

# Teoría de Autómatas y Lenguajes Formales

## Práctica 4

Sara Yue Comino Márquez

16 de diciembre de 2022

### 1. Ejercicio 4.1

Create the simplest **WHILE** program that computes the *diverge* function (with zero arguments) and compute the codification of its code.

El programa WHILE.

```
 $Q = (0, s)$   
 $s :$   
     $X_2 := X_1 + 1;$   
    while  $X_2 \neq 0$  do  
         $X_1 := 0$   
    od
```

Al ejecutarlo en Octave salta el error siguiente:

```
>> CODE2N("X2:=X1+1; while X2!=0 do X1:=0 od")  
error: begdigit(1): out of bound 0 (dimensions are 1x0)  
error: called from  
    sent2N at line 25 column 16  
    CODE2N at line 38 column 18
```

Solución que debería de salir:

```
> CODE2N("X2:=X1+1; while X2!=0 do X1:=0 od")  
ans = 10876
```

## 2. Ejercicio 4.2

Create an Octave script that enumerates all the vectors.

Ejecución en Octave de la función (error con los comandos verbatim y listings, por eso están en captura).

```
>> function printNvectors(N)
for i=0:N-1
x=num2str(godeldecoding(i))
disp (x)
end
end
>> printNvectors(5)
x =

x = 0
0
x = 0  0
0  0
x = 1
1
x = 0  0  0
0  0  0
>>
```

### 3. Ejercicio 4.3

Create an Octave script that enumerates all the WHILE programs.

Al ejecutarlo en Octave salta un error (error con los comandos verbatim y listings, por eso está en captura).

```
>> function printNwhilePrograms(N)
for i=0:N-1
disp(N2WHILE(i))
end
end
>> printNwhilePrograms(1)
error: invalid empty index expression
error: called from
    N2WHILE at line 20 column 24
    printNwhilePrograms at line 3 column 1
>> |
```