

## ESTRUCTURAS DE DATOS Y ALGORITMOS 2

### SEGUNDO SEMESTRE DE 2015

#### SEGUNDO OBLIGATORIO

#### ENTREGA 1

##### Ejercicio 1: Enlaces críticos

Tomás es encargado de mantener la red de computadoras de un laboratorio de la facultad, en donde todas las computadoras están interconectadas de a pares, o sea que existe un cable entre dos computadoras y así consecutivamente formando una gran red local entre las mismas. Para facilitarle el trabajo a Tomás, la facultad le solicita a usted desarrollar un algoritmo que indique los cables críticos. Los cables críticos son aquellos que al ser desconectados ocasionan una pérdida de la conectividad existente en todo el laboratorio.

Las computadoras están identificadas por un número natural y los enlaces son indicados por los números de las computadoras que conecta el mismo.

Datos que recibe la función	Iterador<nat> computadoras Iterador<Tupla<nat, nat>> enlaces
Retorno	Tupla<TipoRetorno, Iterador<Tupla<nat, nat>>>

Si no hay ningún enlace que rompa la conectividad de la red el retorno deberá ser OK y el iterador será vacío.

##### Ejercicio 2: Ahorro de enlaces

La facultad le pidió a Tomás que no utilice demasiado cable para interconectar todas las computadoras del laboratorio, el problema es que hay tantas conexiones posibles que Tomás tiene miedo de confundirse y dejar sin conexión alguna computadora, por lo que le pide ayuda a usted para determinar la mínima cantidad posible de metros de cable y seguir manteniendo la conectividad.

Como datos de entrada, Tomás le proporciona una matriz que contiene las longitudes de los cables entre las computadoras. El largo de los mismos son enteros positivos mayores a cero, por lo que se deberá tomar en cuenta que si un valor de la matriz es cero, significa que no existe un cable entre ese par de computadoras.

Datos que recibe la función	Matriz<nat> &laboratorio
Retrno	Tupla<TipoRetorno, Iterador<Tupla<nat,nat>>>

El retorno esperado incluye un iterador de los cables que forman la interconexión mínima. Para representar un cable se utilizará la Tupla<nat,nat>, la cual representa <origen, destino>.

### Ejercicio 3: Gerente de proyectos

Un gerente de proyectos está liderando actualmente un proyecto demasiado grande para lo cual le solicita ayuda a usted para organizar las tareas a repartir. Para que el proyecto resulte exitoso, se deberán cumplir con una lista de tareas, pero existen algunas que dependen de la finalización de otras para poderse realizar, por lo cuál deberá determinar en que orden se tendrán que realizar las mismas para cumplir con todas las precedencias e incluso si es posible ejecutar el proyecto. Las tareas están identificadas por una descripción y las relaciones de dependencia entre unas y otras por una tupla con las descripciones de cada una.

Datos que recibe la función	Iterador<Cadena> tareas Iterador<Tupla<Cadena, Cadena>> precedencias
Retorno	Tupla<TipoRetorno, Iterador<Cadena>>

La operación deberá retornar ERROR si no existe un orden para las tareas.