

Modelos Avanzados de Computación

RELACIÓN DE EJERCICIOS 3

- ① Construir un programa Post-Turing que calcule la función $f(u) = u^{-1}$ donde $u \in \{0, 1\}^*$.

$0101 \rightarrow 0X01 \rightarrow 10X01 \rightarrow 10XX1 \rightarrow 010XX1 \rightarrow 010XXX \rightarrow 1010XXX \rightarrow 1010$

[A] RIGHT

IF X GOTO A
IF # GOTO H
IF 0 GOTO B
IF 1 GOTO D

Reconocer los X's hasta que se acaben
H terminado de leer la palabra
Leí un 0
Leí un 1

[E] LEFT

IF X GOTO C
IF 0 GOTO C
IF 1 GOTO C
PRINT 1
IF 1 GOTO F

Si leí un 1, voy al principio de la palabra a escribirlo

[B] PRINT X

Tacho el 0 leído

IF X GOTO C

[F] RIGHT

IF 0 GOTO F
IF 1 GOTO F
IF X GOTO A

Me muevo a la derecha hasta encontrar la primera X

[C] LEFT

Si leí un 0, voy al principio de la palabra a escribirlo

IF X GOTO C
IF 0 GOTO C
IF 1 GOTO C
PRINT 0
IF 0 GOTO F

[H] LEFT

IF X GOTO G
HALT

Cuando lo encuentro, repito el proceso

[D] PRINT X

Tacho el 1 leído

IF X GOTO E

[G] PRINT #

IF # GOTO H

Borro las X's

- ② Construir un programa Post-Turing que dado un número u en binario calcule $u + 1$.

IF # GOTO S

[S] La palabra vacía no es una entrada válida, luego, el programa no puede terminar con ella

[A] RIGHT

IF 0 GOTO A
IF 1 GOTO A
LEFT
IF 0 GOTO B
IF 1 GOTO C

Voy al final de la palabra

Loo el último símbolo de la palabra

[B] PRINT 1

Cambio el 0 (o el #) por un 1 y termino

HALT

[C] PRINT 0

LEFT
IF 0 GOTO B
IF # GOTO B
IF 1 GOTO C

Cambio el 1 por un 0 y busco el próximo 0 para cambiarlo por un 1 o, en su defecto, poner un 1 al inicio de la palabra

- ③ Construir un programa Post-Turing que dadas dos cadenas uv donde $u, v \in \{0, 1\}^*$ calcule si la cadena u es una subcadena de la cadena v .

Idea con tres ejemplos:

$$1) \quad 10 \in 100 \rightarrow X0 \in X00 \rightarrow XY \in XY0 \rightarrow XY \in XY2 \# \quad \checkmark$$

$$2) \quad 11 \in 100 \rightarrow X1 \in X00 \rightarrow XX \in X00 \rightarrow XX \in X00 \# \quad \times$$

$$3) \quad 100 \in 1010 \rightarrow X00 \in X010 \rightarrow XY0 \in XY10 \rightarrow XY0 \in XY1Y \rightarrow XY0 \in XY2Y \quad \times$$

- ④ Construir un programa con variables que concatene dos cadenas sobre $\{0, 1\}$. Se supone que ambas cadenas están en las variables X_1 y X_2 y la salida en la variable Y .

$$\mathcal{A} = \{0, 1\} \\ a_1 \quad a_2$$

$[A]$ IF X_2 ENDS a_i , GOTO B_i Copiamos X_2 en Y
 IF X_1 ENDS a_i , GOTO C_i ($i = 1, 2$) Copiamos X_1 en Y
 HALT Terminamos

$[B_i]$ $y \leftarrow a_i Y$
 $X_2 \leftarrow X_2 -$
 GOTO A

$[C_i]$ $y \leftarrow a_i Y$
 $X_1 \leftarrow X_1 -$
 GOTO A

Habrá que repetir con el siguiente a de V (b anterior a a) el paso 1 lo eliminamos con $\#$ y restauraremos los valores $X=0$ e $Y=0$)

- ⑤ Construir un programa con variables que dadas dos cadenas sobre $\{0, 1\}$ (en las variables X_1 y X_2) calcule el número de apariciones de X_1 como subcadena de X_2 . La salida será un número en binario en Y .

