

Teoría de Autómatas y Lenguajes Formales

Práctica 3

José Manuel Rodríguez Chicano

December 26, 2022

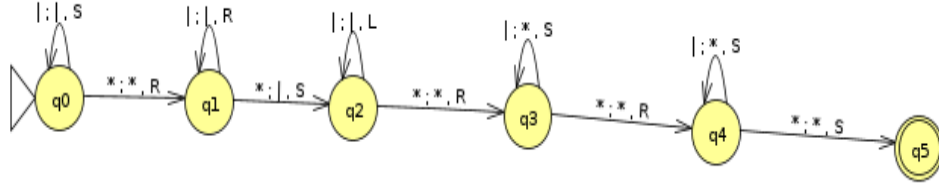
1 Ejercicio 1

Define the TM solution of exercise 3.4 of the problem list and test its correct behaviour.

La maquina de Turing sería la siguiente.

0	*	l	1
0			0
1	*		2
1		l	1
2	*	l	3
2		r	2
3	*	l	4
3		*	3
4	*	h	4
4		*	4

Y su resolución en JFLAP (de manera invertida porque en este comienza a la izquierda en vez de a la derecha) Sería la siguiente.



Esta máquina de Turing ha sido comprobada con éxito, la probé con $1 + 1$, es decir $|| * ||$ al ser unaria y con $||| * ||$ es decir $2+1$ y los resultados fueron los esperados, es decir $|||$ y $||||$

2 Ejercicio 2

Define a recursive function for the sum of three values.

En este caso utilizamos la recursion primitiva que viene definida en los apuntes de la siguiente manera:

Let $k \geq 0$ and the functions

$$g : N^k \rightarrow N$$

$$h : N^{k+2} \rightarrow N$$

$$f : N^{k+1} \rightarrow N$$

$$f(\vec{n}, m) = \begin{cases} g(\vec{n}) & \text{if } m = 0 \\ h(\vec{n}, m - 1, f(\vec{n}, m - 1)) & \text{if } m > 0 \end{cases}$$

Entonces la definición de la función recursiva para la suma de 3 que he implementado es la siguiente:

$$addition3(n) = \langle \langle \pi_1^1 | \sigma(\pi_3^3) | \sigma(\pi_4^4) \rangle (n) \rangle$$

Probando así en octave con los ejemplos $3+3+5$ cuyo resultado es 11:

```

octave:1> recursiveexpression('addition3')
warning: textread is obsolete; use textscan instead
ans = <<n^1_1|o(n^3_3)>|o(n^4_4)>
octave:2> evalrecfunction('addition3', 3, 3, 5)
addition3(3,3,5)
<<n^1_1|o(n^3_3)>|o(n^4_4)>(3,3,5)
<<n^1_1|o(n^3_3)>|o(n^4_4)>(3,3,4)
<<n^1_1|o(n^3_3)>|o(n^4_4)>(3,3,3)
<<n^1_1|o(n^3_3)>|o(n^4_4)>(3,3,2)
<<n^1_1|o(n^3_3)>|o(n^4_4)>(3,3,1)
<<n^1_1|o(n^3_3)>|o(n^4_4)>(3,3,0)
<n^1_1|o(n^3_3)>(3,3)
<n^1_1|o(n^3_3)>(3,2)
<n^1_1|o(n^3_3)>(3,1)
<n^1_1|o(n^3_3)>(3,0)
n^1_1(3) = 3
o(n^3_3)(3,0,3)
n^1_1(3,0,3) = 3

o(3) = 4
o(n^3_3)(3,1,4)
n^1_1(3,1,4) = 4

o(4) = 5
o(n^3_3)(3,2,5)
n^1_1(3,2,5) = 5

o(5) = 6
o(n^4_4)(3,3,0,6)
n^1_1(3,3,0,6) = 6

o(6) = 7
o(n^4_4)(3,3,1,7)
n^1_1(3,3,1,7) = 7

o(7) = 8
o(n^4_4)(3,3,2,8)
n^1_1(3,3,2,8) = 8

o(8) = 9
o(n^4_4)(3,3,3,9)
n^1_1(3,3,3,9) = 9

o(9) = 10
o(n^4_4)(3,3,4,10)
n^1_1(3,3,4,10) = 10

o(10) = 11
ans = 11

```

3 Ejercicio 3

Implement a WHILE program that computes the sum of three values. You must use an auxiliary variable that accumulates the result of the sum.

```

Q: (3, 1, s)
s:
X4 := X1;
while X2 != 0 do
    X2:=X2-1;
    X4:=X4+1
od;

while X3 != 0 do
    X3:= X3-1;
    X4:= X4+1
od;
X1:=X4

```