

Teoría de Autómatas y Lenguajes Formales

Práctica 4: Numeración de Programas y EXWHILE

Rocío Guzmán Arroyo

December 24, 2022

1 Explicación de la Práctica

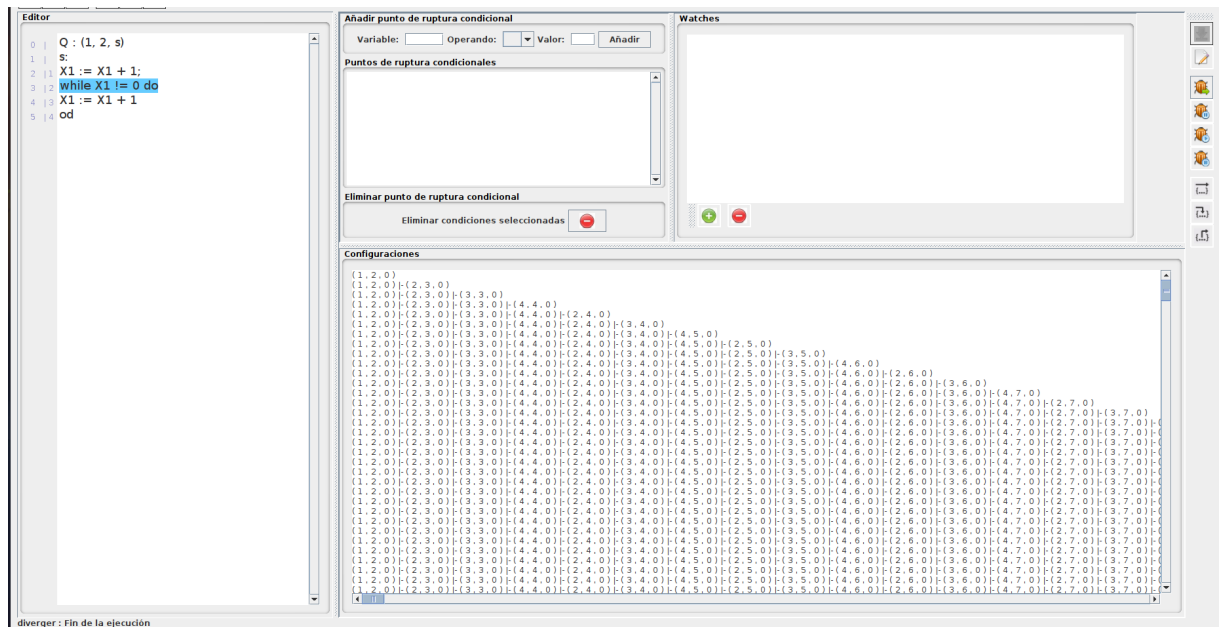
Para realizar la practica se nos pide resolver los siguientes ejercicios:

1.1 Ejercicio 1

Se nos pide el desarrollo del cálculo de la menor codificación del programa WHILE "diverger". En otras palabras, lo que se nos pide es la codificación de un código WHILE el cual nunca finaliza. Tiene iteraciones infinitas

```
Q : (1, 2, s)
s:
X1 := X1 + 1;
while X1 != 0 do
X1 := X1 + 1
od
```

Asociamos una condición que nunca se cumple por lo tanto el bucle continua ejecutandose invinitas veces. En tiempos de ejecución una función infinito.



1.2 Ejercicio 2

Se nos pide un programa en la aplicación 'Octave' que devuelva un número n de vectores que decodifica todos los posibles vectores hasta n . Para esto utilizamos el programa godeldecoding que devuelve la tupla codificada por n .

```
function code = vectoresoptenidos (n)

for i=0:n-1
disp(['(' num2str(godeldecoding(i)) ')'])
end
end
```

En este código simplemente se iterará n veces ya que va de 0 a $n-1$ vez y aplica el programa godeldecoding.m, es un programa que anteriormente ha sido implementado por el profesor.

```

>> vectoresoptenidos (5)
()
(0)
(0 0)
(1)
(0 0 0)
>>

```

1.3 Ejercicio 3

Debemos implementar un código en Octave y adjuntar la salida que proporciona de la misma forma que el código anterior simplemente en este caso lo que devolvemos son los códigos WHILEs.

```

function code = nprogramaswhile (n)

for i=0:n-1

disp(N2WHILE(i))

endfor
end

```

```

>> nprogramaswhile (7)
(0, X1=0)
(1, X1=0)
(0, X1=0; X1=0)
(2, X1=0)
(1, X1=0; X1=0)
(0, X1=X1)
(3, X1=0)

```