

Trabajo Práctico N° 1

1)

$$L_1 = \{cada, edad, cede\}$$

$$L_2 = \{CAFE, CEDA, FEDE\}$$

$$L_3 = \{21, 22, 32\}$$

- $L_1 \cup L_3 = \{cada, edad, cede, 21, 22, 32\}$
- $L_2 \cap L_3 = \{\}$
- Teniendo $U = \{cada, edad, cede, CAFE, CEDA, FEDE, 21, 22, 32\}$ entonces:
 $\tilde{L}_1 = \{CAFE, CEDA, FEDE, 21, 22, 32\}$
- $L_1 \cdot L_2 \cdot L_3 = \{cadaCAFE21, cadaCAFE22, cadaCAFE32, cadaCEDA21, cadaCEDA22, cadaCEDA32, cadaFEDE21, cadaFEDE22, cadaFEDE32, edadCAFE21, edadCAFE22, edadCAFE32, edadCEDA21, edadCEDA22, edadCEDA32, edadFEDE21, edadFEDE22, edadFEDE32, cedeCAFE21, cedeCAFE22, cedeCAFE32, cedeCEDA21, cedeCEDA22, cedeCEDA32, cedeFEDE21, cedeFEDE22, cedeFEDE32\}$

2)




- $L_1^* = \{abc, aabbcc, aaabbbccc, \varepsilon\}$
- $L_1^+ = \{abc, aabbcc, aaabbbccc, \dots\}$
- $L_3 = \{abcabc, abcaabbcc, abcaabbbccc, aabbccabc, aabbccaabbcc, \text{Debería ser igual a } L_1^+, aabbccaaabbbccc, aaabbbcccabc, aaabbbcccabbcc, aaabbbcccabbcc\}$
- $\emptyset^+ = \{\}$
- $\emptyset^* = \{\varepsilon\}$
- $L_1^* \cdot \emptyset = \emptyset$

3)



- Strings que pertenecen: $\{ab, aab, abb\}$
 Strings que no pertenecen: $\{ba, baa, bba\}$
- Strings que pertenecen: $\{abab, ababab, abababab\}$
 Strings que no pertenecen: $\{baa, aaba, bbabb\}$
- Strings que pertenecen: $\{aaa, bbb, aaaaa\}$
 Strings que no pertenecen: $\{ab, baab, babab\}$
- Strings que pertenecen: $\{aaa, aaaaaa, aaaaaaaaa\}$
 Strings que no pertenecen: $\{a, aa, aaaa\}$
- Strings que pertenecen: $\{aba, aaba, aabbaa\}$
 Strings que no pertenecen: $\{aaa, bbb, aaabbb\}$
- Strings que pertenecen: $\{aba, bab\}$
 Strings que no pertenecen: $\{aaa, bbbb, abbbb\}$
- Strings que pertenecen: $\{b, ab\}$
 Strings que no pertenecen: $\{a, bb, bbb\}$
- Strings que pertenecen: $\{aaaaa, babbb, bbaaa\}$
 Strings que no pertenecen: $\{\varepsilon\}$



4)

- a) $1\Sigma^*0$
- b) $\Sigma^*1\Sigma^*1\Sigma^*1\Sigma^*$
- c) $\Sigma^*0101\Sigma^*$
- d) $\Sigma\Sigma\Sigma\Sigma\Sigma$ 
- e) $(1\Sigma)^*$ 
- f) $1^*11^*01^*11^*$ 
- g) $(01 \cup 1 \cup 1) \Sigma^*$
- h) $1\Sigma^*1\Sigma^*1\Sigma^*110$
- i) $\Sigma^*(00 \cup 1)$

5)

- a) $(abb)^*$ 
- b) $(\Sigma\Sigma)^* \cup (\Sigma\Sigma\Sigma)^*$
- c) $a^*(aba)a^*$ 

6)

- a) Lenguaje infinito. $\{w / w \text{ empieza con } xy \text{ seguido de una cantidad nula o infinita de } z\}$
- b) Lenguaje infinito. $\{w / w \text{ empieza con una cantidad nula o infinita de } x, \text{ seguido de } yz\}$
- c) Lenguaje finito. $\{w / w \text{ empieza con } z \text{ o } y, \text{ seguido de una } x\}$
- d) Lenguaje infinito. $\{w / w \text{ tiene una cantidad nula o infinita de } z \text{ y de } y\}$
- e) Lenguaje infinito. $\{w / w \text{ tiene una cantidad nula o par de } y\}$
- f) Lenguaje infinito. $\{w / w \text{ tiene una cantidad nula o infinita de } x \text{ o una cantidad nula o infinita de } y\}$
- g) Lenguaje finito. $\{w / w \text{ es } xx \text{ o es } z\}$
- h) Lenguaje finito $\{w / w \text{ es } x, \text{ es } y \text{ o es } z\}$

7)

- a) $\{w / w \text{ es empieza con una cantidad nula o infinita de } z \text{ y de } y, \text{ y termina siempre con } x\}$
- b) $\{w / w \text{ empieza con } x, \text{ seguido de una cantidad nula o infinita de } x, \text{ seguido por una } y, \text{ terminando con una cantidad nula o infinita de } y\}$
- c) $\{w / w \text{ empieza con } x, \text{ seguido de una cantidad nula o infinita de } x; \text{ o empieza con } y, \text{ seguido de una cantidad nula o infinita de } y\}$
- d) $\{w / w \text{ empieza con una cantidad nula o infinita de } x, \text{ seguido por una cantidad nula o infinita de } y, \text{ terminando con una cantidad nula o infinita de } z\}$

8)

pertenecen = $\{aa, aaaa, aaaaaa\}$
no pertenecen = $\{abbab, abb, aaabbbba\}$

- a) $((\Sigma)^*a \cup (\Sigma\Sigma)^*)$

9)

- a) Es ambigua, ya que podemos obtener el mismo string "a" de dos maneras:
- i) A partir de $a((ab)^*cd)^*$ simplemente repetimos cero veces lo que esta dentro del paréntesis.
 - ii) A partir de $a(ababcb^*)^*a^*$ también lo obtenemos repitiendo cero veces lo que está dentro del paréntesis y la a del final.
- b) Es ambigua, ya que podemos obtener el string "aabb" de dos maneras:
- i) A partir de $aab^*(ab)^*$.
 - ii) A partir de a^*bba^* .
- c) No es ambigua, ya que es imposible obtener el mismo string a partir de ninguna de las cuatro expresiones. En la primera siempre obtenemos un "aab" seguido de nula o infinita cantidad de a; en la segunda solo podemos obtener "aaaba"; en la tercera obtenemos una cadena "aabb" seguida de nula o infinitas a; y en la última expresión únicamente podemos obtener "a".