

Ejercicio semanal 1

Profesor: Noé Salomón Hernández Sánchez
Ayudante de Teoría: Carlos Naranjo Robledo
Ayudante de Teoría: Francisco Emmanuel Anaya González
Ayudante de Laboratorio: José Nieves Morán Silva

El objetivo de este ejercicio semanal sera implementar las sustitución en la lógica de predicados.

El tipo de datos correspondiente en Haskell para la lógica de predicados es la siguiente:

```
data Term = VarP String
          | Func String [Term]

data Pred = Pr String [Term]
          | Neg Pred
          | Conj Pred Pred
          | Disj Pred Pred
          | Imp Pred Pred
          | Eqv Pred Pred
          | All String Pred
          | Ex String Pred
          deriving (Show)
```

Para esta práctica considere lo siguiente:

- Utilizaremos como referencia la nota 6.
- El tipo de datos que utilizaremos para representar a las sustituciones será el siguiente:

```
type Sub = (String ,Term)
```

- Para los primeros dos ejercicios recuerde la definición de variables libres y ligadas que aparece en la misma nota, y que hace referencia al árbol de análisis sintáctico de una fórmula.

1 (2 puntos)

Definir una función que devuelva las variables libres de una formula.

2 (2 puntos)

Definir una función que devuelva las variables ligadas de una formula.

3 (2 puntos)

Definir una función que aplique la sustitución sobre términos.

4 (2 puntos)

Definir una función que tome una formula φ y devuelva una formula α -equivalente a φ .

5 (2 puntos)

Definir una función que aplique la sustitución sobre formulas.