

# CUADERNO de EXAMEN

Tercer Parcial de EIF 203 I-2022



## Prueba

Examen corto ☐

Parcial ☒

Final ☐

Reposición ☐

Otro: \_\_\_\_\_

Estudiante Diego Quirós Artiriano Diego

Carné 315071 / cédula: 901150326

Curso EIF203 I-2022

Grupo 41712

Profesor Carlos Lorra

Fecha 30 de junio, 2022

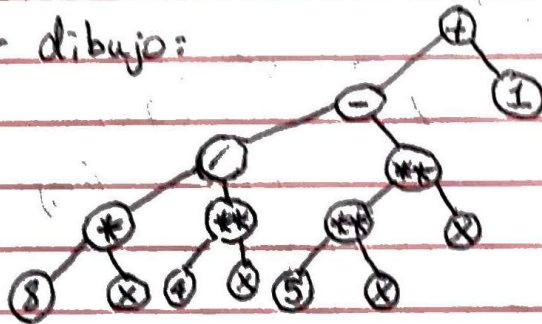
Calificación \_\_\_\_\_

**UNA LIBRERÍA UNIVERSITARIA**  
**Editorial Universidad Nacional**

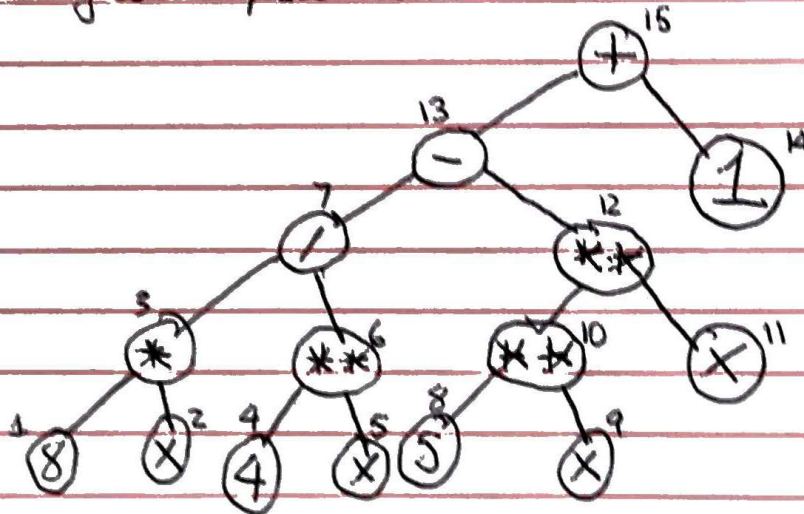
Diego Quiros Artirano Diego Pregunta 1

Expresion:  $8 * X / 4 * * X - 5 * * X * * X + 1$

Primer dibujo:



Más grande para facilidad:



a) Los nodos en orden que reciben x es: (según recorrido post-orden)  
[2, 5, 9, 11]

