Es **verdadero**.

l_ '!!0'. Es **falso**.

5.5) m_ '!(0 == 0)'. Es **falso**.

n_ '10 > 5 && !(10 < 9) || 3 <= 4'. Es **Verdadero**.

5.6) $\neg_1 '1 \ \&\& \ !0 \ || \ 1'$. Es **verdadero**.

o_ $'1 \ \&\& \ !(0 \ || \ 1)'$. Es **falso**.

Ejercicio 6

✓ Si $X = 10$

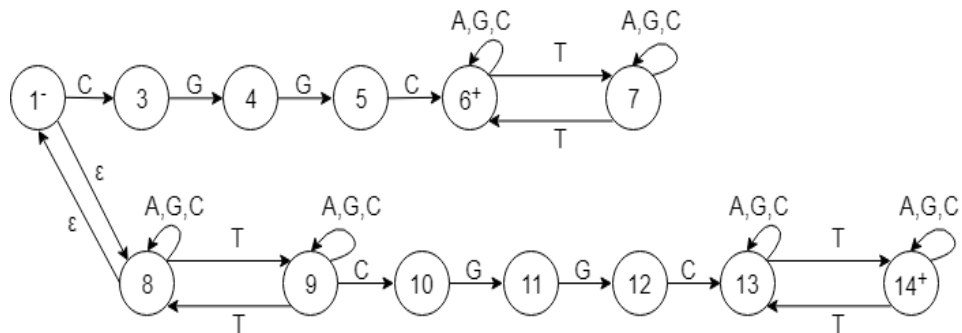
- $y = (x > 9 ? ++x : --x);$
 - La variable y tomará el valor 11.
- $y = (x > 9 ? x++ : x--);$
 - La variable y tomará el valor 11.

✓ Si $X = 8$

- $y = (x > 9 ? ++x : --x);$
 - La variable y tomará el valor 7.
- $y = (x > 9 ? x++ : x--);$
 - La variable y tomará el valor 7.

Ejercicio 7

- ✓ Adenina: A
 - ✓ Timina: T
 - ✓ Citosina: C
 - ✓ Guanina: G
-
- ✓ Una cadena es reconocida cuando:
 - Aparece el patrón CGGC.
 - La Timina (T) siempre es par.
-
- ✓ Obtengo un AFN que interprete el lenguaje pedido:



	A	G	C	T	ϵ
1-	-	-	{3}	-	{8}
3	-	{4}	-	-	-
4	-	{5}	-	-	-
5	-	-	{6}	-	-
6+	{6}	{6}	{6}	{7}	-
7	{7}	{7}	{7}	{6}	-
8	{8}	{8}	{8}	{9}	{1}
9	{9}	{9}	{9, 10}	{8}	-
10	-	{11}	-	-	-
11	-	{12}	-	-	-
12	-	-	{13}	-	-
13	{13}	{13}	{13}	{14}	-
14+	{14}	{14}	{14}	{13}	-

✓ Convierto el AFN en AFD a través del método del conjunto clausura:

