Segundo trabajo Matemática Computacional

Maria Juliana Mantilla Morales

Julio 2020

1 Lenguaje a trabajar

Conjunto de cadenas ω sobre $\Sigma = \{a, b, c\}$ que contienen exactamente dos b's y exactamente una vez la subcadena cc.
Este lenguaje $\mathcal{L}$ se puede ver como el producto cartesiano de los siguientes dos lenguajes:

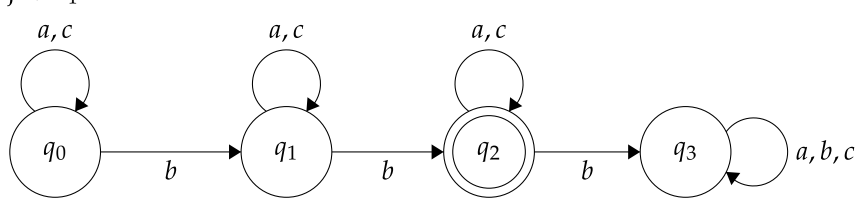
- $\mathcal{M}_1$ = Conjunto de cadenas sobre $\Sigma = \{a, b, c\}$ que contienen exactamente dos b's. Cuya expresión regular esta dada por $(a \vee c)^* b (a \vee c)^* b (a \vee c)^*$.
- $\mathcal{M}_2$ = Conjunto de cadenas sobre $\Sigma = \{a, b, c\}$ que contienen exactamente una vez la subcadena cc. Cuya expresión regular esta dada por $(a \vee b \vee c(a \vee b))^* cc(a \vee b \vee (a \vee b)^+ c)^*$

Por tal motivo, la construcción de el autómata que modela nuestro lenguaje $\mathcal{L}$, se puede dar de la siguiente manera:

2 Autómata

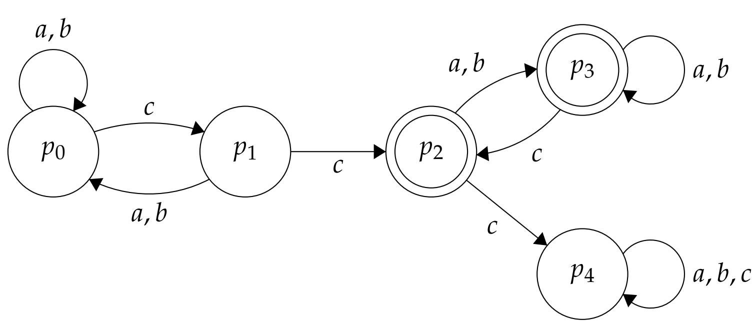
De acuerdo a los dos lenguajes $\mathcal{M}_1$ y $\mathcal{M}_2$ descritos anteriormente, se puede construir los siguientes autómatas que los modelan:

- Para el lenguaje $\mathcal{M}_1$:



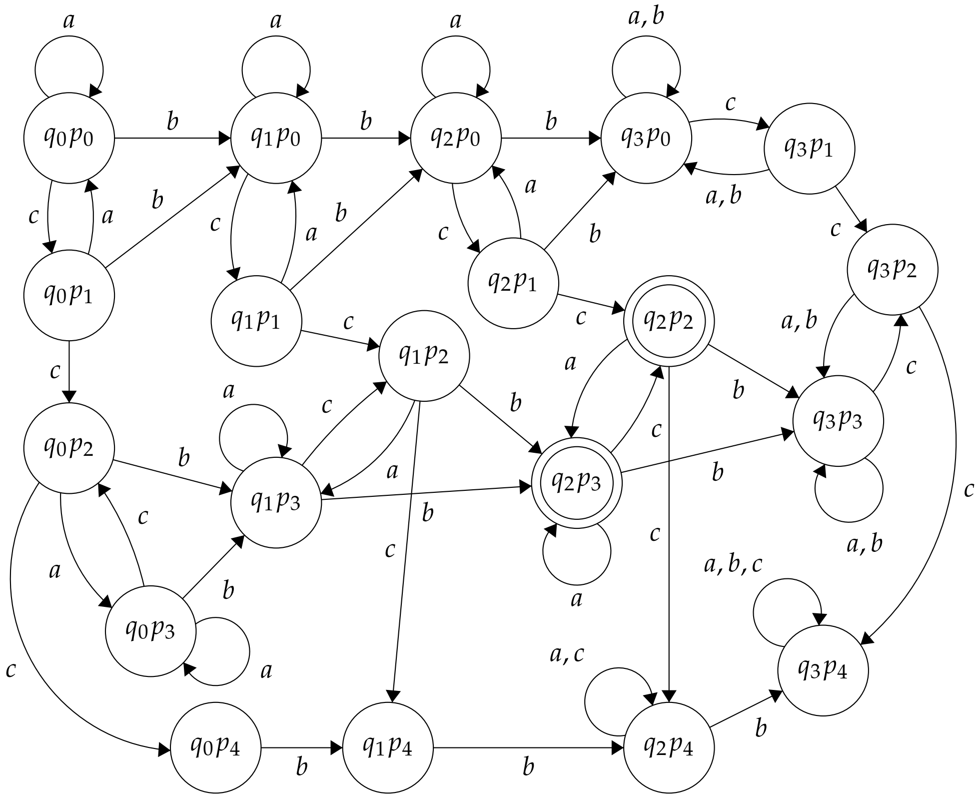
Donde q_0 es el estado inicial, q_2 es el estado de aceptación y q_3 es el infierno.

- Para el lenguaje $\mathcal{M}_2$:



Donde p_0 es el estado inicial, p_2 y p_3 son los estados de aceptación y p_4 es el infierno.

