

```
1 -----FIBONACCI-----
2 (deffunction fibonacci(?n)
3 (if ( or (= ?n 0) (= ?n 1))then
4 ?n
5 else
6 (+ (fibonacci(- ?n 1)) (fibonacci(- ?n 2)))
7 )
8 )
9
10
11 (deffunction fibonacciSerie(?numeroInicial ?numeroFinal)
12 (printout t (fibonacci ?numeroInicial) crlf)
13 (if (< ?numeroInicial (- ?numeroFinal 1)) then
14 (fibonacciSerie (+ ?numeroInicial 1) ?numeroFinal)
15 ))
16
17 (printout t (fibonacciSerie 0 10))
18
19 ----- PRIMO-----
20
21 (deffunction primoRecurso(?numero ?d)
22 (if(< (/ ?numero 2) ?d)then
23 (return TRUE)
24 else
25 (if (= (mod ?numero ?d) 0)then
26 (return FALSE)
27 else
28 (return(primoRecurso ?numero (+ ?d 1))))))
29
30
31 (deffunction verificarPrimo(?numero)
32 (if(primoRecurso ?numero 2) then
33 (printout t (str-cat "EL NUMERO " ?numero " ES PRIMO ") crlf)
34 else
35 (printout t (str-cat "EL NUMERO " ?numero " NO ES PRIMO ") crlf)
36 ))
37
38 (printout t (verificarPrimo 4))
39 (printout t (verificarPrimo 25))
40
41
```