Idea con un ejemplo:

$$X_1 = 10 \quad V = 10 \quad \left(\begin{array}{l} \text{(Copiamos } X_1 \text{ en } V \\ \text{y nunca lo modificamos)} \end{array} \right)$$

$$X_2 = 101011$$

$$\begin{array}{llllllll} X_1 = 10 & & X_1 = 1 \\ X_2 = 101011 \rightarrow & \left(\begin{array}{l} X_2 = 1 \\ V = 10 \end{array} \right) \rightarrow & \left(\begin{array}{l} X_2 = 10 \\ V = 10 \end{array} \right) \rightarrow & \left(\begin{array}{l} X_2 = 101 \\ V = 10 \end{array} \right) \rightarrow & \left(\begin{array}{l} X_2 = 1010 \\ V = 10 \end{array} \right) \rightarrow & \left(\begin{array}{l} X_2 = 10101 \\ V = 10 \end{array} \right) \rightarrow & \left(\begin{array}{l} X_2 = 101011 \\ V = 10 \end{array} \right) \rightarrow & \left(\begin{array}{l} X_2 = 101011 \\ V = 10 \end{array} \right) \rightarrow & \left(\begin{array}{l} X_2 = 101011 \\ V = 10 \end{array} \right) \\ Y = 0 & & Y = 0 & & Y = 0 & & Y = 0 & & Y = 0 \\ & & & & & & & & & \end{array}$$

$$\begin{array}{l} X_1 = \epsilon \\ X_2 = \epsilon \\ V = 10 \\ Y = 10 \end{array} \rightarrow \text{HALT}$$

- 6.) Construir un programa con variables que acepte el lenguaje $L = \{w \in \{0,1\}^* \mid w = w^{-1}\}$.

Entrada: $X \quad A = \{0, 1\}_{a_1 a_2}$

$U \leftarrow \epsilon$

$V \leftarrow \epsilon$

[A] IF X ENDS a_i GOTO $A_i \quad (i=1,2)$

[B] IF U ENDS a_i GOTO $D_i \quad (i=1,2)$ Comprobamos si $U = V$
HALT

[C_i] $U \leftarrow Ua_i$ Mejoramos en U el contenido de X al revés
 $V \leftarrow a_iV \quad (i=1,2)$ Mejoramos en V el contenido de X
 $X \leftarrow X-$
GOTO A

[D_i] IF V ENDS a_i GOTO $E_{(i=1,2)}$ Ambos símbolos coinciden
GOTO N No coinciden

[E_i] $U \leftarrow U-$
 $V \leftarrow V-$
GOTO B

[N] Flechazamos

- 7.) Construir un programa con variables que dada una cadena $u \in \{0,1\}^*$ calcule la cadena w formada por los símbolos que ocupan las posiciones impares de u y en el mismo orden que aparecen en u .

$$X = u \in \{0,1\}^* \quad A = \{0, 1\}_{a_1 a_2}$$

IF X ENDS a_i GOTO A_i
HALT

[A_i] $V \leftarrow a_iV$
 $X \leftarrow X-$
IF X ENDS a_i GOTO $B_i \quad (i=1,2)$
 $y \leftarrow V$
 $HALT$

Guardo posiciones impares empezando por el final

La longitud de X es impar, luego, los impares son los impares comenzando por el final

[B_i] $U \leftarrow a_iU$
 $X \leftarrow X-$
IF X ENDS a_i GOTO $A_i \quad (i=1,2)$
 $y \leftarrow U$
 $HALT$

Guardo posiciones pares empezando por el final

La longitud de X es par, luego, los impares son los pares comenzando por el final

- 10.) Dado el siguiente programa con variables:

