Solutions of the practice lessons of
Theory of automata and
Formal languages

Computer Science and Computer Language Department
University of Málaga

Carlos Gálvez Bravo

December 26, 2022

Practice 4

Program Numbering and EXWHILE

4.2 Coding and Decoding

3. The result of encoding these vectors is different with Cantor that with Gödel. The encoding of $(1, 0, 0, 0, 0)$ with Gödel (the result is 17) is the same as encoding $(3, 2)$ with Cantor.

```
Command Window
>> cantorencoding(3,2)
ans =   17
>> cantorencoding(1,0,0,0,0)
ans =   1
>> godelencoding(3,2)
ans =   189
>> godelencoding(1,0,0,0,0)
ans =   17
>> |
```

4. The main difference between Cantor and Gödel is that Gödel is defined in $\mathbb{N}^*$, that is, all the vectors of any dimension are codified with the same function. However, Cantor is defined for a specific dimension $k \in \mathbb{N}$, so there are different Cantor both encoding and decoding functions in $\mathbb{N}^k$, for each k.

4.3   Extended WHILE

1. No. Since a WHILE program needs the specification of the number of arguments, it is impossible to create a unique EXWHILE code for different number of arguments. This is possible in Octave.

2. The program we need is the same as the original:

```
Prueba
entradas:ent1
salida:    result
metodo:

aux:=SigmaSuper1Sub21(ent1); primero:=SigmaSuper1Sub21(aux)
segundo:=SigmaSuper1Sub22(aux);
tercero:=SigmaSuper1Sub22(ent1);
cod1:=SigmaSuper2Sub1(primero,segundo);
result:=SigmaSuper2Sub1(cod1,tercero)
```

Activities

1. The simplest WHILE program without arguments that computes the *diverge* function is:

$$Q = (0, s)$$
$$s :$$
$$X_2 := X_1 + 1;$$
$$\textbf{while } X_2 = 0 \textbf{ do}$$
$$X_1 := 5$$
$$\textbf{od}$$

And the codification of the code s is:

> CODE2N("X2:=X1+1; while X2!=0 do X1:=5 od")

2. We know that we can establish a biyection between all the vectors and $\mathbb{N}$, so we only need a program with a loop that can print all the set of vectors. The following code prints the N first vectors:

Function printNvectors(N)

for i=0:N-1

Disp([´(´ num2str(godeldecoding(i)) ´)´])
end

end

3. This case is very similar to activity 2 since there exists a bijection between WHILE programs and $\mathbb{N}$, so the Octave script is:

Function printNwhilePrograms(N)

for i=0:N-1

Disp(N2WHILE(i))
end

end