Problem 1:

1. Let (a, d)∈(R1; R2); R3, by definition, there is a *c* with (a, c) ∈( R1; R2) and (c, d) ∈ R3

Equivalent to, there is a *c* with (there is a *b* with (a, b) ∈ R1 and (b, c) ∈ R2) and (c, d) ∈ R3

Equivalent to, there are c, b with (a, b) ∈ R1 and (b, c) ∈ R2 and (c, d) ∈ R3

Equivalent to, there are b, c with (a, b) ∈ R1 and (b, c) ∈ R2 and (c, d) ∈ R3

Equivalent to, there is a *b* with (a, b) ∈ R1 and there is c with (b, c) ∈ R2 and (c, d) ∈ R3

Equivalent to, there is a *b* with (a, b) ∈ R1 and (b, d) ∈ (R2; R3)

Equivalent to, (a, d) ∈ R1 ;(R2; R3)

So, (R1; R2); R3 = R1; (R2; R3).

1. Let I; R1 = {(a, c): there is a *b* with (a, b) ∈ I and (b, c) ∈ R1}

R1; I = {(a, c): there is a *b* with (a, b) ∈ R1 and (b, c) ∈ I}

Because, I = {(x, x): x ∈S}

So, for I; R1: a = b

I; R1 = {(a, c): there is a *a* with (a, a) ∈ I and (a, c) ∈ R1}

For R1; I: b = c

R1; I = {(a, c): there is a *c* with (a, c) ∈ R1 and (c, c) ∈ I}

Therefore, I; R1 = R1; I = R1

1. This does not hold and there is a counterexample

(a, b) ∈ R1, (b, c) ∈ R2

Then, (b, a) ∈ R1 ←, (c, b) ∈ R2←

R1 ←; R2← is not satisfied the definition

(a, c) ∈ R1; R2

(c, a) ∈ (R1; R2) ←

So, (R1; R2) ← = R1 ←; R2← is not true.

1. Consider (a, c) ∈ (R1∪R2); R3, by definition, there is a *b* with (a, b) ∈ (R1∪R2), and (b, c) ∈ R3

Equivalent to, there is a *b* with ((a, b) ∈ R1 or (a, b) ∈ R2) and (b, c) ∈ R3

Equivalent to, there is a *b* with ((a, b) ∈ R1 and (b, c) ∈ R3) or ((a, b) ∈ R2 and (b, c) ∈ R3)

Equivalent to, there is a *b* with (a, b) ∈ R1 and (b, c) ∈ R3 or there is a *b* with (a, b) ∈ R2 and (b, c) ∈ R3

Equivalent to, (a, c) ∈ R1; R3 or (a, c) ∈ R2; R3

Equivalent to, (a, c) ∈ (R1; R3) ∪ (R2; R3)

1. This does not hold and there is a counterexample

R1 = {(a, b), (a, c)}

R2 = (b, d)

R3 = (c, d)

So, R1; R2 = R1; R3 = (a, d)

But for R2 ∩ R3 = Ø

Therefore, R1; (R2∩R3) = (R1; R2) ∩ (R1; R3) is not true

Problem 2:

1. Base case: when j = i, Rj = Ri is true

Inductive case: if j = k ≥ i, Rk = Ri, and then prove Rk+1 = Ri

Rk+1 : = Rk ∪ (R; Rk)

= Ri ∪ (R; Ri)

= Ri+1

= Ri

Therefore, Rj = Ri is true for all j>=i.

1. For j ≥ 0; Rj ⊆ Rj+1

For 0 ≤ k < i; Rk ⊆ Ri

From question(a), when k ≥ i, Rk = Ri, so, Rk ⊆ Ri

So, for k ≥ 0, Rk ⊆ Ri

1. Base case: when n=0, P (0) = R0Rm

Due to R0 = I, and learn from problem1 (b), I; Rm = Rm

So, R0; Rm = I; Rm

For Inductive case, if Rk; Rm = Rk+m, need to prove Rk+1; Rm = Rk+m+1

Rk+1; Rm = [Rk ∪ (R; Rk)]; Rm

= Rk; Rm ∪ (R; Rk); Rm (due to problem1 (d))

= Rk+m ∪ (R; Rk); Rm

= Rk+m ∪ R; (Rk; Rm) (due to problem1 (a))

= Rk+m ∪ R; (Rk+m)

= Rk+m+1 (by definition)

So, P (n) hold for all n∈ N.

