Departamento de Computación
FCEFQyN, Universidad Nacional de Río Cuarto
Asignatura: Programación Avanzada
Primer Cuatrimestre de 2023

Práctico 8: Ejercicios de Lógica

El objetivo de estos ejercicios es ayudar a trabajar sobre lógica de Primer Orden a través de programación funcional. Los perfiles de las funciones, que se deben respetar en su totalidad, están accesibles por medio del **GitHub Classroom:** https://github.com/ProgAv-UNRC/Pract8Logica

NOTA: Recomendamos resolver esta práctica de **manera grupal**, reflejando la colaboración de cada integrante del grupo, por medio de **commits significativos**, en un repositorio propio que uno de los integrantes gestionará.

1. Definir la función nand a b = not (a && b) en Haskell sin utilizar *not* y &&.

2. Definir en Haskell la función

```
maj :: Bool -> Bool -> Bool -> Bool
--retorna True sii al menos 2 argumentos son True
```

3. En Haskell un predicado sobre un tipo A es una función p :: A -> Bool, por ejemplo:

```
even :: Int -> Bool
even x = x `mod` 2 == 0
```

Se puede pensar como un predicado sobre números cuya variable libre es x. Además en Haskell tenemos las siguientes funciones que operan sobre listas de booleanos:

```
and :: [Bool] -> Bool
-- retorna True sii todos los elementos son True
or :: [Bool] -> Bool
-- retorna True sii al menos un elemento es True
```

Con estos dos operadores y listas por comprensión podemos escribir una versión ejecutable de los cuantificadores en Haskell. Por ejemplo el siguiente cuantificador:

$$(\forall i : 0 \leq i < \sharp xs : even\ xs.i)$$

Puede escribirse literalmente como:

```
and [ even xs !! i | i <- [0..(length xs)-1]]
```

Pero en Haskell lo más común es escribirlo de la siguiente forma:

```
and [ even x | x <- xs]
```

Utilizar estas ideas para escribir los siguientes cuantificadores:

- $(\exists i : 0 \leq i < \sharp xs : p\ xs.i)$
- $(\forall i : 0 \leq i < \sharp xs : p\ xs.i)$

Para un predicado p dado.

4. Utilizando las ideas asociadas a listas por comprensión, y las funciones sum, product, y length, escribir los cuantificadores de sumatoria, productoria y contatoria para ejemplos concretos.