MODELOS AVANZADOS DE COMPUTACIÓN

Relación 3

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

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

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

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.

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.

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

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.

8. Construir un programa con variables sobre $\{a,b\}$ que dadas dos cadenas $u_1, u_2 \in \{a,b\}^*$ calcule la cadena u cuyo número verifica $Z(u) = Z(u_1) + Z(u_2)$ (es decir hacer la suma de números representados por cadenas de caracteres sobre $\{a,b\}$.

9. Considerar un lenguage Post Turing para programas con varias cintas. Hay un número finito de cintas y en cada momento una de ellas está activa, inicialmente la primera. Hay dos instrucciones UP and DOWN que se mueven a la cinta superior e inferior respectivamente. Demostrar que todo cálculo realizado por un programa Post Turing con varias cintas, puede realizarse con un programa Post Turing con una sola cinta.

10. Dado el siguiente programa con variables:

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

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

construir un programa Post-Turing equivalente (se pueden usar macros).

11. Dado el siguiene 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.

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 equivalente (se pueden usar macros).

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

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

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

16. Escribir un programa con variables numéricas que calcule $f(x) = y$ donde $y = N(C(x)-)$ donde Z y C son las codificaciones sobre un alfabeto de n símbolos.

17. Escribir un programa con variables numéricas que calcule $f(x) = y$ donde $y = N(a_i C(x))$ donde Z y C son las codificaciones sobre un alfabeto de n símbolos.