ATIVIDADE EXTRA-CLASSE

GABARITO

A.) Dadas as ER's abaixo, resolva conforme enunciado da questão.

1-) Descreva em português a linguagem definida para cada expressão regular abaixo:

a) aa*
{a, aa, aaa, aaaa, aaaaa, ...}
➔ todas as seqüências de zero ou mais a's; ou qualquer seqüência de a^n com n>0.

b) aa^+
{aa, aaa, aaaa, aaaaa, ...}
➔ todas as seqüências de um ou mais a's; ou qualquer seqüência de a^n com n>1.

c) 0*1*2
{2, 02, 12, 012, 002, ...}
➔ todas as seqüências de números {0, 1, 2}, com final 0^n1^m2 com n,m >= 0.

d) (((a*a)b) | b)
{b, ab, aab, aaab, aaaab, ...}
➔ todas as seqüências de zero ou mais a's seguido de um único b; ou qualquer seqüência de a^nb com n >=0.

e) ((((a*b*)*ab) | ((a*b*)*ba))(b | a)*)
{ab, aab, bab, abb, aabb, babb, aba, aaba, baba, ba, aba, bba, bab, abab, bbab, ...}
➔ toda seqüência de a's e b's com no mínimo uma ocorrência de ab ou de ba.

2-) Seja ∑ = {a, b}. Escreva expressões regulares para os seguintes conjuntos:

a) Todas as strings em ∑* iniciando com a e finalizando com b.
a(a | b)*b ➔ garantimos a no início e b no final, com qualquer seqüência de a's ou b's.

b) Todas as strings em ∑* cujo número de a's é divisível por 3.
$(b^*ab^*ab^*a)^+b^*$ ➔ garantimos assim que haja pelo menos 3 a's na seqüência e também qualquer número de b's, já que não foi feita nenhuma restrição.

c) Todas as strings em ∑* com não mais de 3 b's.
b*(a | ε)b*(a | ε)b*(a | ε)b* ➔ garantimos que teremos não mais de 3 a's, já que para cada conjunto (a | ε), devemos

escolher ou um a, ou uma palavra vazia, com qualquer seqüência de b's.

d) Todas as strings em ∑* com exatamente uma ocorrência da substring aaa.

(ab | aab | b)*aaa(ba | baa | b)* ➔ garantimos que teremos a substring aaa, além de outras seqüências possíveis de ab ou ba.

e) Todas as strings em ∑* que contenham pares de a ou pares de b.

((b*ab*ab*) | (a*ba*ba*))$^+$ ➔ garantimos que teremos pelo menos um par de a's ou pelo menos um par de b's na solução.

3-) Quais das seguintes afirmações abaixo é verdadeira? Prove:

a) baa ∈ a*b*a*b*

Verdadeira. Para obtermos baa, devemos ter: nenhum a, um b, dois a's e nenhum b; chegamos à conclusão então que baa ∈ a*b*a*b*.

b) b*a* ∩ a*b* = a*∪b*

Verdadeira. Podemos provar que dois conjuntos de X e Y são iguais mostrando que qualquer string em X deve também estar em Y e vice-versa. Devemos mostrar que qualquer string em b*a* ∩ a*b* (que chamamos de X) deve também estar em a*∪b* (que chamamos de Y). Qualquer string em X deve ter duas propriedades (de b*a*): todos os b's vem antes de todos os a's; e (de a*b*): todos os a's vem antes de todos os b's. O único modo de termos estas propriedades simultaneamen-te é o mesmo ser composto de somente a's ou somente b's. Que é exatamente o que está em Y. Depois disso, devemos mostrar que cada string em Y está em X. Cada string em Y é ou da forma a* ou b*. Todas as strings da forma a* estão em X desde que simplesmente peguemos b* como ε, o qual nos dá a*∩a* = a*. Da mesma forma, se pegarmos todas as strings da forma b* estão em X desde que simplesmente peguemos a* como ε, o qual nos dá b*∩b* = b*, validando então a afirmação.

c) a*b* ∩ c*d* = ∅

Falso. Provamos isso facilmente identificando que ε ∈ a*b*, assim como ε ∈ c*d* e a*b*∩c*d* = ε, e sabemos por definição que ε ≠ ∅.

d) abcd $\in$ (a(cd)*b)*
Falso. Podemos notar que nunca conseguiremos gerar a palavra abcd, já que cada string cd é imediatamente precedida por a.