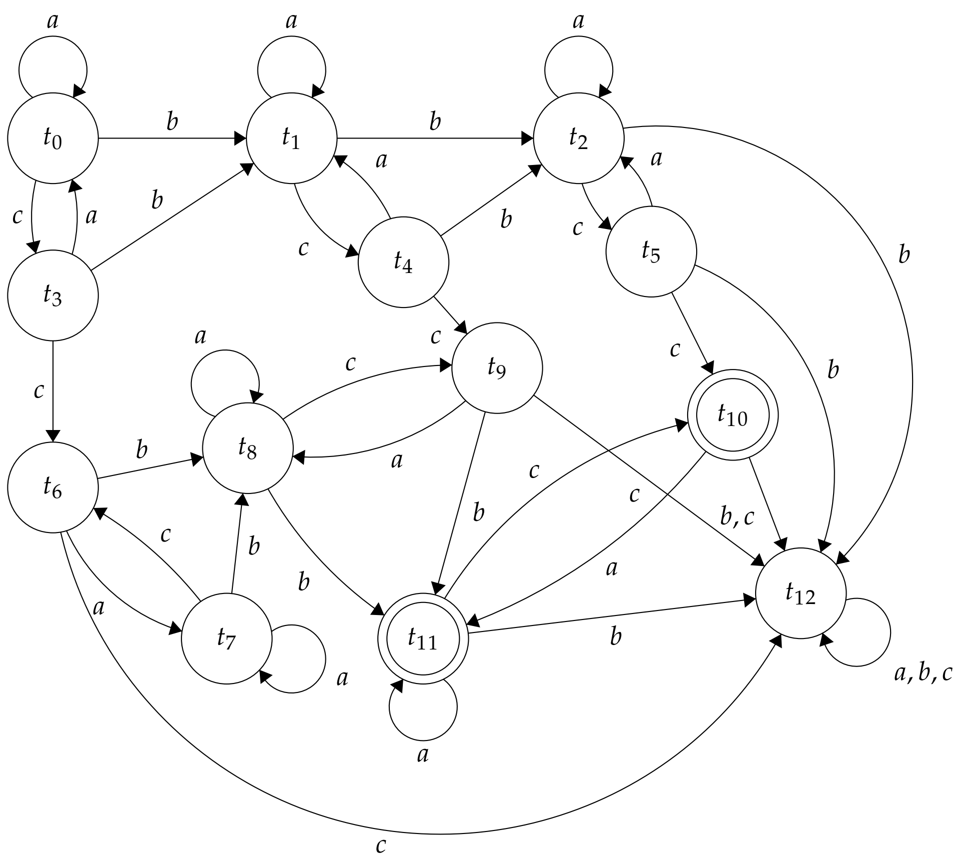
Ahora, para poder construir el autómata que nos modela el lenguaje $\mathcal{L}$, se realiza el producto cartesiano entre estos dos autómatas. Teniendo como resultado, un autómata con 20 estados, de los cuales, q_0p_0 es el estado inicial, los estados q_2p_2 y q_2p_3 son de aceptación y los estados q_3p_0, q_3p_1, q_3p_2, q_3p_3, q_3p_4, q_0p_4, q_1p_4 y q_2p_4 son considerados infierno, pues uno o los dos elementos que conforman el estado, es el infierno ya sea del lenguaje $\mathcal{M}_1$ o del lenguaje $\mathcal{M}_2$. Este autómata se puede ver de la siguiente manera:



Para continuar y reducir la cantidad de estados obtenidos en el autómata anterior, se pueden renombrar de la siguiente manera:

- El estado inicial $q_0p_0 = t_0$.
- Los estados de transición $q_1p_0 = t_1$, $q_2p_0 = t_2$, $q_0p_1 = t_3$, $q_1p_1 = t_4$, $q_2p_1 = t_5$, $q_0p_2 = t_6$, $q_0p_3 = t_7$, $q_1p_3 = t_8$ y $q_1p_2 = t_9$.
- Los estados de aceptación $q_2p_2 = t_{10}$ y $q_2p_3 = t_{11}$.
- El infierno $q_0p_4 = q_1p_4 = q_2p_4 = q_3p_4 = q_3p_3 = q_3p_2 = q_3p_1 = q_3p_0 = t_{12}$.

Teniendo como resultado el siguiente autómata de solo 13 estados incluyendo el infierno, que nos modelará nuestro lenguaje $\mathcal{L}$:



El significado de cada estado de aceptación, se puede determinar de la siguiente manera:

- t_{10} acepta las palabras de $\mathcal{L}$ que terminan en c.
- t_{11} acepta las palabras de $\mathcal{L}$ que terminan en a o b.

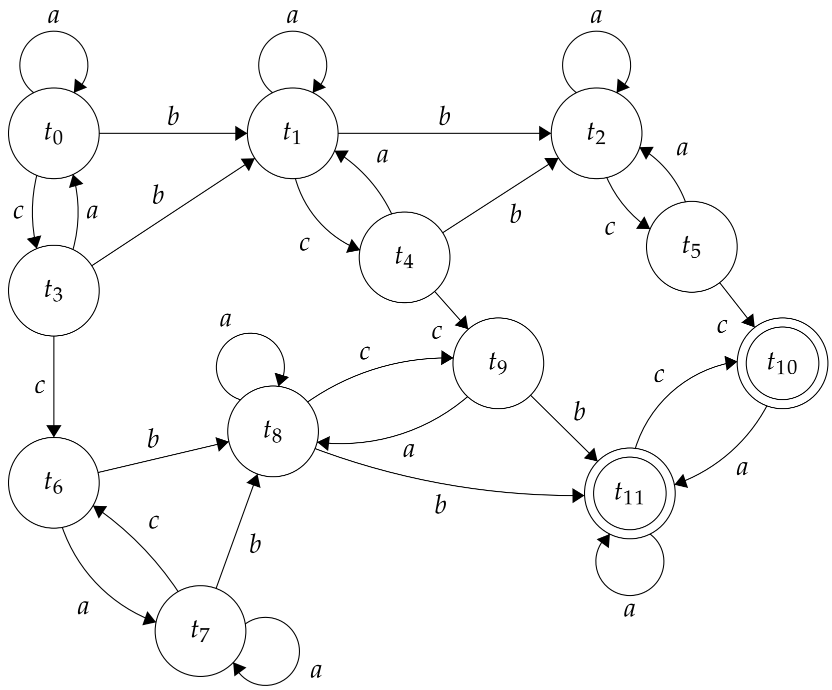
Igualmente, se puede evidenciar que la longitud mas pequeña de las cadenas ω que acepta nuestro lenguaje $\mathcal{L}$ es 4, debido a las condiciones que se deben cumplir. Por lo tanto, se tiene que:

$$\begin{gathered}
\mathcal{L}_0 = \emptyset \\
\mathcal{L}_1 = \emptyset \\
\mathcal{L}_2 = \emptyset \\
\mathcal{L}_3 = \emptyset \\
\mathcal{L}_4 = bbcc, ccbb, bccb. \\
\mathcal{L}_5 = abbcc, babcc, bbacc, cbbcc, bcbcc, accbb, ccabb, ccbab, ccbba, \\
ccbbc, ccbcb, abccb, baccb, bccab, bccba, cbccb, bccbc, bbcca. \\
\vdots
\end{gathered}$$

