Segunda parte (ex.4)

De EjerciciosLMF2014

```
header {* Examen 4 *}

theory ex4_sol
imports Main
begin


text {* ---------------------------------------------------------------------
  Ejercicio 2: (3 puntos) Demostrar
  { ∃x. ∀y. B(y) ⟶ A(x,y), ∃x. ∀y. (∀z. B(z) ⟶ A(y,z)) ⟶ ¬S(x,y) } ⊢
  ∃x. ¬(∀y. S(x,y))
  ------------------------------------------------------------------- *}

lemma ej_2:
  assumes "∃x. ∀y. B(y) ⟶ A(x,y)"
          "∃x. ∀y. (∀z. B(z) ⟶ A(y,z)) ⟶ ¬S(x,y)"
  shows "∃x. ¬(∀y. S(x,y))"
proof -
  obtain p where 1: "∀y. B(y) ⟶ A(p,y)" using assms(1) ..
  obtain q where "∀y. (∀z. B(z) ⟶ A(y,z)) ⟶ ¬S(q,y)" using assms(2) ..
  hence "(∀z. B(z) ⟶ A(p,z)) ⟶ ¬S(q,p)" ..
  hence "¬S(q,p)" using 1 ..
  have "¬(∀y. S(q,y))"
    proof
      assume "∀y. S(q,y)"
      hence "S(q,p)" ..
      with `¬S(q,p)` show False ..
    qed
  thus "∃x. ¬(∀y. S(x,y))" ..
qed

text {*
  ---------------------------------------------------------------------
  Ejercicio 3: (4 puntos) Consideremos el árbol binario definido por
     datatype 'a arbol = H  | N "'a" "'a arbol" "'a arbol"

  Por ejemplo, el árbol
          e
         / \
        /   \
       c     g
      / \   / \

  se representa por "N e (N c H H) (N g H H)".

  (a) Definir la función
         fun nNodos :: "'a arbol ⇒ nat"
  tal que (nNodos a) es el número de nodos de a. Por ejemplo,
     nNodos (N e (N c H H) (N g H H)) = 3
  ---------------------------------------------------------------------
*}

datatype 'a arbol = H  | N "'a" "'a arbol" "'a arbol"

fun nNodos :: "'a arbol ⇒ nat" where
  "nNodos H         = 0"
| "nNodos (N x i d) = 1 + (nNodos i) + (nNodos d)"
```

```
text {*
  ---------------------------------------------------------------------
  (b) Definir la función
         fun nHojas :: "'a arbol ⇒ nat"
  tal que (nHojas a) es el número de hojas de a. Por ejemplo,
     nHojas (N e (N c H H) (N g H H)) = 4
  ---------------------------------------------------------------------
*}

fun nHojas :: "'a arbol ⇒ nat" where
  "nHojas H          = 1"
| "nHojas (N x i d) = (nHojas i) + (nHojas d)"


text {*
  ---------------------------------------------------------------------
  (c) Probar que el número de hojas de un árbol es el número de nodos
  más 1.
  ---------------------------------------------------------------------
*}

lemma "nHojas a = (nNodos a) + 1"
by (induct a) auto

lemma "nHojas a = (nNodos a) + 1" (is "?P a")
proof (induct a)
  show "?P H" by simp
next
  fix x i d
  assume HI1: "?P i"
  assume HI2: "?P d"
  show "?P (N x i d)"
  proof -
    have "nHojas (N x i d) = (nHojas i) + (nHojas d)" by simp
    also have "... = (nNodos i) + 1 + (nNodos d) + 1" using HI1 HI2 by simp
    also have "... = (nNodos i) + (nNodos d) + 1 + 1" using HI1 HI2 by simp
    also have "... = (nNodos (N x i d)) + 1" by simp
    finally show ?thesis by simp
  qed
qed

end
```