4-) Coloque V ou F para cada uma das expressões regulares abaixo:
a) $(ab)^*a = a(ba)^*$ **(V)**
b) $(a \cup b)^* b(a \cup b)^* = a^*b(a \cup b)^*$ **(V)**
c) $[(a \cup b)^* b(a \cup b)^* \cup (a \cup b)^* a(a \cup b)^*] = (a \cup b)^*$ **(F)**
d) $[(a \cup b)^* b(a \cup b)^* \cup (a \cup b)^* a(a \cup b)^*] = (a \cup b)^+$ **(V)**
e) $[(a \cup b)^* ba(a \cup b)^* \cup a^*b^*] = (a \cup b)^*$ **(V)**

5-) Escreva um expressão regular que gere número inteiros ímpares sem zeros à esquerda.
(ε | ((1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9) (0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9)*) (1 | 3 | 5 | 7 | 9))

6-) Escreva uma expressão regular que gere números binários quaisquer, sendo obrigatório pelo menos 1 dígito.
$(0 \mid 1)^+$

B.) Dadas as Gramáticas abaixo, resolva conforme enunciado da questão.

1-) Dada a gramática G = (V, T, P, X) onde:

V = {X}
T = {a, b}
P = {X→aX, X→b}

a) A palavra *abb* é gerada pela gramática G?
➔ Não é gerada.
b) A palavra *aba* é gerada pela gramática G?
➔ Não é gerada.
c) A palavra *ba* é gerada pela gramática G?
➔ Não é gerada
d) A palavra *aaab* é gerada pela gramática G?
➔ É gerada.
X $\Rightarrow^4$ aaab (X → aX → aaX → aaaX → aaab)
e) A palavra *aaaab* é gerada pela gramática G?
➔ É gerada.
X $\Rightarrow^5$ aaaab (X → aX → aaX → aaaX → aaaaX → aaaab)

2-) Dada a gramática G = (V, T, P, A) onde:

V = {A, B}
T = {0, 1}
P = {A→0A, A→B, B→1B, B→1}

a) A palavra *010101* é gerada pela gramática G?
➔ Não é gerada.
b) A palavra *00110* é gerada pela gramática G?
➔ Não é gerada.
c) A palavra *110* é gerada pela gramática G?
➔ Não é gerada.
d) A palavra *00111* é gerada pela gramática G?
➔ É gerada.
A $\Rightarrow^6$ 00111 (A → 0A → 00A → 00B → 001B → 0011B → 00111)

3-) Seja a gramática G = (V, T, P, S) onde:

V = {S, B, C}
T = {a, b, c}
P = {S→aSBC, S→aBC, CB→BC, aB→ab, bB→bb, bC→bc, cC→cc}

Apresente uma derivação para as palavras:
a) aabbcc
➔ É gerada.
S $\Rightarrow^6$ aabbcc (S → aSBC → aaBCBC → aaBBCC → aabBCC → aabbCC → aabbcc)
b) abbc
➔ Não é gerada.

4-) Dada a gramática G = (V, T, P, S) onde:
V = {S, B, C, D}
T = {0, 1}
P = {S→0B, S→1C, S→0C, B→0S, B→1D, B→1B,
B→ε, C→1S, C→0D, C→ε, D→0C, D→1B}

Apresente uma derivação para as palavras:

a) 0111

É gerada.

➔ S $\Rightarrow^5$ 0111 (S → 0B → 01B → 011B → 0111B → 0111ε)

b) 1101

É gerada.

