[ENUNCIADO 1](https://drive.google.com/file/d/1v9tU3YMeRZO4HKTWDjQok8PE_v26Yfdn/view), [ENUNCIADO 2](https://drive.google.com/file/d/14a2t6PxF4M7p552hq4p4dY07U4njRk9-/view)

1. 1
   1. Postula que todo problema computable es resoluble por una MT
2. 2
   1. El lenguaje está formado por todos los posibles conjuntos que contienen un subconjunto y tales que , y
3. 3
   1. Significa que para cada MT de cintas semi-infinitas, existe otra de cintas infinitas capaz de reconocer el mismo lenguaje; y viceversa
4. 4
5. 5
   1. Porque tanto como tienen lenguajes con MTs que aceptan sus cadenas
   2. Como son todos los lenguajes posibles, tiene un subconjunto de esos lenguajes
   3. Como , todas los lenguajes de están en y en
   4. d
      1. :
      2. :
6. 6
   1. No, porque como son lenguajes de y las MTs no están restringidas, no puede detectar tras una cantidad de pasos que y/o se quedaron loopeando
   2. Corrección: Si, como es RE y no hablamos de la unión, no hay problema si se queda loopeando
7. 7
   1. a
      1. ejecuta infinitos pasos de sobre , verificando luego de cada uno, si terminó en o
      2. Si acepta a , termina en
      3. Si rechaza a , termina en
      4. Si loopea a , termina loopeando
      5. Por lo tanto,
8. 8
   1. Significa que:
      1. Si ,
      2. Si ,
   2. Pertenece a , porque como debe ser igual o más difícil que , si la dificultad máxima es , no puede ser más difícil
   3. Por contrarrecíproco, si , entonces
9. 9

|  | **MTs generales** | **AFs** | **APs** |
| --- | --- | --- | --- |
| **a)** | A izquierda, derecha, o ninguno | A derecha | A derecha |
| **b)** | Puede escribir | Sólo lectura | Sólo lectura |
| **c)** | No siempre () | Siempre | Siempre |
| **d)** | Acepta si  Puede rechazar si | Acepta si  Rechaza si | Acepta si  Rechaza si |

1. 10
   1. Significa que existe una MT que acepta o rechaza cadenas de símbolos, en pasos
2. 11
   1. Postula que si tengo un problema que se resuelve en tiempo eficiente (polinomial), en un modelo de ejecución razonable, también lo puedo resolver en tiempo eficiente en otro modelo
3. 12
   1. a
      1. Si está en , significa que tiene soluciones verificables en tiempo polinomial
      2. Dado un , como sumar tarda tiempo polinomial, sumar los puntos también
      3. Así, verificar una solución usa tiempo polinomial, por lo que el lenguaje está en
4. 13
   1. a
      1. Por definición, la única forma que puede darse que y , es que y
      2. Como se sabe por hipótesis que , entonces, necesariamente se da que
5. 14
   1. a
      1. i
      2. ii
      3. Por definición, si , entonces
      4. Como , es igual o más difícil que , por lo que se puede reducir de a
      5. Si , y si , ; entonces si , se da que , por lo que
      6. Como se pudo reducir de a , siendo uno de los lenguajes a reducir, entonces está en
6. 15
   1. Porque el hecho de que use celdas de trabajo, no asegura que siempre parará