

FUNDAMENTOS DE COMPUTACIÓN
TRABAJO ENTREGABLE 6
JUNIO 2022

Este trabajo tiene un puntaje de 6 puntos y debe ser realizado en forma **INDIVIDUAL**.
Se debe subir a Aulas antes del día 12/6 a las 21:00 hs.

(1) Defina, sin utilizar funciones auxiliares, las siguientes funciones:

a) `elim :: (a -> Bool) -> [a] -> [a]`, tal que `elim p l` borra todos los elementos de la lista `l` para los cuales se cumple el predicado `p`. Si no hay ningún elemento de `l` que cumpla `p`, debe devolver `l` sin modificar.

Ejemplos: `elim (>3) [1,5,8,7,2,9,3] = [1,2,3]`
`elim not [True,False,False] = [True]`

b) `count :: (a -> Bool) -> [a] -> N`, tal que `count p l` calcula la cantidad de elementos de la lista `l` para los cuales se cumple el predicado `p`.

Ejemplos: `count (>3) [1,5,8,7,2,9,3] = S(S(S(S 0)))`
`count id [False,True,False] = S 0`

(2) Demuestre que $(\forall p :: a \rightarrow \text{Bool}) (\forall l :: [a]) \text{length} (\text{elim } p \ l) + \text{count } p \ l = \text{length } l$, donde `length :: [a] -> N` y `(+) :: N -> N -> N` son las funciones definidas en clase. Puede utilizar las propiedades de la suma vistas en clase, enunciándolas como lemas y sin necesidad de demostrarlos.

(3) Considere la siguiente función:

```
f = \x y h -> case y of { [] -> case h x of { True -> h x || x;  
                                                False -> h x || not (h x)};  
  z:zs -> case z of { 0 -> f x zs h;  
                     S k -> f x (k:zs) h}}
```

Donde `(||) :: Bool -> Bool -> Bool` es la disyunción booleana y `not :: Bool -> Bool` es la negación definidas en clase.

a) Dé el tipo de `f`.

b) Demuestre que $(\forall x :: \dots) (\forall y :: \dots) (\forall h :: \dots) \ f \ x \ y \ h = \text{True}$.