➔ S $\Rightarrow^5$ 1101 (S → 1C → 11S → 110B → 1101B → 1101ε)

c) 01110

É gerada.

➔ S $\Rightarrow^6$ 01110 (S → 0B → 01B → 011B → 0111D → 01110C → 01110ε)

d) 10011

É gerada.

➔ S $\Rightarrow^6$ 10011 (S → 1C → 10D → 100C → 1001S → 10011C → 10011ε)

5-) Dada da gramática G=(V, T, P, INT) onde:
V = {DIG, INT}
T = {+, –, 0, 1, 2, 3, 4, 5, 6, 7, 8, 9}
P = {INT→ +DIG | –DIG, DIG→0DIG | 1DIG | ... | 9DIG | 0 | 1 | ... | 9}

a) A palavra *0 + 1* é gerada pela gramática G?

Não é gerada.

b) A palavra *– 0 + 1* é gerada pela gramática G?

Não é gerada.

c) A palavra *– 101* é gerada pela gramática G?

É gerada.

➔ INT $\Rightarrow^4$ –101 (INT → –DIG → –1DIG → –10DIG → –101).

6-) Gere uma Gramática G, tal que tenhamos números pares de a validados.

➔ G=(V, T, P, S) onde:
V = {S}
T = {a}
P = {S → aaS | ε}

7-) Gere uma Gramática G, tal que tenhamos números 0 e 1 consecutivos: 01, 0011, 000111, ..., validados.

➔ G=(V, T, P, S) onde:
V = {S, Z, U}
T = {0, 1}

P = {S → 0Z, Z → 0Z | 0U, U → 1U | 1}

8-) Gere uma Gramática G, tal que tenhamos os pares (a^nb^{n-1}), ou seja, aε, ab, aab, aaabb, aaaabbb, ..., validados.

➔ G=(V, T, P, S) onde:

V = {S}

T = {a, b}

P = {S → aSb, Sb → ab | ε}

9-) Gere uma Gramática G, tal que tenhamos uma palavra que seja identificador do C++ validada, ou seja, palavras formadas por uma ou mais letras e dígitos, sempre iniciando com uma letra.

➔ G=(V, T, P, S) onde:

V = {S, R, L, D}

T = {a, b, c, ..., z, _, 0, 1, 2, ..., 9}

P = {S → L | LR, R → L | D, L → a | b | c | ... | z | aR | bR | cR | ... | zR,

D → 0 | 1 | 2 | ... | 9 | _ | 0R | 1R | 2R | ... | 9R | _R}

10-) Gere uma Gramática G, tal que tenhamos um endereço de e-mail validado, ou seja, x@x, onde @ ocorre apenas uma vez.

➔ G=(V, T, P, S) onde:

V = {S, L}

T = {a, b, c, ..., z, @}

P = {S → L@L, L → a | b | c | ... | z | aL | bL | cL | ... | zL}

11-) Classifique as gramáticas dos exercícios 1 a 5 segundo a hierarquia de Chomsky.

➔ Exercício 1 → Gramática Regular.

➔ Exercício 2 → Gramática Livre de Contexto.

➔ Exercício 3 → Gramática Sensível ao Contexto.

➔ Exercício 4 → Gramática Irregular.

➔ Exercício 5 → Gramática Regular.

12-) Gere uma Gramática Regular G_R, tal que tenhamos um número real negativo ou positivo validado, sendo que apenas o símbolo negativo deve estar representado.

➔ G=(V, T, P, S) onde:

V = {S, D, P}

T = {–, 0, 1, 2, 3, 4, 5, 6, 7, 8, 9}

P = {S →–D | 0 | 1 | 2 | ... | 9 | 0D | 1D | 2D | ... | 9D,

D → 0 | 1 | 2 | ... | 9 | 0D | 1D | 2D | ... | 9D | 0P | 1P | 2P | ... | 9P,

P → .D}