	A	G	C	T
$\{1,8\}^-$	$\{1,8\}$	$\{1,8\}$	$\{1,3,8\}$	$\{9\}$
$\{1,3,8\}$	$\{1,8\}$	$\{1,4,8\}$	$\{1,3,8\}$	$\{9\}$
$\{9\}$	$\{9\}$	$\{9\}$	$\{9,10\}$	$\{1,8\}$
$\{1,4,8\}$	$\{1,8\}$	$\{1,5,8\}$	$\{1,3,8\}$	$\{9\}$
$\{9,10\}$	$\{9\}$	$\{9,11\}$	$\{9,10\}$	$\{1,8\}$
$\{1,5,8\}$	$\{1,8\}$	$\{1,8\}$	$\{1,3,6,8\}$	$\{9\}$
$\{9,11\}$	$\{9\}$	$\{9,12\}$	$\{9,10\}$	$\{1,8\}$
$\{1,3,6,8\}^+$	$\{1,6,8\}$	$\{1,4,6,8\}$	$\{1,3,6,8\}$	$\{7,9\}$
$\{9,12\}$	$\{9\}$	$\{9\}$	$\{9,10,13\}$	$\{1,8\}$
$\{1,6,8\}^+$	$\{1,6,8\}$	$\{1,6,8\}$	$\{1,3,6,8\}$	$\{7,9\}$
$\{1,4,6,8\}^+$	$\{1,6,8\}$	$\{1,5,6,8\}$	$\{1,3,6,8\}$	$\{7,9\}$
$\{7,9\}$	$\{7,9\}$	$\{7,9\}$	$\{7,9,10\}$	$\{1,6,8\}$
$\{9,10,13\}$	$\{9,13\}$	$\{9,11,13\}$	$\{9,10,13\}$	$\{1,8,14\}$
$\{1,5,6,8\}^+$	$\{1,6,8\}$	$\{1,6,8\}$	$\{1,3,6,8\}$	$\{7,9\}$
$\{7,9,10\}$	$\{7,9\}$	$\{7,9,11\}$	$\{7,9,10\}$	$\{1,6,8\}$
$\{9,13\}$	$\{9,13\}$	$\{9,13\}$	$\{9,10,13\}$	$\{1,8,14\}$
$\{9,11,13\}$	$\{9,13\}$	$\{9,12,13\}$	$\{9,10,13\}$	$\{1,8,14\}$
$\{1,8,14\}^+$	$\{1,8,14\}$	$\{1,8,14\}$	$\{1,3,8,14\}$	$\{9,13\}$
$\{7,9,11\}$	$\{7,9\}$	$\{7,9,12\}$	$\{7,9,10\}$	$\{1,6,8\}$
$\{9,12,13\}$	$\{9,13\}$	$\{9,13\}$	$\{9,10,13\}$	$\{1,8,14\}$
$\{1,3,8,14\}^+$	$\{1,8,14\}$	$\{1,4,8,14\}$	$\{1,3,8,14\}$	$\{9,13\}$
$\{7,9,12\}$	$\{7,9\}$	$\{7,9\}$	$\{7,9,10,13\}$	$\{1,6,8\}$
$\{1,4,8,14\}^+$	$\{1,8,14\}$	$\{1,5,8,14\}$	$\{1,3,8,14\}$	$\{9,13\}$
$\{7,9,10,13\}$	$\{7,9,13\}$	$\{7,9,11,13\}$	$\{7,9,10,13\}$	$\{1,6,8,14\}$
$\{1,5,8,14\}^+$	$\{1,8,14\}$	$\{1,8,14\}$	$\{1,3,6,8,14\}$	$\{9,13\}$
$\{7,9,13\}$	$\{7,9,13\}$	$\{7,9,13\}$	$\{7,9,10,13\}$	$\{1,6,8,14\}$
$\{7,9,11,13\}$	$\{7,9,13\}$	$\{7,9,12,13\}$	$\{7,9,10,13\}$	$\{1,6,8,14\}$
$\{1,6,8,14\}^+$	$\{1,6,8,14\}$	$\{1,6,8,14\}$	$\{1,3,6,8,14\}$	$\{7,9,13\}$
$\{1,3,6,8,14\}^+$	$\{1,6,8,14\}$	$\{1,4,6,8,14\}$	$\{1,3,6,8,14\}$	$\{7,9,13\}$

{7,9,12,13}	{7,9,13}	{7,9,13}	{7,9,10,13}	{1,6,8,14}
{1,4,6,8,14} ⁺	{1,6,8,14}	{1,5,6,8,14}	{1,3,6,8,14}	{7,9,13}
{1,5,6,8,14} ⁺	{1,6,8,14}	{1,6,8,14}	{1,3,6,8,14}	{7,9,13}