IF X ENDS 0 GOTO A
IF X ENDS 1 GOTO B
HALT

[A] $X \leftarrow X-$
 $Y \leftarrow 0Y$
IF X ENDS 0 GOTO A

IF X ENDS 1 GOTO B
HALT

[B] $X \leftarrow X-$
 $Y \leftarrow 1Y$
IF X ENDS 0 GOTO A
IF X ENDS 1 GOTO B
HALT

$$\begin{array}{ccccccc}
 X = 10110 & X = 1011 & X = 101 & X = 10 & X = 1 & X = \epsilon \\
 Y = \epsilon & \xrightarrow{} Y = 0 & \xrightarrow{} Y = 10 & \xrightarrow{} Y = 110 & \xrightarrow{} Y = 0110 & \xrightarrow{} Y = 10110
 \end{array}$$

El programa con variables dada copia X en Y y devuelve Y .

Un programa Post-Turing equivalente sería el siguiente:

[A] IF 0 GOTO B
IF 1 GOTO C

[B] PRINT 0
RIGHT
IF 0 GOTO B
IF 1 GOTO C
HALT

[C] PRINT 1
RIGHT
IF 0 GOTO B
IF 1 GOTO C
HALT

11) Dado el siguiente programa Post-Turing

LEFT
[C] RIGHT
IF # GOTO E
IF 0 GOTO A
IF 1 GOTO C
[A] PRINT #
IF ~~#~~ GOTO C
[E] HALT

construir una MT equivalente.

$$01101 \rightarrow \#1101 \rightarrow \#\#1$$

El programa Post-Turing dado cambia los 0's de la palabra por #

Una MT equivalente sería la siguiente:

$$M = \left(Q, A, B, \delta, q_0, \#, \{ q_f \} \right) \quad \left. \begin{array}{l} \rightarrow Q = \{ q_0, q_f \} \\ \rightarrow A = \{ 0, \# \} \\ \rightarrow B = \{ 0, \# \} \end{array} \right\} \quad \left. \begin{array}{l} \delta(q_0, 0) = (q_0, \#, D) \\ \delta(q_0, 1) = (q_0, 1, D) \\ \delta(q_0, \#) = (q_f, \#, D) \end{array} \right\}$$

- 12) Dada la MT $M = (\{q_0, q_1, q_2, q_3, q_4\}, \{0, 1\}, \{0, 1, X, Y, \#\}, \delta, q_0, \#, \{q_4\})$ donde las transiciones no nulas son las siguientes:

$$\begin{array}{ll} \delta(q_0, 0) = (q_1, X, D) & \delta(q_0, Y) = (q_3, Y, D) \\ \delta(q_1, 0) = (q_1, 0, D) & \delta(q_1, 1) = (q_2, Y, I) \\ \delta(q_1, Y) = (q_1, Y, D) & \delta(q_2, 0) = (q_2, 0, I) \\ \delta(q_2, X) = (q_0, X, D) & \delta(q_2, Y) = (q_2, Y, I) \\ \delta(q_3, Y) = (q_3, Y, D) & \delta(q_3, \#) = (q_4, \#, D) \end{array}$$

construir un programa con variables equivalentes (se pueden usar macros).

La MT dada acepta palabras de la forma 0^n1^n , con $n > 0$.

Un programa con variables equivalentes sería el siguiente:

$$\begin{array}{l} U \leftarrow \epsilon \\ V \leftarrow \epsilon \end{array}$$

IF X ENDS 1 GOTO A
GOTO S Si X no termina en 1, entrada inválida.

[B] V \leftarrow 1V
 $X \leftarrow X-$
IF X ENDS 1 GOTO A
GOTO B Copiamos los 1's en V

[B] U \leftarrow 0U
 $X \leftarrow X-$
IF X ENDS 0 GOTO B
IF $X \neq \epsilon$ GOTO S
GOTO C Tras copiar los 1's, copiamos los 0's en U
 Hay 1's precediendo a los 0's

[C] U \leftarrow U-
V \leftarrow V-
IF $U \neq \epsilon$ GOTO D
GOTO E

[D] IF $V \neq \epsilon$ GOTO C
GOTO S Hay más 0's que 1's

[E] IF $V = \epsilon$ GOTO S
HALT Hay más 1's que 0's

[S] Rechazamos

- 13) Construir un programa con variables numéricas que calcule $f(x_1, x_2) = x_1 + x_2$ y otro que calcule $f(x_1, x_2) = x_1x_2$.

