## Coloquio AYED

- Considerá la función queHace :: Int  $\rightarrow$  Int  $\rightarrow$  Int  $\rightarrow$  Int especificada como queHace.b.a.xs =  $\langle$  Max p, q :  $0 \le p \le q < \#xs \land \langle \forall i : p \le i < q : b < xs!i < a \rangle : q p \rangle$
- a) describí con tus palabras lo que hace la función
- b) transformá esa especificación en una que use segmentos
- c) derivá queHace (podés usar cualquiera de las dos especificaciones).
- 2) La función calcula el largo del mayor segmento de XS (sin tener en cuenta nunca el ultimo elemento) que cumpla que todos los elementos de dicho segmento estendentro del intervalo abierto (b,a).
- b) Como la especificación nunca toma en cuenta al ultimo elemento de la lista, para transformar la especificación en una que use segmentos podemos hacer que el segmento final de xs sea no vario. De tal modo que nos quedaria la siguiente especificación:

queffere.b. a. 
$$xs = \langle Max bs, \omega, ds : xs = bs + t \omega + t ds \wedge ds \neq [] \wedge \langle \forall i : 0 \leq i \leq \#\omega : b < \omega : i < a \rangle : \#\omega \rangle$$

c) Anter de proceder a derivar quettace derivemos la especificación esta En Rango. b. a.  $xs = \langle \forall i : 0 \le i < \# xs : b < xs : i < a \rangle$ , con el fin de reemplazar la en quettace y así facilitar la escritura en el desarrollo de la derivación.

Derivation esta Endango.b.a.xs:

