[ENUNCIADO](https://mega.nz/folder/1oU1AD6J#J8ukFAJMMmpik0Otc0fZ5g/file/wtsx0III)

1. 1
   1. Se construirá una MT que recibe un , a partir de una (cuyo lenguaje es ) y (que genera todas las cadenas ):
      1. Copia a la cinta 2
      2. Escribe al final de la cinta 2
      3. Ejecuta a , generando un al final de la cinta 2
      4. Ejecuta a sobre la cinta 2
      5. Si terminó en , termina en
      6. Sino, vuelve a i)
   2. Entonces, , porque:
      1. Si existe tal que , terminará en
      2. Si no existe, se quedará loopeando
2. 2
   1. Es una reducción válida, porque:
      1. Es total computable:
         1. Verificar la sintaxis tarda tiempo polinomial
         2. Generar la cadena tarda tiempo constante
         3. Generar la cadena tarda tiempo polinomial
      2. Si tiene sintaxis , y ( acepta a ): El generado pasará por antes de llegar a , por lo que
      3. Si tiene sintaxis , y ( rechaza a o loopea): El generado no pasará por porque tampoco llegará a , por lo que
      4. Si no tiene sintaxis : El generado no es una cadena válida, por lo que
   2. b
      1. Se construirá una MT que recibe un con sintaxis :
         1. Si no tiene sintaxis , termina en
         2. Si no contiene al estado , termina en
         3. Inicializa y
         4. Si está en el estado , termina en
         5. Ejecuta un paso de sobre
         6. Vuelve a (4)
      2. Si nunca pasa por , puede quedarse loopeando, por lo que
3. 3
   1. Como se requiere calcular asignaciones, para las combinaciones, para cada , siendo un valor variable (contenido en el input), el tiempo necesario es del orden , por lo que
   2. porque se puede encontrar un certificado de longitud (sucinto) para verificar si . Esto es, una asignación de valores de verdad que hacen a verdadera
4. 4
   1. Es una reducción polinómica válida, porque:
      1. tarda tiempo polinomial: Copiar o a la salida tarda tiempo lineal
      2. Si : , por lo que
      3. Si : , por lo que