

Auxiliar #5

Profesor: Patricio Inostroza

Jaime Salas
rittoshadow@gmail.com

Sebastián Blasco
sblasco@dcc.uchile.cl

P1. Siempre es difícil calcular las horas de término de los controles en la facultad, ello pues aunque los tiempos para cada sección son iguales, nunca comienzan todas las secciones exactamente al mismo tiempo. Se desea diseñar el módulo `estructuraTiempo` que incluya una estructura y funciones para manipular representaciones horarias.

1. Implemente una estructura `tiempo` compuesta por horas, minutos y segundos.
2. Diseñe e implemente funciones para trabajar con su nueva estructura:
 - función `validarHora` que dada una estructura `tiempo` verifica si esta es una representación horaria válida o no.
 - función `normalizarTiempo` que dada una estructura `tiempo` retorna otra estructura con los valores internos normalizados para representar una hora.
 - función `sumarTiempos` que dadas dos estructuras `tiempo` retorna una nueva estructura `tiempo` con la suma de las dos.
 - función `totalDias` que dada una estructura `tiempo` retorna el número entero de días con que se corresponde.

P2. Suponga que los nombres (sólo nombres) de los integrantes del curso CC1002 se han ingresado en una lista. El profesor desea poder realizar ciertas operaciones con dicha lista, para lo cual usted debe diseñar las siguientes funciones:

1. `buscar` que recibe una lista y un nombre, y retorna si está presente o no dicho nombre en la lista.
2. `totalAlumnos` que recibe una lista y retorna la cantidad de elementos de la lista.
3. `contar` que recibe una lista y un nombre, y retorna la cantidad de apariciones de dicho nombre en la lista.
4. `borrar` que recibe una lista y un nombre, y retorna una lista a la que se le han removido todas las apariciones del nombre especificado.

P3. Dadas las siguientes listas:

```
>>> lista1 = crearLista("Claudia" , crearLista("Maximiliano" ,
        crearLista("Javier" , crearLista("Lisa" , listaVacia))))
>>> lista2 = crearLista(True , crearLista(False , crearLista(False,
        crearLista(True , crearLista(True, listaVacia))))))
>>> lista3 = crearLista(4 , crearLista(3.14 , crearLista(1.41421,
        crearLista(9 , crearLista(-1, listaVacia))))))
>>> lista4 = crearLista("fcfm" , crearLista("UCH" , lista1))
>>> lista5 = crearLista(lista2, lista3)
```

¿Qué retornan las siguientes expresiones?

1.

```
>>> cabeza( cola( cola( lista1)))
```
2.

```
>>> cola( cola( cola( cola( lista2))))
```
3.

```
>>> cabeza( cola( cola( cola( lista3 ))))
```
4.

```
>>> cola( cola( cola( cola( lista4 ))))
```
5.

```
>>> cabeza( cola( cola( cabeza( lista5 ))))
```

P4. En el Departamento de Ingeniería Civil de la facultad se desea diseñar un sistema para medir el largo de un camino, del cual solo se tienen algunas mediciones de puntos espaciales. Para ello:

1. Usando el módulo *estructura* implemente **punto3D** que almacene las 3 coordenadas x , y y z .
2. Diseñe e implemente una función **distancia** que calcule la distancia entre dos estructuras **punto3D**
3. Diseñe e implemente una función **distanciaTotal** que reciba una lista de **punto3D** y retorne la suma de las distancias entre cada punto contiguo en la lista.
4. Construya un programa interactivo que solicite las coordenadas de 5 puntos y calcule la distancia del camino representado.