•) $f(x_1, x_2) = x_1 + x_2$

$$\begin{array}{l} U \leftarrow x_1 \\ V \leftarrow x_2 \\ \text{IF } x_2 \neq 0 \text{ GOTO A} \\ \text{GOTO H} \end{array}$$

Si $x_2 = 0$, devolvemos x_1

[B] U \leftarrow U+1
V \leftarrow V-1
IF V $\neq 0$ GOTO B
GOTO H

[H] Y \leftarrow U
HALT

•) $f(x_1, x_2) = x_1x_2$

$$\begin{array}{l} U \leftarrow 0 \\ V \leftarrow x_2 \\ \text{IF } x_2 \neq 0 \text{ GOTO A} \\ \text{GOTO H} \end{array}$$

Si $x_2 = 0$, devolvemos 0

[A] U \leftarrow U+x₁
V \leftarrow V-1
IF V $\neq 0$ GOTO A
GOTO H

[H] Y \leftarrow U
HALT

(Hemos hecho esta
instrucción ya en
 $f(x_1, x_2) = x_1 + x_2$)

Resultado: $x_1 + x_2$

Resultado: x_1x_2

- 14) Escribir un programa con variables numéricas que calcule $f(x) = 1$ si x es par y 0 en caso contrario.

```

IF X≠0 GOTO A
Y←1
HALT

[A] X←X-1
IF X≠0 GOTO B
Y←0
HALT

[B] X←X-1
IF X≠0 GOTO A
Y←1
HALT
    
```

Impar

- 15) Escribir un programa con variables numéricas que $f(x) = 1$ si x es primo y 0 en caso contrario.

•) MACROS:

* IF $X_1 < X_2$ GOTO E ≡

<u>$U \leftarrow X_1$</u>	<u>$V \leftarrow X_2$</u>
<u>[A]</u>	<u>$U \leftarrow U - 1$</u>
	<u>$V \leftarrow V - 1$</u>
	<u>IF $U \neq 0$ GOTO A</u>
	<u>IF $V \neq 0$ GOTO E</u>
	<u>$V = 0$ si $X_2 \leq X_1$</u>

* $Y \leftarrow X_1 \% X_2$ ≡

(Suponemos $X_2 > 0$)

<u>$U \leftarrow X_2$</u>	<u>$V \leftarrow X_1$</u>
<u>[A]</u>	<u>IF $V < X_2$ GOTO H</u>
	<u>$U \leftarrow U - 1$</u>
	<u>$V \leftarrow V - 1$</u>
	<u>IF $U \neq 0$ GOTO A</u>
	<u>IF $V < X_2$ GOTO H</u>
	<u>$U \leftarrow X_2$</u>
	<u>GOTO A</u>
<u>[H]</u>	<u>$Y \leftarrow V$</u>
	<u>Si $X_1 < X_2$, el resto es X_1</u>
	<u>Resto $V - X_2$</u>
	<u>Si $V < X_2$, el resto es V</u>
	<u>Si $V \geq X_2$, vuelvo a restarle X_2 a V</u>

•) Programa con variables numéricas:

```

V ← X
W ← 0
W ← W + 1
IF W < V GOTO A
Y ← 0
HALT

[A] V ← V - 1
IF W < V GOTO B
Y ← 1
HALT

[B] 2 ← X % V
IF 2 ≠ 0 GOTO A
Y ← 0
HALT
    
```

