



# UNIVERSIDAD NACIONAL AUTÓNOMA DE MÉXICO

## FACULTAD DE CIENCIAS

### LÓGICA COMPUTACIONAL



Profesora: ESTEFANÍA PRIETOS LARIOS  
Ayudante: HERNÁNDEZ OLVERA MAURICIO E.  
Ayudante Lab: SALAZAR GONZÁLEZ EDWIN MAX

Fecha de Entrega: Jueves 21 de Febrero, 2019. 13:59

## PRÁCTICA 1. Parte 2.

### RECURSIÓN.

#### OBJETIVOS

- Aplicar conceptos básicos del lenguaje de programación Haskell.
- Aplicar el concepto de recursión en un lenguaje funcional como lo es Haskell.

#### INSTRUCCIONES

Descargar el archivo *Practica1p2.hs* y resolver los ejercicios definidos sobre éste.

#### EJERCICIOS

##### Función **permutaciones**

Recibe una lista de elementos y devuelve todas las permutaciones de los elementos de una lista.

```
*Practica1p2> permutaciones ['a', 'b', 'c']  
["abc", "bac", "bca", "acb", "cab", "cba"]
```

##### Función **factores**

Recibe un entero y regresa la lista de sus factores.

```
*Practica1p2> factores 100  
[1,2,4,5,10,20,25,50,100]
```

### Función **perfectos**

Recibe un entero y devuelve la lista de números perfectos que se encuentran hasta n.

```
*Practica1p2> perfectos 25
```

```
[6]
```

### Función **ternasPitagoricas**

Recibe un número y regresa la terna pitagórica que corresponda.

```
*Practica1p2> ternasPitagoricas 12
```

```
[(3,4,5),(4,3,5),(6,8,10),(8,6,10)]
```

### Función **isSubSet**

Recibe dos listas y nos dice si la primer lista es subconjunto de la segunda.

```
*Practica1p2> isSubSet [2,3,4] [1,2,3,4,5,6]
```

```
True
```

### Función **deleteT**

Elimina un elemento de un árbol binario (5pts).

```
*Practica1p2> let tree = (Branch 2 (leaf 1) (leaf 3))
```

```
*Practica1p2> delete tree  
Branch 1 Empty (Branch 3 Empty Empty)
```

### Función **balanced**

Recibe un árbol binario y nos dice si esta balanceado o no.

```
*Practica1p2> let tree = (Branch 'A' (leaf 'B') Empty)
```

```
*Practica1p2> balanced tree  
True
```

## Función **pre**

Regresa la lista que se obtiene al recorrer el árbol en pre-orden.

```
*Practica1p2> let tree = (Branch 'A' (leaf 'B') (leaf 'C'))
```

```
*Practica1p2> pree tree  
"ABC"
```

## ESPECIFICACIONES

- ✓ Respetar las firmas de las funciones.
- ✓ Todas las funciones deberán estar documentadas.
- ✓ La legibilidad y documentación tendrá un impacto sobre la calificación de la práctica.
- ✓ La práctica se podrá realizar en equipos de a lo más 3 personas.
- × Plagiar parcial o totalmente una trabajo es castigado por el reglamento Universitario de la UNAM, por fa no lo hagan.
- × Cualquier práctica entregada posterior a la fecha límite no será tomada en cuenta.
- ×

Se deberá contar con un directorio cuyo nombre sea Practica1. Dentro del directorio se debe tener:

- README.txt, donde se incluya número de cuenta y comentarios sobre la práctica.
- Practica1p2.hs, script requerido para ésta práctica.

Comprimir el directorio con el formato ApellidoNombreP1p2. Comprimir con extensión .tar.gz o .zip

Si la práctica se realiza en equipo, agregar el nombre y número de cuenta de los integrantes en el README.txt

Enviar la práctica al correo [ciclomax9@ciencias.unam.mx](mailto:ciclomax9@ciencias.unam.mx) con el asunto [LC-Apellido-Nombre-P1p2].

*Suerte*

*"Es genial trabajar con computadoras.*

*No discuten, lo recuerdan todo y sobre todo no se beben tu cerveza"*

-- Paul Leary