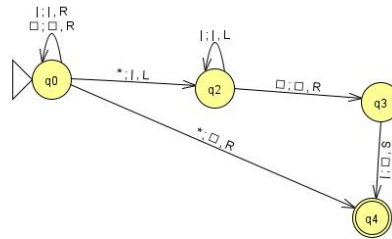


Apartado 1. Describe la máquina de Turing en JFLAP y adjuntar imagen de la misma tras comprobar su funcionamiento



Representación de la máquina de Turing. Funciona incluso cuando una de las cadenas, o ambas, son vacías.

Apartado 2. Describe en Octave una función recursiva que suma tres valores

Para definir la función recursiva, empezamos pensando en propiedades que debe tener la función $\Sigma_3(x, y, z) = x + y + z$, siendo Σ_n la función que suma n números naturales: $\Sigma_3(x, y, z) = \Sigma_3(x, y + z, 0) = \Sigma_2(x, y + z)$, y esto nos permite definir Σ_3 mediante recursión primitiva:

$$\Sigma_3(x, y, z) = \begin{cases} \Sigma_2(x, y) & z = 0 \\ \sigma(\pi_4^4(x, y, z - 1, \Sigma_3(x, y, z - 1))) & z > 0 \end{cases} \quad (1)$$

Nótese que $\sigma(\pi_4^4(x, y, z, w)) = w + 1$ es la función sucesor del cuarto argumento. Entonces,

$$\Sigma_3 = \langle \Sigma_2 | \sigma(\pi_4^4) \rangle = \langle \langle \pi_1^1 | \sigma(\pi_3^3) \rangle | \sigma(\pi_4^4) \rangle$$

Se ha descrito el autómata de forma que pueda ser reconocido por *recursiveexpression.m*, que se encuentra en el repositorio de la asignatura. Este es el resultado de la ejecución:

```
>> evalrecfunction('addition_3', 2, 1, 1)
addition_3(2, 1, 1)
<addition|sigma(pi^4_4)>(2, 1, 1)
<addition|sigma(pi^4_4)>(2, 1, 0)
addition(2, 1)
<pi^1_1|sigma(pi^3_3)>(2, 1)
<pi^1_1|sigma(pi^3_3)>(2, 0)
pi^1_1(2) = 2
sigma(pi^3_3)(2, 0, 2)
pi^3_3(2, 0, 2) = 2

sigma(2) = 3
sigma(pi^4_4)(2, 1, 0, 3)
pi^4_4(2, 1, 0, 3) = 3

sigma(3) = 4
ans = 4
>> recursiveexpression('addition_3', 'LaTeX')
ans = $<<\pi^1_1|\sigma\left(\pi^3_3\right)>|\sigma\left(\pi^4_4\right)>$
addition          <n^1_1|sigma(n^3_3)>
addition_3        <addition|sigma(n^4_4)>
```

Ejemplo de ejecución de la función recursiva y su definición en Octave

Apartado 3. Describe la función que sume tres valores en lenguaje WHILE usando el paquete whilecode

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