TT	A	G	C	T
1 ⁻	1	1	2	3
2	1	4	2	3
3	3	3	5	1
4	1	6	2	3
5	3	7	5	1
6	1	1	8	3
7	3	9	5	1
8 ⁺	10	11	8	12
9	3	3	13	1
10 ⁺	10	10	8	12
11 ⁺	10	14	8	12
12	12	12	15	10
13	16	17	13	18
14 ⁺	10	10	8	12
15	12	19	15	10
16	16	16	13	18
17	16	20	13	18
18 ⁺	18	18	21	16
19	12	22	15	10
20	16	16	13	18
21 ⁺	18	23	21	16
22	12	12	24	10
23 ⁺	18	25	21	16
24	26	27	24	28
25 ⁺	18	18	29	16
26	26	26	24	28
27	26	30	24	28

28 ⁺	28	28	29	26
29 ⁺	28	31	29	26
30	26	26	24	28
31 ⁺	28	32	29	26
32 ⁺	28	28	29	26

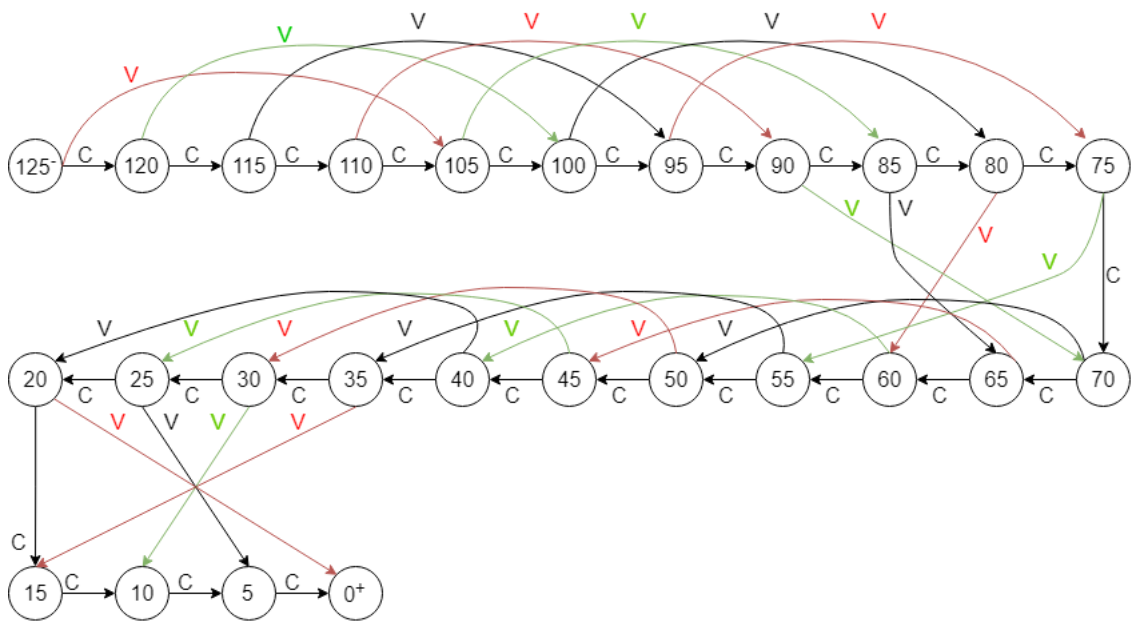
✓ Damos la definición formal del AFD obtenido:

AFD = ({n / 1 ≤ n ≤ 32}, {A,G,C,T}, TT, 1, {8,10,11,14,18,21,23,25,28,29,31,32})

Ejercicio 8

- ✓ Cada gaseosa cuesta 125
- ✓ C: Monedas de 5
- ✓ V: Monedas de 20
- ✓ Al pasarse de 125, devuelve todas las monedas (es decir que no entra en el conjunto de palabras reconocidas por el autómata).

