

Ejercicio 2

a) pred esDescompresion (cod:Codigo, pal:Palabra) {
 $|pal| = \text{sumaCaracteresAnteriores}(\text{cod}, |\text{cod}|) \wedge$
 $(\forall k: \mathbb{Z}) (0 \leq k < |\text{cod}| \rightarrow$
 $(\forall j: \mathbb{Z}) (0 \leq j < |\text{cod}[k]_0| \rightarrow$
 $(\forall rep: \mathbb{Z}) (0 \leq rep < \text{cod}[k]_1 \rightarrow$
 $\text{cod}[k]_0[j] = \text{pal}[\text{sumaCaracteresAnteriores}(\text{cod}, k)$
 $+ j + (rep * |\text{cod}[k]_0[j]|)]$
 $)))$
 $}$

espectacular!!

aux sumaCaracteresAnteriores (cod, i): $\mathbb{Z} =$

$$\sum_{k=0}^{i-1} |\text{cod}[k]_0| * \text{cod}[k]_1$$

b) procu comprimir (in pal: Palabra, out cod: Codigo) {
 Pre {true}
 Post { esDescompresion (cod, pal) \wedge noHayFragmentosVacuos(cod)
 \wedge noHayRepeticionesNulas (cod) }
 $}$

bien, pero tenés que poner esDesc al final
 y un $\wedge L$ porque si $\text{cod}[i]_1 < 0$ se indefin

pred noHayFragmentosVacuos (cod: Codigo) {
 $(\forall k: \mathbb{Z}) (0 \leq k < |\text{cod}| \rightarrow |\text{cod}[k]_0| > 0)$
 $}$

pred noHayRepeticionesNulas (cod: Codigo) {
 $(\forall k: \mathbb{Z}) (0 \leq k < |\text{cod}| \rightarrow \text{cod}[k]_1 > 0)$
 $}$

c)

~~proc optimizer Código (inout: Cód~~

proc optimizer Código (inout cod: Código) {

Pre { cod = CODo } ✓

Post { (∃pal: Palabra) (esDescompreson (pal, CODo) ∧

esDescompreson (pal, cod)) ∧ ✓

costo (cod) ≤ costo (CODo)

f

}

Para que sea óptimo hay que pedir que todo código alternativo tenga costo \geq