Caso base: XS:=[7]estaEnRango.b.a.[7]  $= \{Especification 3 \\ \{\forall i: 0 \le i \le \#[7:b \le [7]: i \le a\}\}$   $= \{Def. cordinal \}$ 

```
( \ti: 0 \le i < 0 : b < []! i < a >
= E Evalvo rango 3
 ( \ti: False: b < []! i < a >
= ERANGO VALIO 3
  True
Caso Inductivo: X5:=(x >x5)
esto EnRongo.b.o. (x > x5)
= { Esperificación 3
  ( \ti: 0 \le i < \#(x DXS) : b < (XDXS) ! i < a >
= EDef dercordinal 3
 ( \ti: 0 \( i \) ( \( 1 + \) \( x \) : \( b \) ( \( x \) \( x \) \( i \) \( a \)
= E Aritmético 3
  ( \fi: i=0 v 1 \le i < 1+ \fix5 : b < (x0x5) ! i < 2 \fi
= E Partición de rango 3
 ⟨\forall i: i=0: b < (xoxs) ! i < a > ∧ ⟨\forall i: 1 ≤ i < 1+ \forall xs : b < (xoxs) ! i < a >
= Elango unitario 3
 b < (xox5)!0 < 0 ∧ (∀i:1 ≤ i < 1+ *x5 : b < (xox5)!i < 0 >
= { Def ! 3
  b < x < 2 1 ( \forall i: 1 \le i < 1+ \forall x5 : b < (x0x5) ! i < a >
= E Cambio de variable i ← i+1 3
  b < x < 20 1 (∀i:1 ≤ i+1 < 1+ *x5 : b < (x0x5) !i+1 < 0 >
= E Aritmetico 3
 b ∠ x ∠ o n ( \ti: 0 ≤ i < \pre> \pre> x5: b < (x0x5) ! i+1 ∠ o >
= { Aritmetico y Def! 3
b < x ∧ x ≥ 3 ∧ (\fi: 0 ≤ i < \fix5: b < x5! i < 3)
= { HI 3
b = x 1 x = o1 esta En Rango. b. a. xs
```

```
Resultado final esta Endango:
 esta En Rango. b. a. [] = True
 esta Enhango.b.a. (xxxx) = b < x n x < a n esta Enhango.b.a. xx
Teniendo en cuenta esto, la especificación de que Hace quedario de la sigurente forma:
quefface.b.o.xs = < Max bs, w, ds: xs=bs+t cs+t dr n ds = [] n
                          esto En Romo. b.o. w: #w>
                        Derivacion que Hace. b.a.xs
(aso base: XJ:=[]
gueHace.b.a.[]
= E Experificación 3

∠Max br.cs, dr: []=br++cr++dr n dr ≠ [] n extoEndango.b.a.cs: #cs>

= EPropiedad de Listar 3

∠Max br.cs, dr: []=brn []=crn[]=drndr≠[] n extoEndango.b.a.cr: #cs>

= Elrinigio de no contradicción 3
Max br.cs, dr: []=brn []=crn Falsen extaEndango.b.a.cs: #cs>
= Elemento absorbente con juncion 3

\( \text{Max br, cs, dr: Fabre: #cs} \)

= ERango valio 3
-00
Caso inductivo: XJ:= (X >XJ)
HI: quettere.b. a. xs = < Max bs, cs, ds: xs = bs ++ cs ++ dr \ ds \ \neq [] \ \Lambda
                                esto En Romo. b.o. w: #w>
```

```
queHave.b. a. (xbx5)
= EEsperificación 3

∠Max br, cs, dr: (x xx)=bx++cx++dx n dx ≠ [] n extaEndango.b.a.cs: #cs>

= EElemento neutro conjunción y tercero excluido 3
(Max br. cs, dr: (bs=[]vbr=[]) n (x =xs)=br++cr++dr n dr = [] n esta Endanço. b.a.cs: #cs)
= EDistributivided conjunción dispunción 3

∠Max br, cs, ds: (bs=[] n (x xx)=bx++cs++dx n ds ≠[] n extatn@ango.b.a.cs) V

               (bs $[] n (x xx)=bs++cs++ds n ds $[] n estaEndango.b.a.cs): #cs>
= Elarteción de ranço 3

∠Max br.cs, dr: bs=[] n (x xx)=b+cs+ds n ds ≠[] n estaEndango.b.a.cs: #cs> max

∠Max br.cs, ds: bs ≠ [] n (x ×xs) = bs ++ cs ++ ds n ds ≠ [] n estatendango. b.a.cs: #cs >

= Ellememos E a la primer expression 3
  E max < Max br. cs, di: bs + [] n (x xx) = bx ++ cx ++ dr n ds + [] n extaEndango. b.a.cs: #cs>
= {Cambio de variable bs < Kobs3
E max < Max K, br, cs, dr: Kobs + [] n (x xx) = (Kobs) ++ cr ++ dr n ds + [] n esta Endanço. b.a.cs: #cs>
= { Logice y elemento neutro conjuncion 3
E max < Max K, br, cs, ds: (x xxs)=(k+bx)++cs++ds n ds + [] n esta Enlango. b.a.cs: #cs>
= { Concatenación x2 3
E max < Max K, br, cs, ds: (x xx) = Ko(bs++cs++ds) n ds + EI n estaEndango. b.a.cs: #cs>
= E Propiedad de livtas 3
E max < Max K, br, cs, dr: x=K x x = br++cr++dr x dr + [] x extatendango.b.a.cs: #cs>
= {Eliminación de variable K3
E max < Max br.cs.ds: xs = botter+don ds ≠ [] n esta Endança.b.a.cs: #cs>
= EHI3
Emax que Hace. b. a. XS
= EREEMPlazo E3

∠Max br.cs, dr: bs=[] n (x >xs)=br++cr++dr n ds ≠ [] n extaEndango.b.a.cs: #cs> max

   gueltace.b.a.xs
= { Elimination de variable bs }
```

```
∠Max cs, ds: (x xx)=[7++cx++dx n ds ≠ [] n estatendango.b.a.cs: #cs > max ved+zue.b.a.xs

= E Def convotención 3

∠Max cs, ds: (x xs)= cs++ds n ds ≠ El n esta Enlanço. b.a.cs: #cs> max queltace. b.a.xs

= E Modularizamos que Hace 2.b.a. xs 3
    queHace2.b.a.(x xs) max queHace.b.a.xs
                                  Derivación que Have 2. b.a.xs