○ Autómata Finito Determinista:



T	C	V
125	120	105
120	115	100
115	110	95
110	105	90
105	100	85
100	95	80
95	90	75
90	85	70
85	80	65
80	75	60
75	70	55

70	65	50
65	60	45
60	55	40
55	50	35
50	45	30
45	40	25
40	35	20
35	30	15
30	25	10
25	20	5
20	15	0
15	10	1000
10	5	1000
5	0	1000
0+	1000	1000
1000	1000	1000

AFD = ({5n / 0 ≤ n ≤ 25}, {C,V}, T, 125, {0})

- Expresión regular

$$(C+V)^{21}V + (C+V)^{21}C^4$$

- Programación en C

```

✓ #include <stdio.h>
✓
✓ void introduccion();
✓ void automata(int[][2], char[]);
✓ int averiguarColumna(char);
✓
✓ int main(){
✓     /*Genero la matriz del automata*/
✓     int tablaTransicion[27][2] = {
✓         {1, 4}, //Estado 125 (Inicial) - 0
✓         {2, 5}, //Estado 120 - 1
✓         {3, 6}, //Estado 115 - 2
✓         {4, 7}, //Estado 110 - 3
✓         {5, 8}, //Estado 105 - 4
✓         {6, 9}, //Estado 100 - 5

```



```

✓      {7, 10}, //Estado 95 - 6
✓      {8, 11}, //Estado 90 - 7
✓      {9, 12}, //Estado 85 - 8
✓      {10, 13}, //Estado 80 - 9
✓      {11, 14}, //Estado 75 - 10
✓      {12, 15}, //Estado 70 - 11
✓      {13, 16}, //Estado 65 - 12
✓      {14, 17}, //Estado 60 - 13
✓      {15, 18}, //Estado 55 - 14
✓      {16, 19}, //Estado 50 - 15
✓      {17, 20}, //Estado 45 - 16
✓      {18, 21}, //Estado 40 - 17
✓      {19, 22}, //Estado 35 - 18
✓      {20, 23}, //Estado 30 - 19
✓      {21, 24}, //Estado 25 - 20
✓      {22, 25}, //Estado 20 - 21
✓      {23, 26}, //Estado 15 - 22
✓      {24, 26}, //Estado 10 - 23
✓      {25, 26}, //Estado 5 - 24
✓      {26, 26}, //Estado 0 (Final) - 25
✓      {26, 26} //Estado 1000 (Rechazo) - 26
✓  };
✓
✓  introduccion();
✓
✓  system("cls");
✓  /*Ingreso la palabra que pasara por el automata*/
✓  char cadena[100];
✓  printf("Ingrese la combinacion de monedas que desee (Solo
mayusculas): ");
✓  scanf("%s", cadena);
✓
✓  automata(tablaTransicion, cadena);
✓
✓  system("pause");
✓  return 0;
✓ }
✓
✓ void introduccion(){
✓     printf("***** Maquina de Refrescos *****\n");
✓     printf("El automata reconocera solamente monedas de 5 y 20
centavos\n");
✓     printf("Las mismas estaran representadas por los caracteres 'C'
de Cinco y 'V' de Veinte\n\n");
✓     system("pause");
✓ }
✓
✓ void automata(int tabla[][2], char palabra[]){
✓     int estado = 0;

```

```

✓      int contador = 0;
✓      while(palabra[contador] != 0 && estado != 26){
✓          estado =
tabla[estado][averiguarColumna(palabra[contador])]; //Averiguamos
dependiendo el caracter de la palabra a que columna se dirige en la
Tabla de Transicion
✓          contador++;
✓      }
✓      //Averiguo cual fue el ultimo estado y asigno si cumple con que
este sea un estado final o no//
✓      if(estado != 25) printf("\nLa combinacion '%s' no es exactamente
125, por lo que sus monedas seran devueltas.\n\n", palabra);
✓      else printf("\nLa combinacion '%s' es exactamente 125, disfrute
su bebida!\n\n", palabra);
✓  }
✓
✓  int averiguarColumna(char character){
✓      if(caracter == 'C') return 0; //Si vale C, entonces utilizamos
la primera columna de la matriz
✓      return 1; //En cualquier otro caso (osea que valga V),
utilizamos la segunda columna de la matriz
✓  }

```

*Dejo adjunto en el correo el programa en C y el ejecutable para que se puedan realizar los testeos deseados.