3 Grafos de expresiones

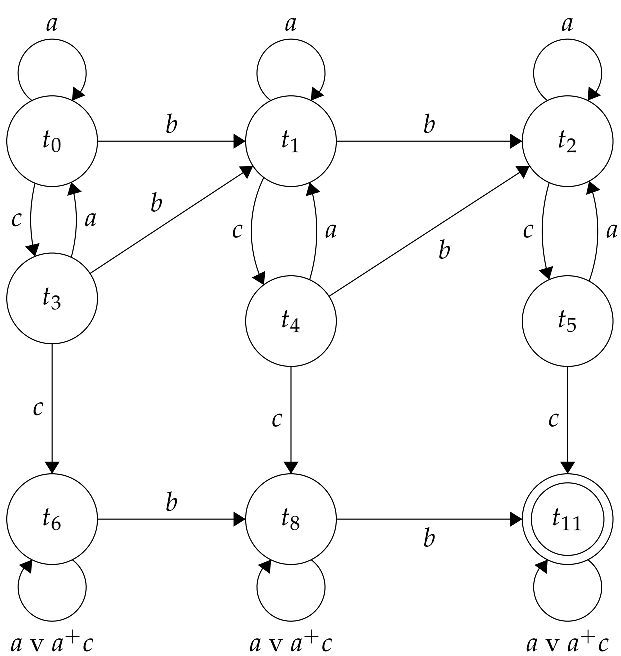
Para encontrar la expresión regular que representa nuestro lenguaje $\mathcal{L}$, emplearemos los grafos de expresiones, debido a que contamos con el autómata que nos modela dicho lenguaje. Para esto realizamos el siguiente procedimiento:

- Primero se elimina el estado considerado como infierno, pues, para la expresión regular nos interesan las cadenas ω que son aceptadas.



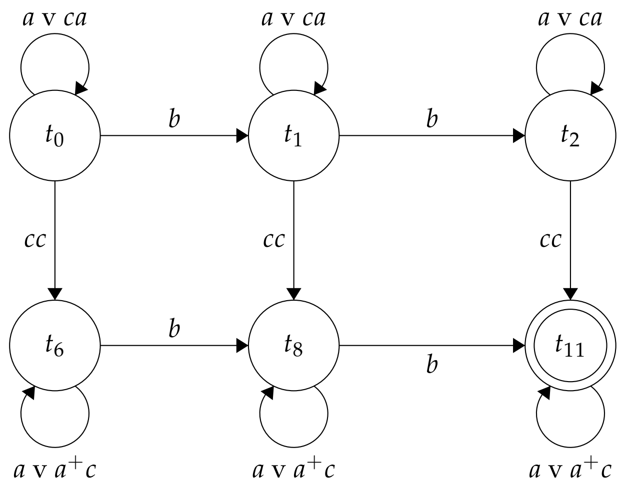
- En segundo lugar, podemos eliminar aquellos estados que presentan un estilo de bucle con otros estados. Como:
  - El estado t_6 con el estado t_7. Por lo tanto, en el estado t_6 queda un bucle con la expresión $a \vee a^+c$ y la arista entre el estado t_6 y el estado t_8 tendrá la expresión b.
  - El estado t_8 con el estado t_9. Por lo tanto, en el estado t_8 queda un bucle con la expresión $a \vee a^+c$ y la arista entre el estado t_8 y el estado t_{10} tendrá la expresión b.
  - El estado t_{10} con el estado t_{11}. Por lo tanto, en el estado t_{10} queda un bucle con la expresión $a \vee a^+c$.

  Por lo cual el grafo nos queda de la siguiente manera:



- Para finalizar, Se observa que aun se puede continuar con el procedimiento anterior, pues hay bucles entre algunos estados, como:
  - El estado t_0 con el estado t_3. Por lo tanto, en el estado t_0 queda un bucle con la expresión $a \vee ca$ y las aristas entre el estado t_0 con los estados t_1 y t_6 tendrán la expresión b y cc respectivamente.
  - El estado t_1 con el estado t_4. Por lo tanto, en el estado t_1 queda un bucle con la expresión $a \vee ca$ y las aristas entre el estado t_1 con los estados t_2 y t_4 tendrán la expresión b y cc respectivamente.
  - El estado t_2 con el estado t_5. Por lo tanto, en el estado t_2 queda un bucle con la expresión $a \vee ca$ y la arista entre el estado t_2 y el estado t_{10} tendrá la expresión cc.

  Por lo tanto, el grafo nos queda de la siguiente manera:



  De acuerdo a este grafo, podemos considerar que se tienen tres expresiones que nos llevan desde el estado inicial t_0 hacia el estado de aceptación t_{11}:
  - La primera expresión α_1 esta dada por el transcurso entre los estados t_0, t_1, t_2 y t_{11}, es entonces, $(a \vee ca)^* b(a \vee ca)^* b(a \vee ca)^* cc(a \vee a^+c)^*$.
  - La segunda expresión α_2 esta dada por el transcurso entre los estados t_0, t_1, t_8 y t_{11}, es entonces, $(a \vee ca)^* b(a \vee ca)^* cc(a \vee a^+c)^* b(a \vee ca^*)^*$.
  - La tercera expresión α_3 esta dada por el transcurso entre los estados t_0, t_6, t_8 y t_{11}, es entonces, $(a \vee ca)^* cc(a \vee a^+c)^* b(a \vee ca)^* b(a \vee ca^*)^*$.

