Teoría de Autómatas y Lenguajes Formales

Práctica 3: Turing Machine, recursive functions and WHILE language

Santiago, Díez Cantisano

December 25, 2022

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

Ejemplo 1.1. Máquina de Turing en JFLAP

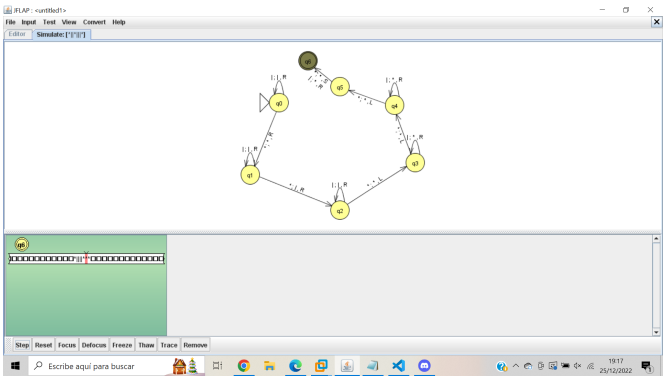

Figure 1: Máquina de Turing

2 Define a recursive function for the sum of three values

Definición 2.1. Composition composición de funciones
Let $m > 0, k \geq 0$ and the functions:
$g : N^m \to N$
$h_1, \ldots, h_m : N^k \to N$
If the function $f : N^k \to N$ is

$$f(\vec{n}) = g(h_1(\vec{n}), \ldots, h_m(\vec{n}))$$

then f is a composition of g and $h_1, \ldots, h_m$, expressed as $f(\vec{n}) = g(h_1, \ldots, h_m)(\vec{n})$, or simply $f = g(h_1, \ldots, h_m)$.

Definición 2.2. Primitive recursion recursión primitiva
Let $k \geq 0$ and the functions
$g : N^k \to N$
$h : N^{k+2} \to N$
If the function $f : N^{k+1} \to N$ is

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

then f is obtained from g and h by primitive recursion.
We will express it as $f(\vec{n}) = gh(\vec{n})$, or simply $f = gh$.

Ejemplo 2.1. $add3 = \langle\langle \pi_1^1 | successor_3 \rangle | successor_4 \rangle$

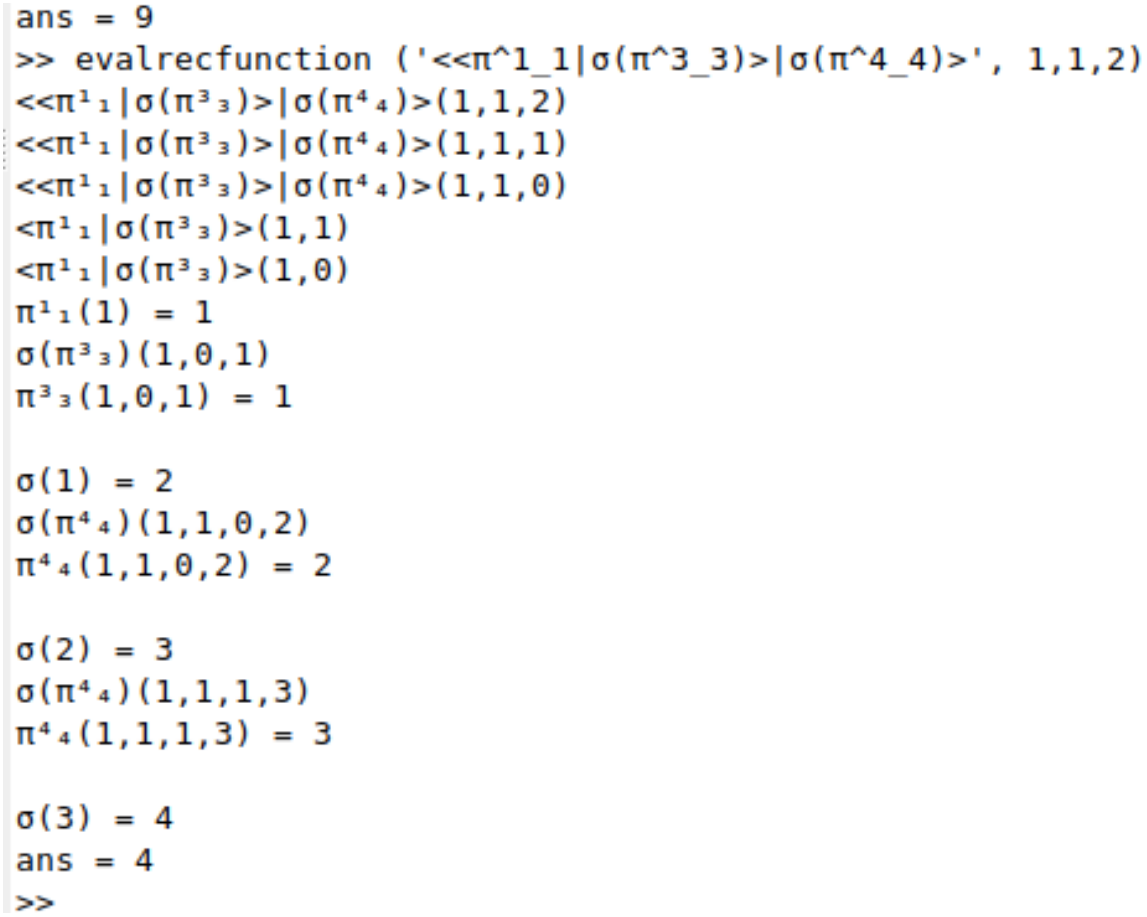

Figure 2: Suma de tres números

Ejemplo 2.2. Función recursiva en Octave

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

$X_4 := X_1;$
while $X_2 \neq 0$ **do**
    $X_4 := X_4 + 1;$
    $X_2 := X_2 - 1;$
od
while $X_3 \neq 0$ **do**
    $X_4 := X_4 + 1;$
    $X_3 := X_3 - 1;$
od
$X_1 := X_4;$