

# UNIVERSIDAD NACIONAL AUTÓNOMA DE MÉXICO

## Facultad de Ciencias

Integrantes:  
Marco Silva Huerta  
Adrián Aguilera Moreno



### Lógica Computacional

## Práctica 3

- `alcance`: Hacemos recursión en los cuantificadores para determinar el alcance de las variable que esta reconociendo y concatenamos con la formula.
- `bv`: Para obtener las variables ligadas haces recursión sobre los cuantificadores y vamos concatenando. Usamos una función auxiliar para eliminar variables repetidas.
- `fv`: Para obtener las variables libres primero obtenemos todas las variables haciendo recursión en los operadores y cuantificadores eliminando los repetidos que van saliendo. Para eso usamos la primera opción auxiliar que es separar la variable de su termino o bien usando otra función auxiliar quitamos las variables de un conjunto de términos. Finalmente obteniendo la lista de todas las variables hacemos la operacion diferencia con las variables ligadas para así obtener las variables libres.
- `sustTerm`: Función que realiza la sustitución de variables en algún término. Se emplea la función auxiliar `sustTermAux` que hace la sustitución en un conjunto de términos.
- `sustForm`: Por medio de patrones se realiza la identificación de casos base y recursivos, así definimos los casos en los que se puede o no aplciar la sustitución. En algunos casos se segmenta usando `if` y `else` que nos proporciona Haskell.
- `alphaEq`: Se emplean algunas funciones auxiliares definidas con patrones (por lo que esta función esta definida con patrones). Las funciones son explicadas en el archivo de la práctica (están comentadas las auxiliares). Por esto, no se incluyen las funciones auxiliares.

Matriculas:

1. Marco Silva Huerta: 316205326.
2. Adrian Aguilera Moreno: 421005200.