Por lo tanto la expresión regular de nuestro lenguaje $\mathcal{L}$ es $\alpha_1 \vee \alpha_2 \vee \alpha_3$:

$$(a \vee ca)^* b(a \vee ca)^* b(a \vee ca)^* cc(a \vee a^+c)^* \vee (a \vee ca)^* b(a \vee ca)^* cc(a \vee a^+c)^* b(a \vee ca^*)^* \vee (a \vee ca)^* cc(a \vee a^+c)^* b(a \vee ca)^* b(a \vee ca^*)^*$$

4 Función de censo

La matriz del multigrafo asociado a nuestro autómata es:

$$A = \begin{pmatrix} 1 & 1 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & {\color{red}0} & {\color{red}0} \\ 0 & 1 & 1 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 \\ 1 & 1 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 1 & 1 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 1 & 0 & 1 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 1 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 1 \end{pmatrix}$$

De acuerdo a la construcción de esta matriz, los estados de aceptación de nuestro autómata se evidencian en las columnas 11 y 12 de la primera fila (Los elementos que están de color rojo), y el valor de estos se entienden como la cantidad de palabras de cierta longitud (esta es determinada por la potencia de dicha matriz) que son aceptadas en nuestro lenguaje. Ahora, al realizar las primeras trece potencias de dicha matriz tenemos como resultado los siguientes valores.

n	$k(n)$	$l(n)$	$\lvert\mathcal{L}_n\rvert$
1	0	0	0
2	0	0	0
3	0	0	0
4	1	2	3
5	7	11	18
6	27	42	69
7	85	131	216
8	235	362	597
9	597	921	1518
10	1427	2206	3633
11	3257	5047	8304
12	7170	11136	18306
13	15330	23862	39192

Donde $k(n)$ indica la cantidad de palabras aceptadas por nuestro lenguaje de longitud n que terminan en c y $l(n)$ las que terminan en a o b. Por lo tanto $\lvert\mathcal{L}_n\rvert = k(n) + l(n)$.

Ahora, debemos encontrar la función de censo que representa nuestro lenguaje $\mathcal{L}$.

Para esto, primero debemos encontrar el polinomio asociado a nuestra matriz A por medio de la forma canónica de jordan. Para esto se desarrolla el $\det(A - \lambda I_{12})$, cuya matriz se puede evidenciar a continuación,

$$\det\begin{pmatrix} 1-\lambda & 1 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 1-\lambda & 1 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1-\lambda & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 \\ 1 & 1 & 0 & -\lambda & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & 1 & 0 & -\lambda & 0 & 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & -\lambda & 0 & 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & -\lambda & 1 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 1 & 1-\lambda & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1-\lambda & 1 & 0 & 1 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & -\lambda & 0 & 1 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & -\lambda & 1 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 1-\lambda \end{pmatrix}$$

Ahora, por ser una matriz 12x12, se acude al programa matrixcalc para realizar el procedimiento, teniendo como resultado:

$\det(A-\lambda I_{12}) = (\lambda^2-\lambda-1)^6$
$\det(A-\lambda I_{12}) = \lambda^{12}-6\lambda^{11}+9\lambda^{10}+10\lambda^9-30\lambda^8-6\lambda^7+41\lambda^6+6\lambda^5-30\lambda^4-10\lambda^3+9\lambda^2+6\lambda+1.$

Polinomio que de acuerdo a su estructura inicial $\lambda^2-\lambda-1$, se podría esperar que la función de censo, tenga una relación con la sucesión de Fibonacci.
A partir de este polinomio, se tiene la ley de recurrencia que representa nuestro lenguaje. Para esto, se realiza el cambio de variable de λ^n por $|\mathcal{L}_n|$ y resolviendo, se tiene:

$$|\mathcal{L}_{12}| = 6|\mathcal{L}_{11}|-9|\mathcal{L}_{10}|-10|\mathcal{L}_9|+30|\mathcal{L}_8|+6|\mathcal{L}_7|-41|\mathcal{L}_6|-6|\mathcal{L}_5|+30|\mathcal{L}_4|+10|\mathcal{L}_3|-9|\mathcal{L}_2|-6|\mathcal{L}_1|-1$$

Ahora, se procede a encontrar los valores propios de dicho polinomio de grado 12. Para esto se renombra a λ como x. Por la complejidad del proceso, se acude nuevamente al programa matrixcalc. Y de los 12 valores que se pueden esperan, se tienen solo 2, por lo cual, cada uno es de multiplicidad 6.

- Como primer valor propio se tiene a $x=\frac{1+\sqrt{5}}{2}$, y al ser de multiplicidad 6, entonces se tiene como expresión:
  $a(\frac{1+\sqrt{5}}{2})^n + b*n(\frac{1+\sqrt{5}}{2})^n + c*n^2(\frac{1+\sqrt{5}}{2})^n + d*n^3(\frac{1+\sqrt{5}}{2})^n + e*n^4(\frac{1+\sqrt{5}}{2})^n + f*n^5(\frac{1+\sqrt{5}}{2})^n$
  Expresión que al factorizar se tiene:
  $((\frac{1+\sqrt{5}}{2})^n)(a+b*n+c*n^2+d*n^3+e*n^4+f*n^5).$
- Como segundo valor propio se tiene a $x=\frac{1-\sqrt{5}}{2}$,y al ser de multiplicidad 6, entonces se tiene como expresión:
  $g(\frac{1-\sqrt{5}}{2})^n + h*n(\frac{1-\sqrt{5}}{2})^n + i*n^2(\frac{1-\sqrt{5}}{2})^n + j*n^3(\frac{1-\sqrt{5}}{2})^n + k*n^4(\frac{1-\sqrt{5}}{2})^n + l*n^5(\frac{1-\sqrt{5}}{2})^n$
  Expresión que al factorizar se tiene:
  $((\frac{1-\sqrt{5}}{2})^n)(g+h*n+i*n^2+j*n^3+k*n^4+l*n^5).$

Por lo tanto:

$|\mathcal{L}_n| = ((\frac{1+\sqrt{5}}{2})^n)(a+b*n+c*n^2+d*n^3+e*n^4+f*n^5) + ((\frac{1-\sqrt{5}}{2})^n)(g+h*n+i*n^2+j*n^3+k*n^4+l*n^5).$

Ahora, para completar la función de censo de nuestro lenguaje $\mathcal{L}$, debemos encontrar las constantes a,b,c,d,e,f,g,h,i,j,k,l.
Para este proceso, creamos un sistema de ecuaciones 12x12 con la ecuación $|\mathcal{L}_n|$ con valores de $n \geq 4$.
Al dar solución a dicho sistema de ecuaciones por medio del programa matrixcalc aplicando la regla de Cramer, se tiene como resultado:

- $a=\frac{-3\sqrt{5}}{125}$, $b=\frac{135+13\sqrt{5}}{500}$, $c=\frac{-45-9\sqrt{5}}{250}$, $d=\frac{3+\sqrt{5}}{100}$, $e=0$ y $f=0$.

