

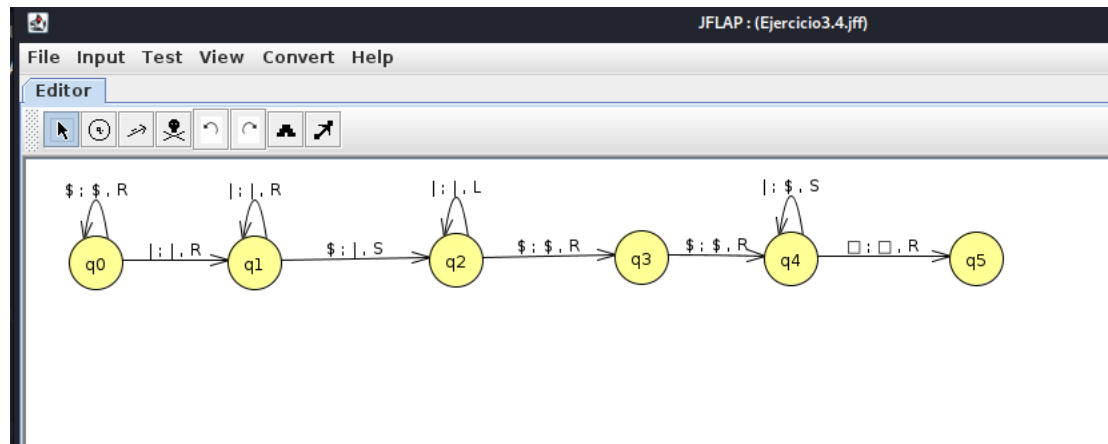
Práctica 3

Pedro Antonio Aguilar Lima

2022-2023

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

3.4. Prove that the function $add(x, y) = x + y$, with $x, y \in \mathbb{N}$ is Turing-computable using the unary notation $\{| \}$. You have to create a TM with two arguments separated by a blank symbol that starts and ends behind the strings.



2 Define a recursive function for the sum of three values.

Se usa la funcion definida de adición: el programa haría algo parecido a lo siguiente, es decir primero sumará los primeros dos elementos y luego al resultado que esto ofrezca, le suma el siguiente término:

$$suma3 = \langle \langle \pi_1^3 | \sigma(\pi_2^3) \rangle | \sigma(\pi_3^3) \rangle$$

```
octave:5> evalrecfunction('addition(addition(pi^3_1,pi^3_2),pi^3_3)',3,2,1)
addition(addition(pi^3_1,pi^3_2),pi^3_3)(3,2,1)
addition(pi^3_1,pi^3_2)(3,2,1)
pi^3_1(3,2,1) = 3

pi^3_2(3,2,1) = 2

addition(3,2) recursive function for the sum of three values
<pi^3_1|sigma(pi^3_2)>(3,2)
<pi^3_1|sigma(pi^3_2)>(3,1) ya definida:
<pi^3_1|sigma(pi^3_2)>(3,0)
pi^3_1(3) = 3
sigma(pi^3_2)(3,0,3) suma dos números para crear otra funcion que sume
pi^3_2(3,0,3) = 3
pi^3_2(3,0,3) primero sumara los dos primeros elementos y despues le sumará
sigma(3) = 4 el tercer elemento.
sigma(pi^3_2)(3,1,4)
pi^3_2(3,1,4) = 4 sigma(pi^3_1,pi^3_2),pi^3_3)

sigma(4) = 5 de la funcion de addition3 ejecutada en Octave
octave:5> evalrecfunction('addition3(addition3(pi^3_1, pi^3_2), pi^3_3)', 3, 2, 1)
addition3(addition3(pi^3_1,pi^3_2),pi^3_3)(3,2,1)
addition3(pi^3_1,pi^3_2)(3,2,1)
addition(5,1)
<pi^3_1|sigma(pi^3_2)>(5,1)
<pi^3_1|sigma(pi^3_2)>(5,0)
pi^3_1(5) = 5
sigma(pi^3_2)(5,0,5)
pi^3_2(5,0,5) = 5
pi^3_2(5) = 3
sigma(5) = 6 sigma(pi^3_1,pi^3_2),pi^3_3)
ans = 6
octave:6>
sigma(pi^3_2)(3,1,4)
pi^3_2(3,1,4) = 4
```

- 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.

```
X4 := 0
while X1 ≠ 0 do
    X4 := X4 + 1;
    X1 := X1 - 1;
od
while X2 ≠ 0 do
    X2 := X2 - 1;
    X4 := X4 + 1;
od
while X3 ≠ 0 do
    X4 := X4 + 1;
    X3 := X3 - 1;
od
```