



Universidad del Istmo de Guatemala  
Facultad de Ingenieria  
Ing. en Sistemas  
Informatica 1  
Prof. Ernesto Rodriguez - erodriguez@unis.edu.gt

---

## Hoja de trabajo #7

Fecha de entrega: 10 de Septiembre, 2019 - 11:59pm

---

*Instrucciones: Resolver cada uno de los ejercicios siguiendo sus respectivas instrucciones. El trabajo debe ser entregado a traves de Github, en su repositorio del curso, colocado en una carpeta llamada "Laboratorio 7". Al menos que la pregunta indique diferente, todas las respuestas a preguntas escritas deben presentarse en un documento formato pdf, el cual haya sido generado mediante Latex.*

**Nota:** Para esta tarea, debe tener instalado "Elm" en su computadora. Puede obtener el lenguaje "Elm" en: <https://guide.elm-lang.org/install.html>

### Ejercicio #1 (20%)

Un *arbol binario* es una estructura de datos similar a una lista excepto que cada sección del arbol tiene dos arboles más en vez de tener solamente uno.

Esto significa que se necesitan dos constructores algebraicos:

- Un constructor que representa un arbol vacio
- Un constructor que acepta un numero y dos arboles más

Su tarea es definir un tipo que represente un arbol.

### Ejercicio #2 (20%)

Defina una función llamada `masUno` que toma un arbol definido mediante el tipo de la sección anterior y le suma uno (1) a cada valor del arbol.

### Ejercicio #3 (20%)

Defina una función llamada `map` :  $(\mathbf{Int} \rightarrow \mathbf{Int}) \rightarrow \mathbf{Arbol} \rightarrow \mathbf{Arbol}$ . Esta función toma un arbol y aplica la función pasada como primer parametro a cada elemento del arbol.

### Ejercicio #4 (20%)

Defina una función llamada `sum` :  $\mathbf{Arbol} \rightarrow \mathbf{Int}$  la cual suma todos los numeros que aparecen en el arbol dado como primer parametro.

## Ejercicio #5 (20%)

Defina una función llamada **foldTree** :  $(\mathbf{Int} \rightarrow \mathbf{Int} \rightarrow \mathbf{Int} \rightarrow \mathbf{Int}) \rightarrow \mathbf{Int} \rightarrow \mathbf{Arbol} \rightarrow \mathbf{Int}$ . Esta función recorre todos los elementos del arbol. Si el elemento se encuentra vacío, simplemente retorna el valor dado como segundo parametro, de lo contrario llama la función **foldTree** recursivamente en cada uno de los arboles y luego le aplica a los dos resultados y al valor del elemento del arbol.