- $g = \frac{3\sqrt{5}}{125}$, $h = \frac{135-13\sqrt{5}}{500}$, $i = \frac{-45+9\sqrt{5}}{250}$, $j = \frac{3-\sqrt{5}}{100}$, $k = 0$ y $l = 0$.

Por lo tanto, la función de censo que representa nuestro lenguaje $|\mathcal{L}_n|$ esta dada por:

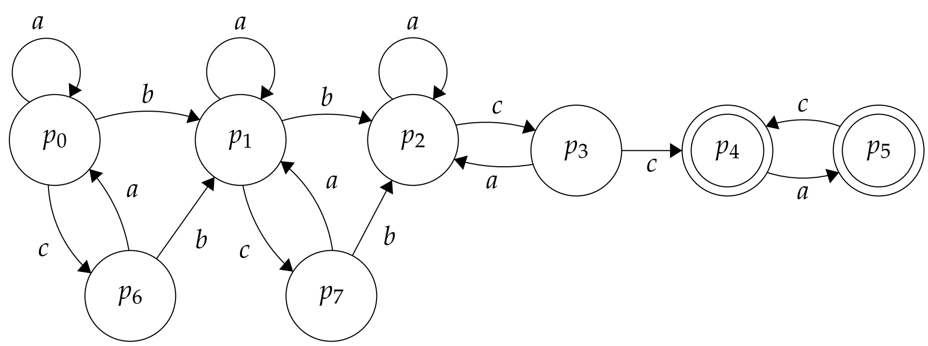
$|\mathcal{L}_n| = ((\frac{1+\sqrt{5}}{2})^n)(\frac{-3\sqrt{5}}{125} + \frac{135+13\sqrt{5}}{500} * n + \frac{-45-9\sqrt{5}}{250} * n^2 + \frac{3+\sqrt{5}}{100} * n^3) + ((\frac{1-\sqrt{5}}{2})^n)(\frac{3\sqrt{5}}{125} + \frac{135-13\sqrt{5}}{500} * n + \frac{-45+9\sqrt{5}}{250} * n^2 + \frac{3-\sqrt{5}}{100} * n^3)$

Con el fin de confirmar que el resultado obtenido anteriormente es correcto, se procede a realizar un análisis a partir de la expresión regular del lenguaje.
De acuerdo a la expresión regular de $\mathcal{L}$, se tiene que esta es la unión de otras tres expresiones:

- La expresión α_1 dada por $(a \vee ca)^* b(a \vee ca)^* b(a \vee ca)^* cc(a \vee a^+c)^*$
- La expresión α_2 dada por $(a \vee ca)^* b(a \vee ca)^* cc(a \vee a^+c)^* b(a \vee ca^*)^*$
- La expresión α_3 dada por $(a \vee ca)^* cc(a \vee a^+c)^* b(a \vee ca)^* b(a \vee ca^*)^*$

Expresiones que de acuerdo a su estructura se puede decir que son conjuntos de cadenas disjuntos, pues la ubicación de las dos b's y la subcadena cc varia en cada una de las expresiones. En α_1 las dos b's aparecen antes que la subcadena cc; α_2 tiene la subcadena cc entre las dos b's; y α_3 tiene las dos b's despues de la subcadena cc. Para una mejor evidencia de lo dicho anteriorente, se analizará cada una de estas expresiones.

- El autómata que representa a α_1, es el recorrido de la parte superior del autómata de nuestro lenguaje:



Observando el autómata, se puede evidenciar que las palabras aceptadas por α_1 solo terminan en a y en c. Ahora, para saber con exactitud la cantidad de palabras aceptadas, pasamos a crear la matriz de adyacencia.

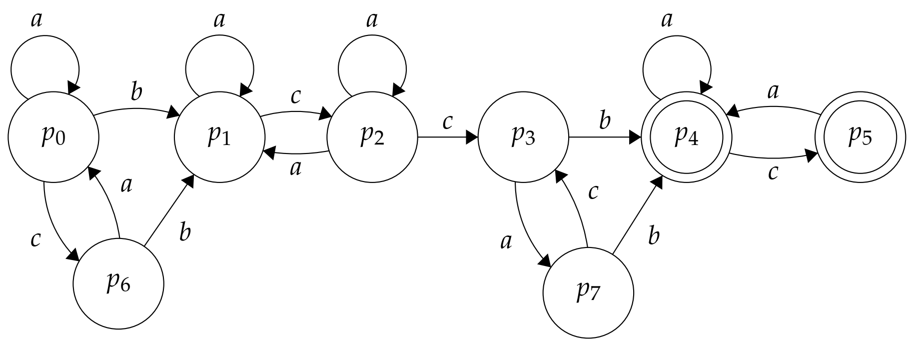
$$A = \begin{pmatrix} 1 & 1 & 0 & 0 & 0 & 0 & 1 & 0 \\ 0 & 1 & 1 & 0 & 0 & 0 & 0 & 1 \\ 0 & 0 & 1 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 1 & 0 & 0 \\ 1 & 1 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & 1 & 0 & 0 & 0 & 0 & 0 \end{pmatrix}$$

Donde las casillas de color rojo son las que determinan el número de palabras aceptadas. Ahora, realizando las primeras nueve potencias de esta matriz, se obtienen los siguientes resultados:

n	$e(n)$	$f(n)$	$\lvert\alpha_1\rvert$
1	0	0	0
2	0	0	0
3	0	0	0
4	1	0	1
5	5	1	6
6	17	6	23
7	49	23	72
8	127	72	199
9	307	199	506

Donde $e(n)$ son las palabras aceptadas que terminan en c y $f(n)$ las que terminan en a, por tanto, $|\alpha_1| = e(n) + f(n)$.

- El autómata que representa a α_2, es el recorrido de la parte central del autómata de nuestro lenguaje:



Observando el autómata, se puede evidenciar que las palabras aceptadas por α_2 terminan en a, b y c. Ahora, para saber con exactitud la cantidad de palabras aceptadas, pasamos a crear la matriz de adyacencia.