1. We can know from (a), (b), Rk ⊆ Rk+1

So, we should prove Rk+1 ⊆ Rk

1. Define (a, b) ∈ Rk and (b, c) ∈ Rk, need to prove (a, c) ∈ Rk

From problem1, (a, c) ∈ Rk; Rk

Due to question(c), Rk; Rk = R2k

Due to question(d), Rk = Rk+1

For j ≥ k, Rj = Rk

So, R2k = Rk

Therefore, (a, c) ∈ Rk

So, Rk is transitive.

1. For equivalence relation, need to prove Reflexivity, Symmetry and Transitive.

For Reflexivity, from question(b), I = R0 ⊆ R1 ⊆ R2 ⊆ Rk.

then, (x, x) ∈ Rk, so, (x, x) ∈ (R ∪ R←)k

So, Reflexivity holds

For Symmetry, I do not have idea to prove it

For Transitive, due to (e), Rk is Transitive, so (R ∪ R←)k is transitive.

Problem3

1. Definition: A Binary Tree is either:

(B) an empty Tree, or

(R) an ordered pair (LeftTree, RightTree)

1. count(T):

if (T.isEmpty()): (base case)

return 0

else: (Recursive)

return 1 + count(T.left) + count(T.right)

1. leaves(T):

if (T.isEmpty()): (base case)

return 0

else: (Recursive)

if (T.left.isEmpty() && T.Right.isEmpty()):

return 1

else:

return leaves(T.left) + leaves(T.right)

1. internal(T):

if (T.isEmpty()): (base case)

return 0

else: (recursive)

if (T.left.isEmpty() || T.right.isEmpty()):

return internal(T.left) + internal(T.right)

else:

return internal(T.left) + internal(T.right)+1

1. For this question, define n0 as leaves, n1 as an internal node which has one successor and n2 as a fully internal node.

For every node, it has parent node except root and the tree with n node has N-1 edges.

For n0, n1, n3, n0 has no child(edge), n1 has one child(edge), n2 has two child(edge)

Then, N-1 = 0 \* n0 + 1 \* n1 + 2 \* n2

Then, n0 + n1 +n2 -1 = 0 \* n0 + 1 \* n1 + 2 \* n2

So, n0 = 1 + n2

Therefore, leaves(T) = 1 + internal(T)

Problem4

1. For hi channel: hiAlpha, hiBravo, hiCharlie, hiDelta

For lo channel: loAlpha, loBravo, loCharlie, loDelta

1. (hiAlpha ∨ loAlpha) ∧ (hiBravo ∨ loBravo) ∧ (hiCharlie ∨ loCharlie) ∧ (hiDelta ∨ loDelta)
2. [(hiAlpha ∧ ¬loAlpha) ∨ (loAlpha ∧ ¬hiAlpha)] ∧ [(hiBravo ∧ ¬loBravo) ∨ (loBravo ∧ ¬hiBravo)] ∧

[(hiCharlie ∧ ¬loCharlie) v (loCharlie ∧ ¬hiCharlie)] ∧ [(hiDelta ∧ ¬loDelta) ∨ (loDelta ∧ ¬hiDelta)]

1. [(hiAlpha ∧ loBravo) ∨ (loAlpha ∧ hiBravo)] ∧ [(hiBravo ∧ loCharlie) ∨ (loBravo ∧ hiCharlie)] ∧

[(hiCharlie ∧ loDelta) ∨ (loCharlie ∧ hiDelta)]

(b)

(i) if ϕ1 ∧ ϕ2 ∧ ϕ3 is satisfiable, Alpha should be hi channel, Bravo should be lo channel,

Charlie should be hi channel and Delta should be lo channel.

So, hiAlpha = 1, loAlpha = 0

hiBravo = 0, loBravo = 1

hiCharlie = 1, loCharlie = 0

hiDelta = 0, loDelta = 1

(ii) if avoid interference, Alpha and Charlie use hi channel, Bravo and Delta use lo channel.