 queltace 2.6.2.xs = < Max cs, ds: xs = cs++ds n ds + [] n estatendango.6.2.cs: #cs>
Caro bare: XJ:=[]
gue Hove 2.6.2. []
= EEsperificación 3
< Max cs, dr: [] = cs++ds n ds ≠ [] n esta Endanço. b.a.cs: #cs>
= E Propiedad de listes 3

∠Max cs, ds: []=cs 1 []=ds 1 ds ≠ [] 1 esta Endango. b.a.cs: #cs >

= { Principio de no contradición 3
< Max cs, di: []= cs 1 False 1 esta Endanço. b. a.cs: #cs >
= { Elemento absorbente conjunción 3

= { Rango voixo 3
 - 00
Caso inductivo: X5:= (X > X5)
 HI: gueltace2.b.a.xs = < Max cs, ds: xs=cr++ds n ds + El n esta Enlango.b.a.cs: #cs>
queHoce2.b. a. (xxxs)
= Especificación 3

\[
\text{Max cs, ds: (x \times x s) = cs ++ ds n ds \neq EI n esta \text{Enlange. b. a.cs: #cs}
\]

₹ Neutro conjuncion y tercero excluido 3
```

```
∠Max cs, ds: (cs=E]vcs≠E]) ∧ (xo xs=cs+ds n ds≠E] n estateldango.b.a.cs: #cs>

= EDistribution disjunción disjunción 3

∠Max cs, ds: (cs=E] ∧ (xo xs)= cs++ ds ∧ ds ≠ E] ∧ estaEndango. b.a.cs) ∨

                                        (cs ≠ [] 1 (x × xs)= cs ++ ds 1 ds ≠ [] 1 esto Endanço. b. a.cs): #cs>
= ¿Paticion de rango 3

  \[
  \lambda \text{cs, ds: cs=EI \( \lambda \text{x} \text{x} \rangle = \text{cs+ds \( \text{ds \) \end{\( \text{ds \( \) \) \} \end{\( \text{ds \( \) \) \} \end{\( \text{ds \( \) \) \} \} \ext{ds \( \text{ds \( \text{ds \( \text{ds \( \text{ds \( \) \) \) \} \end{\( \text{ds \( \text{ds \( \) \) \end{\( \text{ds \( \) \) \end{\( \) \\ \end{\( \) \\ \) \} \end{\( \text{ds \( \) \) 

\[
\left\) Max cs, ds: cs \(\xi\) \(\left\) \(\left\) \(\xi\) \(\x
= Ellememos E a la segunda expresión 3

∠Max cs, ds: cs=[] ∧ (xo xs)=cs++ds ∧ ds ≠ [] ∧ estaEndan o.b.a.cs: #cs> max E

= Eliminación de variable cos

∠Max ds: (xo xs)=[]++ds n ds ≠ [] n extoEndan o.b.a.[]: #[] max E

= { Def. cardinal }

∠Max ds: (xo x5=[]++ ds n ds ≠ [] n extoEndan o.b.a.[7:0) max E

= € Termino constante 3
        0 max E
= Eleemplozamos E 3
      0 max < Max cs, ds: cs + E] n (xxx)=cs++ ds n ds + E] n esta Endango. b.a.cs: #cs>
= { Cambio devariable (5 < (1 > cs) }
     0 max < Max c, cs, ds:(c > cs) ≠[] ∧ (x × x)=(c > cs) ++ ds ∧ ds ≠ [] ∧ esta Endanço. b.a. (c > cs): *(c > cs)
= ¿ Lógico y elemento neutro conjunción }
     0 max < Max c, cs, ds: (xxxs)=(cxcs)++ds n ds + [] n estatendango.b.a.(cxcs): **(cxcs) >
 = { Def. concatenación }
  0 max < Max c,cs,ds: (xoxs)=co(cs++ds)nds≠[] n estaEndango.b.a. (cocs): *(cocs)
 = E Propiedad de listar 3
      0 max < Max c, cs, ds: x=c, x5=cs++ds n ds + E1 n estatellange. b.a. (cocs): #(cocs)>
= Elimination de variable 3
      0 max < Max cs, ds: XS=Cs++ds n ds ≠ [] n estaEndango. b.a. (xo cs): #(xocs)>
 = E Def. rewround esta Endanço 3
  0 max < Max cs, ds: x5=cs++ds n ds≠[] nb=x n x=a n estaEndango.b.a.cs: *(xocs)>
```

```
= E Def. cordinal 3
0 max Chaxcs, ds: X5=cs++ds n ds + [] n b = x n x = a n estatendango. b.a.cs: 1+ #cs)
: 20262 709 civilenA
Laso 1: bex 1 x20 = True
0 max < Max cs, ds: x5=cs++ds n ds + [] n b = x n x = a n estatellango. b.a.cs: 1+ #cs >
= Epor hipotesis 3
0 max < Max cs, ds: x5=cs++ds n ds≠[] n True n esta Endango. b.a.cs: 1+ #rs>
= E Elemento nevtro conjunción 3
0 max < Max cs, ds: X5=cs++ds n ds ≠ EIn estaEndango. b.a.cs: 1+#cs>
= E Distributivided Max 3
0 max (K Max cs, ds: XS=cs++ ds n ds + [] n estatendango.b.a.cs: #cs } +1
= £ HI 3
0 max (queHace 2. b. o. x5 +1)
Caso 2: bex 1 xed = false
0 max < Maxcs, ds: x5=cs++ds n ds≠[] n b = x n x = a n esta Endango. b.a.cs: 1+#cs}
= Epor hipotesis3
0 max < Max cs, ds: X5=Cs++ds n ds≠[] n False n estatendango.b.a.cs: 1+#cs>
= E Elemento > buorbente conjunción 3
0 max < Max cs, ds: False: 1+#cs>
E Rango vavio 3
0 max -00
= Elemento neutro max 3
```

## Resultado Final:

que Have: int  $\rightarrow$  int  $\rightarrow$  [int]  $\rightarrow$  int que Have. b.a.  $\mathbb{C} \mathcal{I} = -\infty$ 

que Hore. b.o. (x xx) = queHore2.b. o. (x xx) mox queHore.b.o. xx

donde

queHaie2. b.a. [7 = -00

queHace2.b.a.(xxxs) = (bcx 1 xca  $\rightarrow 0$  max (queHace2.b.a.xs +1)  $\Box \gamma(bcx 1 xca) \rightarrow 0$