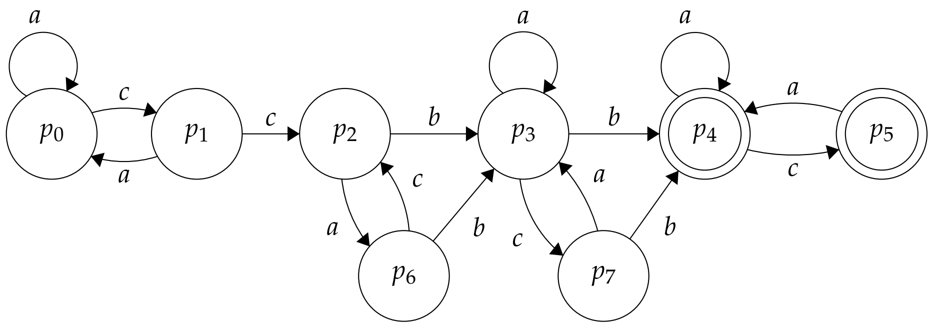
$$A = \begin{pmatrix} 1 & 1 & 0 & 0 & \color{red}{0} & \color{red}{0} & 1 & 0 \\ 0 & 1 & 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 & 0 & 1 \\ 0 & 0 & 0 & 0 & 1 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 \\ 1 & 1 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 1 & 0 & 0 & 1 \end{pmatrix}$$

Donde las casillas de color rojo son las que determinan el número de palabras aceptadas. Ahora, realizando las primeras nueve potencias de esta matriz, se obtienen los siguientes resultados:

n	$e(n)$	$f(n)$	$\lvert\alpha_2\rvert$
1	0	0	0
2	0	0	0
3	0	0	0
4	1	0	1
5	5	1	6
6	18	5	23
7	54	18	72
8	145	54	199
9	361	145	506

Donde $e(n)$ son las palabras aceptadas que terminan en a o b y $f(n)$ las que terminan en c, por tanto, $|\alpha_2| = e(n) + f(n)$.

- El autómata que representa a α_3, es el recorrido de la parte inferior del autómata de nuestro lenguaje:



  Observando el autómata, se puede evidenciar que las palabras aceptadas por α_3 terminan en a, b y c. Ahora, para saber con exactitud la cantidad de palabras aceptadas, pasamos a crear la matriz de adyacencia.

$$A = \begin{pmatrix} 1 & 1 & 0 & 0 & {\color{red}0} & {\color{red}0} & 0 & 0 \\ 1 & 0 & 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 & 1 & 0 & 0 & 1 \\ 0 & 0 & 0 & 0 & 1 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 1 & 1 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 & 1 & 0 & 0 & 0 \end{pmatrix}$$

  Donde las casillas de color rojo son las que determinan el número de palabras aceptadas. Ahora, realizando las primeras nueve potencias de esta matriz, se obtienen los siguientes resultados:

n	$e(n)$	$f(n)$	$\|\alpha_3\|$
1	0	0	0
2	0	0	0
3	0	0	0
4	1	0	1
5	5	1	6
6	18	5	23
7	54	18	72
8	145	54	199
9	361	145	506

  Donde $e(n)$ son las palabras aceptadas que terminan en a o b y $f(n)$ las que terminan en c, por tanto, $|\alpha_3| = e(n) + f(n)$.

A partir de este análisis y para verificar si realmente estos tres conjuntos son disjuntos, la suma de la cantidad de palabras que terminan en a, b o c de cualquier longitud y que son aceptadas, debe coincidir con las palabras aceptadas en $|\mathcal{L}_n|$, vamos a comprobarlo:

- Sea $n = 6$, la cantidad palabras que terminan en a o b en $|\alpha_1| = 6$, en $|\alpha_2| = 18$ y en $|\alpha_3| = 18$ por tanto $|\alpha_1| + |\alpha_2| + |\alpha_3| = 42$ y de acuerdo a la tabla de $|\mathcal{L}_n|$, estas palabras están dadas por $l(n)$ y se tiene que $l(6) = 42$ (si se desea se puede continuar con este cálculo). Por tanto, las palabras aceptadas por $|\alpha_1|$, $|\alpha_2|$ y $|\alpha_3|$ que terminan en a o b son distintas.

- Sea $n = 6$, la cantidad palabras que terminan en c en $|\alpha_1| = 17$, en $|\alpha_2| = 5$ y en $|\alpha_3| = 5$ por tanto $|\alpha_1| + |\alpha_2| + |\alpha_3| = 27$ y de acuerdo a la tabla de $|\mathcal{L}_n|$, estas palabras están dadas por $k(n)$ y se tiene que $k(6) = 27$ (si se desea se puede continuar con este cálculo). Por tanto, las palabras aceptadas por

$|\alpha_1|$, $|\alpha_2|$ y $|\alpha_3|$ que terminan en c son distintas.

Después de verificada la disyunción de las palabras aceptadas por cada una de las tres expresiones. Podemos empezar a analizar, que aunque estas expresiones no comparten ninguna palabra, la cantidad de palabras aceptadas es la misma, pues $|\alpha_1| = |\alpha_2| = |\alpha_3|$ y por tanto $|\mathcal{L}_n| = 3|\alpha_1| = 3|\alpha_2| = 3|\alpha_3|$.

Ahora, a partir de esta relación, pasamos a encontrar el polinomio asociado y sus valores propios, para cada una de las matrices de α_1, α_2 y α_3 con el fin de evidenciar si la relación entre estas tres expresiones es aún más estrecha (este proceso, se realizará con ayuda del programa matrixcalc).

- Para la matriz de α_1, se tiene que su polinomio asociado es $x^8 - 4x^7 + 2x^6 + 8x^5 - 5x^4 - 8x^3 + 2x^2 + 4x + 1$ y como valores propios de este, se tiene $x = \frac{1+\sqrt{5}}{2}$ y $x = \frac{1-\sqrt{5}}{2}$, por lo cual, cada valor propio es de multiplicidad 4.

- Para la matriz de α_2, se tiene que su polinomio asociado es $x^8 - 4x^7 + 2x^6 + 8x^5 - 5x^4 - 8x^3 + 2x^2 + 4x + 1$ y como valores propios de este, se tiene $x = \frac{1+\sqrt{5}}{2}$ y $x = \frac{1-\sqrt{5}}{2}$, por lo cual, cada valor propio es de multiplicidad 4.