Divido X entre V

Si el resto no es 0, sigo dividiendo X por otros números

Si el resto es 0, X no es primo

Ejercicios extras

- ① Diseñar un programa con variables que dadas dos cadenas $u, v \in \{0,1\}^*$, calcule u^v .

```

 $U \leftarrow X_1$ 
 $V \leftarrow X_2$ 
 $Y \leftarrow \epsilon$ 
IF  $V \neq \epsilon$  GOTO A
HALT
    
```

Si $V = \epsilon$, entonces $Y = u^0 = \epsilon$

[A]

```

 $V \leftarrow V-$ 
IF U ENDS 0 GOTO B
IF U ENDS 1 GOTO C
HALT
    
```

Comprobó X_1 con Y

Si $U = \epsilon$, entonces $Y = \epsilon$

[B]

```

 $Y \leftarrow 0Y$ 
 $U \leftarrow U-$ 
IF U ENDS 0 GOTO B
IF U ENDS 1 GOTO C
 $U \leftarrow X_1$ 
IF  $V \neq \epsilon$  GOTO A
HALT
    
```

Cuando $U = \epsilon$, lo actualizo para la siguiente iteración

Cuando $V = \epsilon$, ya tendríamos $Y = u^{\mid V \mid}$

[C]

```

 $Y \leftarrow 1Y$ 
 $U \leftarrow U-$ 
IF U ENDS 0 GOTO B
IF U ENDS 1 GOTO C
 $U \leftarrow X_1$ 
IF  $V \neq \epsilon$  GOTO A
HALT
    
```

Si U termina en 1, añado un 1 al principio de Y

Cuando $U = \epsilon$, lo actualizo para la siguiente iteración

Cuando $V = \epsilon$, ya tendríamos $Y = u^{\mid V \mid}$

- ② Diseñar un programa con variables numéricas que calcule el máximo común divisor de dos enteros.

•) Idea: Miro cuál es el menor de los dos números y divido ambos por todos los números menores o iguales a este. Si ambos tienen resto 0 al dividir, ese es el MCD. Si no, pruebo a dividir con el siguiente número.

MACROS:

* IF $X_1 < X_2$ GOTO E \equiv
 [A] $U \leftarrow X_1$
 $V \leftarrow X_2$
 $U \leftarrow U-1$
 $V \leftarrow V-1$
 IF $U \neq 0$ GOTO A
 IF $V \neq 0$ GOTO E
 Resto $X_2 - X_1$

* $Q, R \leftarrow X_1 / X_2 \equiv$
 (Suponemos $X_2 > 0$) \equiv
 [A] $U \leftarrow X_2$
 $R \leftarrow X_1$
 $Q \leftarrow 0$
 IF $R < X_2$ GOTO H \quad Si $X_1 < X_2$, el resto es X_1
 Resto $R - X_2$
 [B] $U \leftarrow U-1$
 $R \leftarrow R-1$
 IF $U \neq 0$ GOTO A
 $Q \leftarrow Q+1$
 IF $R < X_2$ GOTO H
 $U \leftarrow X_2$
 GOTO A
 Sumaré 1 al cociente
 Si $R < X_2$, el resto es R
 Si $R \geq X_2$, vuelvo a restarle X_2 a R
 [E] FIN

•) Programa con variables numéricas:

$U \leftarrow X_1$
 $V \leftarrow X_2$
 $W \leftarrow 0$
IF $U < V$ GOTO A
 $W \leftarrow V$ En W meto una copia de V (el menor número)
GOTO B

[A] $2 \leftarrow U$
 $U \leftarrow V$
 $V \leftarrow 2$
 $W \leftarrow V$ En W meto una copia de V (el menor número)

[B] $Q, R \leftarrow U/W$
Dividir U entre W
IF $R \neq 0$ GOTO C
 $Q, R \leftarrow V/W$
Dividir V entre W
IF $R \neq 0$ GOTO C
 $Y \leftarrow W$
Si el resto no es cero, sigo dividiendo
Si el resto es cero, $W = \text{mcd}(X_1, X_2)$
HALT

[C] $W \leftarrow W - 1$ Decremento W en una unidad, y vuelvo a dividir
GOTO B