*Basado en el algoritmo de comprobar si L(A1) = L(A2) (Clase magistral – 12 feb.)*

1. Convertir el AFN en un AFD en el siguiente orden: (Clase magistral – 6 feb.)
   1. Algoritmo de la pizarra para ver cuáles son accesibles. (Clase magistral – 6 feb.)
   2. Algoritmo de la pizarra para ver cuáles son co-accesibles. (Clase magistral – 6 feb.)
      1. S'0 = ø
      2. S'1 = F (conjunto de estados finales)
      3. i = 1
      4. while (Si != Si-1)
         1. i++
         2. S'i = { q | ∃s ϵ S'i-1, a ϵ Σ tal que s ϵ δ(q,a) }
2. Completar el AFD. (Clase magistral – 12 feb.)
   1. Aquellas transiciones del abecedario que no sean especificadas se llevarán a un estado trampa del que no podrán salir.
3. Calcular complementario del AFD. (**y !ϵ L(AFD)** == **y ϵ L(complementario AFD)**) (Clase magistral – 12 feb.)
   1. Al estar completo, basta con hacer que los estados finales no lo sean y viceversa.
4. L(A) = L(AFD) \* L(complementario AFD) = { xy | x ϵ L(AFD), y ϵ L(complementario AFD) }. (Clase magistral – 8 feb.)
   1. Consiste en crear transiciones vacías desde los estados finales del AFD hasta los iniciales del complementario del AFD.
5. (OPCIONAL) Quitar los estados vacíos del autómata resultante. (Clase magistral – 8 feb.)
   1. λ(q) = conjunto de todos los estados accesibles desde q con sólo λ-transiciones.
   2. δ(q,a) ϶ s si y solo si δ(t,a) ϶ s para w,t ϵ λ(q).
   3. q ϵ F' si y solo si λ(q) ∩ F = ø