- Para la matriz de α_3, se tiene que su polinomio asociado es $x^8 - 4x^7 + 2x^6 + 8x^5 - 5x^4 - 8x^3 + 2x^2 + 4x + 1$ y como valores propios de este, se tiene $x = \frac{1+\sqrt{5}}{2}$ y $x = \frac{1-\sqrt{5}}{2}$, por lo cual cada valor propio es de multiplicidad 4.

Realizado este proceso, se puede observar que las tres expresiones cuentan con el mismo polinomio asociado y por ende con los mismos valores propios.
Por otra parte, se puede observar que los valores propios obtenidos coinciden con los valores propios del polinomio asociado a nuestro lenguaje $\mathcal{L}$.
Por esto último, se decide indagar sobre la relación entre el polinomio asociado del lenguaje $\mathcal{L}$ el cual es de grado 12 y el polinomio asociado de las expresiones α_1, α_2 y α_3 que es de grado 8.

En un primer momento, se tiene como polinomio asociado de $\mathcal{L}$ la expresión $(\lambda^2 - \lambda - 1)^6$, por la cual, se tuvo la idea de que tendría una relación con la sucesión de Fibonacci. Es así, como se empieza a indagar sobre la estructura de las expresiones de α_1, α_2 y α_3, con el fin de encontrar la relación con esta misma sucesión.
Se observa que cada una de las expresiones cuenta con cuatro bucles, cuya estructura esta asociada con la sucesión de Fibonacci. Estos cuatro, se dividen en tres con la estructura $(a \vee ca)^*$ y uno con la estructura $(a \vee ac)^*$, por lo cual, se decide acudir al programa SAGE para realizar el cálculo de $(x^2 - x - 1)^4$, teniendo como resultado:

[1]:
```
var('x')
expand((x^2-x-1)^4)
```

[1]:
```
x^8 - 4*x^7 + 2*x^6 + 8*x^5 - 5*x^4 - 8*x^3 + 2*x^2 + 4*x + 1
```

Polinomio que coincide exactamente con el polinomio asociado de las tres expresiones. Por lo cual, se puede confirmar la relación entre la sucesión de Fibonacci con la sucesión de las tres expresiones.
Ahora, se decide formar un sistema de ecuaciones 8x8, donde cada ecuación tiene la forma:

$a(\frac{1+\sqrt{5}}{2})^n + b*n(\frac{1+\sqrt{5}}{2})^n + c*n^2(\frac{1+\sqrt{5}}{2})^n + d*n^3(\frac{1+\sqrt{5}}{2})^n + e*(\frac{1-\sqrt{5}}{2})^n + f*n(\frac{1-\sqrt{5}}{2})^n + g*n^2(\frac{1-\sqrt{5}}{2})^n + h*n^3(\frac{1-\sqrt{5}}{2})^n$

Con el fin de encontrar las constantes a, b, c, d, e, f, g y h. Por la relación que ya se evidenciado entre nuestro lenguaje y las expresiones, se espera que estos valores al ser multiplicados por 3 sean los valores obtenidos en un principio para nuestro lenguaje en general.
Para el desarrollo de este sistema de ecuaciones se acude nuevamente al programa Matrixcalc, donde se tiene como resultado:
$a = (-\sqrt{5})/125$
$b = (13 * \sqrt{5} + 135)/1500$
$c = (-3 * \sqrt{5} - 15)/250$
$d = (\sqrt{5} + 3)/300$
$e = \sqrt{5}/125$
$f = (-13 * \sqrt{5} + 135)/1500$
$g = (3 * \sqrt{5} - 15)/250$
$h = (-\sqrt{5} + 3)/300$
Ahora, si multiplicamos por 3 cada uno de estos resultados, tenemos como respuesta:
$a = (-3\sqrt{5})/125$
$b = (13 * \sqrt{5} + 135)/500$
$c = (-9 * \sqrt{5} - 45)/250$
$d = (\sqrt{5} + 3)/100$
$e = 3\sqrt{5}/125$
$f = (-13 * \sqrt{5} + 135)/500$
$g = (9 * \sqrt{5} - 45)/250$
$h = (-\sqrt{5} + 3)/100$
Valores, que coinciden con los que encontramos al inicio del análisis de la función de censo de nuestro lenguaje. Por tal motivo, la función de censo de nuestro lenguaje $\mathcal{L}$ es:

$|\mathcal{L}_n| = ((\frac{1+\sqrt{5}}{2})^n)(\frac{-3\sqrt{5}}{125} + \frac{135+13\sqrt{5}}{500} * n + \frac{-45-9\sqrt{5}}{250} * n^2 + \frac{3+\sqrt{5}}{100} * n^3) + ((\frac{1-\sqrt{5}}{2})^n)(\frac{3\sqrt{5}}{125} + \frac{135-13\sqrt{5}}{500} * n + \frac{-45+9\sqrt{5}}{250} * n^2 + \frac{3-\sqrt{5}}{100} * n^3)$

5 Simulación en SAGE

Para trabajar en este simulador nuestro lenguaje, primero debemos cambiar nuestro alfabeto de $\{a, b, c\}$ al alfabeto $\{0, 1, 2\}$ respectivamente. Por lo cual, se tiene que:

[1]:
```
ALF=range(3)
```

Ahora debemos introducir nuestro autómata en el simulador, de la siguiente manera:

[2]:
```
aut=[[0,1,3], [1,2,4], [2,12,5], [0,1,6], [1,2,9], [2,12,10], [7,8,12],[7,8,6],␣
↪[8,11,9],[8,11,12], [11,12,12], [11,12,10], [12,12,12]]
```

Donde cada tripla representa un estado de nuestro autómata, así mismo, cada elemento de esta tripla identifica el estado al que debe pasar dependiendo del elemento del alfabeto que tenga la cadena a leer, por ejemplo:

La tripla $[2, 12, 5]$ representa el estado t_2 de nuestro autómata ya que si estando en el estado t_2, se lee una $a = 0$, se queda en el estado t_2, si lee una $b = 1$, pasa al estado t_{12} y si lee una $c = 2$, pasa al estado t_5.

Ahora debemos definir los procedimientos, que permitirán que el simulador actúe como lo hace nuestro autómata.

- Primero, definimos la expresión delta. Por medio de esta se determina la transición de nuestro autómata.

