

Práctica 4

Antonio Trujillo Reino

December 19, 2022

1 Actividad 1

Consideramos el siguiente código while:

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

Con la primera instrucción nos aseguramos que entre al bucle, pues todas las variables están a cero para este código sin entradas. Dentro del bucle metemos la mínima instrucción que hace que no finalice el bucle, y solo metemos una instrucción pues meterle más solo la haría más compleja. El bucle es necesario, pues de otra manera no podremos hacer que diverja. Una vez tenemos el código, pasamos a calcular su codificación. Para ello, usamos octave que nos calculará lo que queremos:

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

2 Actividad 2

```
function actividad2()  
    i=0  
    while(i>=0)  
        disp(['(' num2str(godeldecoding(i)) ')'])  
        i=i+1  
    end  
end
```

3 Actividad 3

```
function actividad3()  
    i=0  
    while(i>=0)  
        disp(N2WHILE(i))  
        i=i+1  
    end  
end
```

```
>> actividad2()
i = 0
()
i = 1
(0)
i = 2
(0 0)
i = 3
(1)
i = 4
(0 0 0)
i = 5
(1 0)
i = 6
(2)
i = 7
(0 0 0 0)
i = 8
(1 0 0)
i = 9
(0 1)
i = 10
(3)
i = 11
(0 0 0 0 0)
i = 12
(1 0 0 0)
i = 13
(0 0 1)
i = 14
(2 0)
i = 15
(4)
i = 16
(0 0 0 0 0 0)
i = 17
(1 0 0 0 0)
i = 18
(0 0 0 1)
```

```

>> actividad3()
i = 0
(0, X1=0)
i = 1
(1, X1=0)
i = 2
(0, X1=0; X1=0)
i = 3
(2, X1=0)
i = 4
(1, X1=0; X1=0)
i = 5
(0, X1=X1)
i = 6
(3, X1=0)
i = 7
(2, X1=0; X1=0)
i = 8
(1, X1=X1)
i = 9
(0, X1=0; X1=0; X1=0)
i = 10
(4, X1=0)
i = 11
(3, X1=0; X1=0)
i = 12
(2, X1=X1)
i = 13
(1, X1=0; X1=0; X1=0)
i = 14
(0, X1=X1; X1=0)
i = 15
(5, X1=0)
i = 16
(4, X1=0; X1=0)
i = 17
(3, X1=X1)
i = 18

```