b) Notación polaca inversa:  $[8, X, *, 4, X, **, 1, 5, X, **, X, **, -, 1, +]$

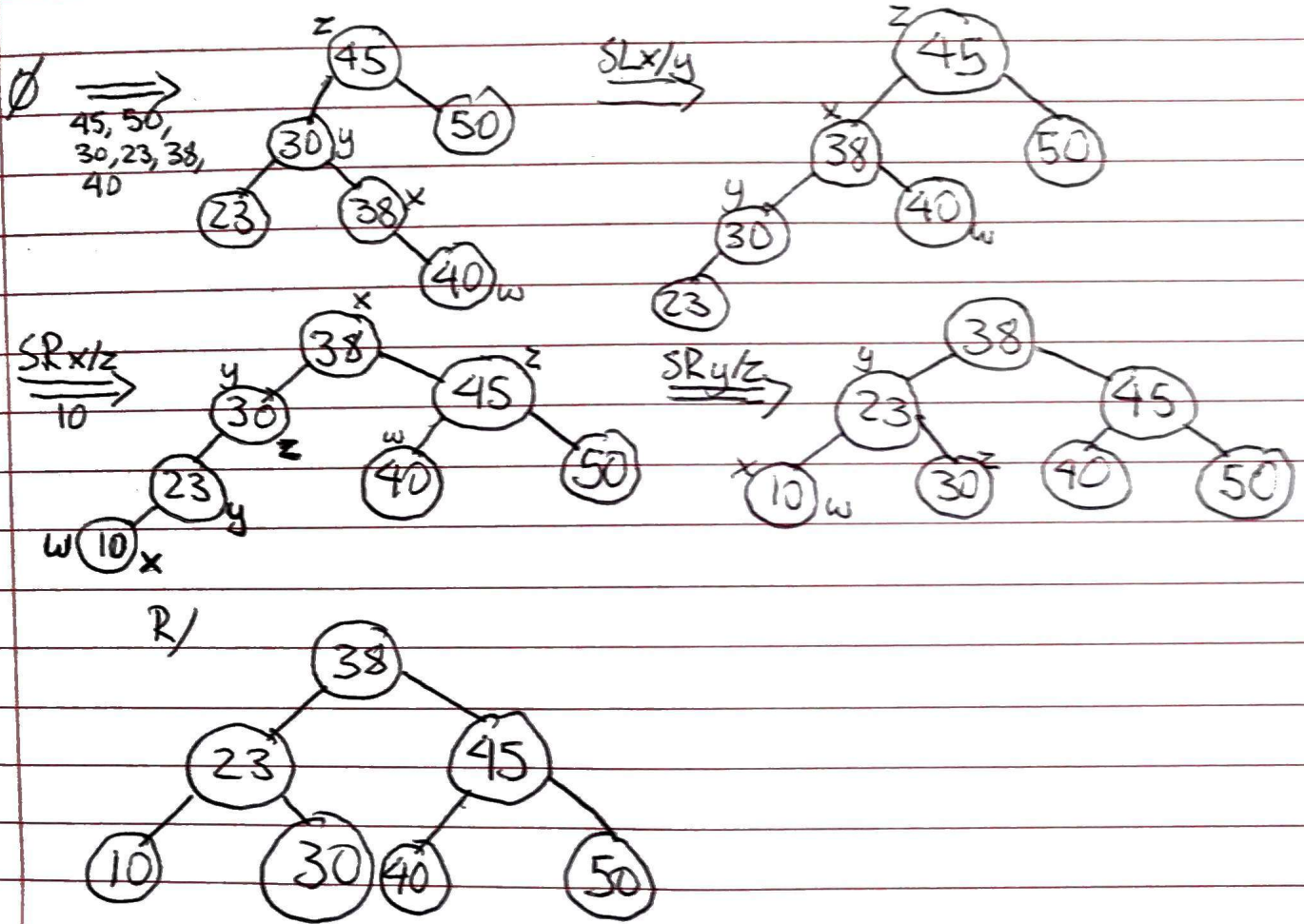


Diego Quiro's Art.iano

Diego

Pregunta 2

Lista de inserción: [45, 50, 30, 23, 38, 40, 10]



Diego Quiros Artizano Diego · Pregunta 3

hipotesis  $(A(B?))? = A|(AB)?$

$(A(B?))? = (A(B|\{\epsilon\}))?$  Sabiendo que  $B|\{\epsilon\} = B \cup \{\epsilon\}$

$(B? = B|\{\epsilon\}) = (A(B \cup \{\epsilon\}))?$  Como  $X(Y \cup Z) = XY \cup XZ$ ,  $XY, Z$  conjuntos

$= (AB \cup A\{\epsilon\})?$   $A\{\epsilon\} = A$

$= (AB \cup A)?$   $X \cup Y = Y \cup X$ ,  $X, Y$  conjuntos

$= (A \cup AB)?$   $A \cup AB = A|AB$

$= (A|AB)?$   $X? = X|\{\epsilon\}$   $X$  conjunto

$= (A|AB)|\{\epsilon\}$  (Asociatividad)

$= A|(AB|\{\epsilon\})$   $X|\{\epsilon\} = X?$   $X$  conjunto

$= A|(AB)?$

$\therefore (A(B?))? = A|(AB)?$



Diego Quiros Artigiano Diego Pregunta 4

$L =$  conjuntos de hileras binarias: Nunca ocurren más de dos 1 seguidos

$V = \{0, 1\}$  (Enunciado)

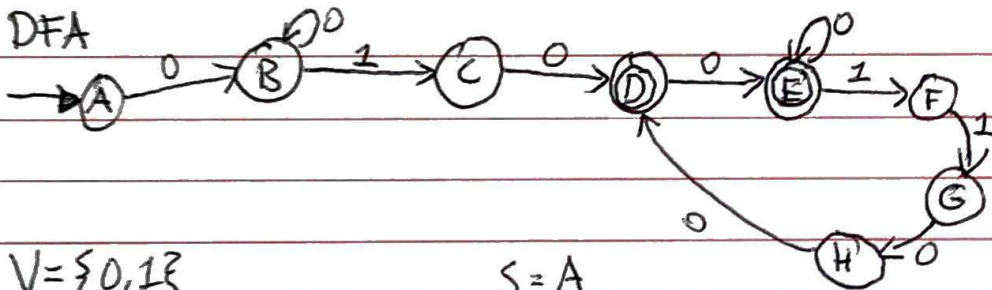
a) Ejemplos =  $\{01000, 0100001100110001100\}$

$RE1 = 0+10+$

$RE2 = RS2 = \{0\} + \{1\} + \{0^n / n \geq 2\} \{1^2\} \{0^n / n \geq 2\} = 0+10+(00+1100+)*$   
cumple las dos especificaciones

$RE = 0+10+(00+1100+)*$

b) DFA



$V = \{0, 1\}$

$S = A$

$Q = \{A, B, C, D, E, F, G, H\}$   $F = \{D, E\}$

$\delta$	0	1
A	B	
B	B	C
C	D	
D	E	
E	E	F
F		G
G	H	
H	D	

Diego Quiros Artiano ~~Diego~~ Pregunta 5

```
def intersection(Tree1, Tree2):
    newBST = BST()
    nodeT1 = Tree1.root()
    nodeT2 = Tree2.root()
    if Tree1.search(nodeT2): newBST.insert(nodeT2)
    if Tree2.search(nodeT1): newBST.insert(nodeT1)
    newBST.insert(intersection(Tree1.left, Tree2))
    newBST.insert(intersection(Tree1.right, Tree2))
    newBST.insert(intersection(Tree2.left, Tree1))
    newBST.insert(intersection(Tree2.right, Tree1))
    return newBST
```