[3]:
```
def delta(auto,qi,a):
    return aut[qi][a]
```

- Luego, definimos la expresión deltat, la cual, reproducirá la expresión anterior la cantidad de veces necesarias dependiendo de la longitud de la cadena ω que se quiera analizar.

[4]:
```
def deltat(aut,pal):
    estado=0
    for jj in pal:
        estado=delta(aut,estado,jj)
    return estado
```

Ahora, un ejemplo:
Si queremos analizar la cadena *abaccbac* en este simulador seria 01022102 y para que corra el programa se introduce como:

[5]:
```
deltat(aut,(0,1,0,2,2,1,0,2))
```

[5]:
```
10
```

Donde se tiene como resultado 10, pues hace referencia al estado t_{10} de nuestro autómata, el cual es uno de los estados de aceptación, más exactamente de las cadenas que cumplen las condiciones y terminan en c que para este simulador es 2.

Si queremos conocer mediante este simulador, la cantidad de cadenas ω de diferentes longitudes que son aceptadas por nuestro autómata, debemos emplear el siguiente comando:

[6]:
```
LO=cartesian_product([ALF,ALF,ALF,ALF,ALF,ALF,ALF,ALF,ALF,ALF])
```

[7]:
```
hh=[uu for uu in LO if deltat(aut,uu)==10]
vv=[uu for uu in LO if deltat(aut,uu)==11]
show(len(hh))
show(len(vv))
```

[7]:
```
1427
```

[7]:
```
2206
```

Donde, LO determina la longitud de las cadenas ω, en este caso como se tienen diez ALF, entonces son las cadenas de longitud 10. hh determina la cantidad de palabras que llegan al estado de aceptación t_{10} y vv determina la cantidad de palabras que llegan al estado de aceptación t_{11}. Ahora, como queremos saber cual es el número exacto de hh y vv, se emplea el comando show(len(hh)) y show(len(vv)), donde se tiene como resultado 1427 y 2206 respectivamente. Así, la cantidad de palabras aceptadas por el autómata de longitud 10 es $3633 = |\mathcal{L}_{10}|$.

De acuerdo a lo trabajado en la sección titulada Función de censo, se tiene que la sucesión de la cantidad de cadenas aceptadas de diferentes longitudes esta dada por $|\mathcal{L}_n| = ((\frac{1+\sqrt{5}}{2})^n)(\frac{-3\sqrt{5}}{125} + \frac{135+13\sqrt{5}}{500} * n + \frac{-45-9\sqrt{5}}{250} * n^2 + \frac{3+\sqrt{5}}{100} * n^3) + ((\frac{1-\sqrt{5}}{2})^n)(\frac{3\sqrt{5}}{125} + \frac{135-13\sqrt{5}}{500} * n + \frac{-45+9\sqrt{5}}{250} * n^2 + \frac{3-\sqrt{5}}{100} * n^3)$. Ahora, para evidenciar si esta es correcta, se introdujo en SAGE, con el siguiente comando:

[8]:
```
var('n')
```

```
a=((((1+sqrt(5))/2)^n)*((-3*sqrt(5)/125)+(((135+13*sqrt(5))/
↪500)*(n))+(((-45-9*sqrt(5))/250)*(n^2))+(((3+sqrt(5))/100)*(n^3))))+((((1-sqrt(5))/
↪2)^n)*((3*sqrt(5)/125)+(((135-13*sqrt(5))/500)*n)+(((-45+9*sqrt(5))/
↪250)*n^2)+(((3-sqrt(5))/100)*n^3)))
```

Y con el fin de obtener los resultados para los diferentes valores de n entonces se tiene el siguiente comando:

- Para $n = 4$:

[9]:
```
show(a(4))
```

[9]:
```
3/2000*(6*sqrt(5) + 5)*(sqrt(5) + 1)^4 - 3/2000*(6*sqrt(5) - 5)*(sqrt(5) - 1)^4
```

Expresión que al realizarla con la calculadora, se tiene como resultado 3.

- Para $n = 11$:

[10]:
```
show(a(11))
```

[10]:
```
3/16000*(24*sqrt(5) + 55)*(sqrt(5) + 1)^11 + 3/16000*(24*sqrt(5) - 55)*(sqrt(5)
- 1)^11
```

Expresión que al realizarla con la calculadora, se tiene como resultado 8304.

- Para $n = 18$:

[11]:
```
show(a(18))
```

[11]:
```
3/2621440*(157*sqrt(5) + 405)*(sqrt(5) + 1)^18 - 3/2621440*(157*sqrt(5) -
405)*(sqrt(5) - 1)^18
```

Expresión que al realizarla con la calculadora, se tiene como resultado 1310559.

De esta manera confirmamos que el análisis realizado en dicha sección es correcto.

6 Enciclopedia OEIS

Como la sucesión que representa nuestro lenguaje no se encuentra dentro de esta enciclopedia OEIS. Entonces, se plantea una posible nueva sucesión dada por:

0,0,0,3,18,69,216,597,1518,3633,8304,18306,39191,..

La cual se obtiene por medio de la siguiente expresión para valores $n \geq 1$:

$$a_n = ((\tfrac{1+\sqrt{5}}{2})^n)(\tfrac{-3\sqrt{5}}{125} + \tfrac{135+13\sqrt{5}}{500} * n + \tfrac{-45-9\sqrt{5}}{250} * n^2 + \tfrac{3+\sqrt{5}}{100} * n^3) + ((\tfrac{1-\sqrt{5}}{2})^n)(\tfrac{3\sqrt{5}}{125} + \tfrac{135-13\sqrt{5}}{500} * n + \tfrac{-45+9\sqrt{5}}{250} * n^2 + \tfrac{3-\sqrt{5}}{100} * n^3)$$

Donde:

$a(n)$ es el número de palabras ω de longitud n sobre el alfabeto $\{0,1,2\}$ talque ω contiene exactamente dos 1's y exactamente una vez la subpalabra 22.

$a(n)$ is the number of words ω of length n over the alphabet $\{0,1,2\}$ such that ω has two 1's and exactly once the subword 22.

7 Bibliografía

De Castro, Rodrigo. (2004). Teoría de la computación. Lenguajes, autómatas, gramáticas. Bogotá, Colombia. Unilibros.
Isaacs, Rafael. (2020). Función de Censo para algunos lenguajes regulares.
Zill, Dennis. (2009). A first course in Differential Equations with modeling applications.