

Estructura de Datos y Algoritmos
Grado Ingeniería Informática
Doble Grado Ingeniería Informática
y Administración de Empresas
Universidad Carlos III de Madrid
CURSO 2019-2020



Convocatoria Extraordinaria. 26 Junio 2020

Instrucciones:

- Duración: **30 minutos** (para desarrollar y subir tu solución).
- Lee atentamente el problema. Vuelve a leer el problema una segunda vez.
- Si tienes alguna duda pregunta en el chat. **NO PREGUNTES COSAS QUE ESTÁN EN EL ENUNCIADO. Sólo se responderán preguntas los 10 primeros minutos.**
- Descarga el fichero **problema3.py** y completa tu solución. Pon atención en qué carpeta vas a descargar y modificar el fichero. **NO RENOMBRES EL FICHERO.**
- Este fichero contiene una clase Test (unittest) que te permitirá comprobar si tu solución es correcta o no. **NO BORRES NI MODIFIQUES Test.**
- Cuando queden 5 minutos el profesor, te avisará para que subas tu solución (fichero **problema3.py**) a la actividad **Entrega Problema 3**. Las soluciones enviadas fuera de tiempo no serán evaluadas.
- Es tu responsabilidad comprobar que has subido la versión correcta del fichero.

Problema 3 (20 puntos):

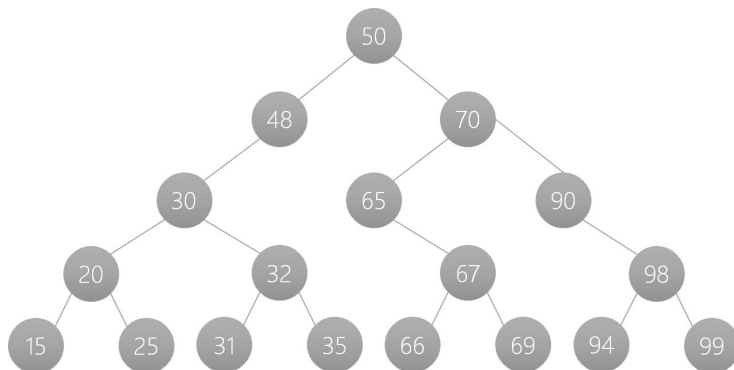
Sea MyBST la clase que implementa un árbol binario de búsqueda en Python (es una versión reducida que sólo incluye los métodos necesarios para este problema).

En un árbol binario, el **mínimo ascendiente común** (*lowest common ancestor*) de dos nodos n y m es el nodo que contiene a los nodos n , m y a todos sus descendientes.

Implementa una función, ***lwc***, que reciba dos enteros a y b , encuentre sus nodos y obtenga el mínimo ascendiente común de ambos nodos. La función debe devolver el elemento asociado de dicho nodo ascendente. Si a ó b , no existen, la función debe devolver None. Si el árbol está vacío, la función debe devolver None.

Pista: La solución más fácil sigue un enfoque recursivo. Puedes utilizar una función auxiliar.

Dado el siguiente árbol, veamos algunos ejemplos:



```
- tree.lwc(48,70)=50
- tree.lwc(30,70)=50
- tree.lwc(30,65)=50
- tree.lwc(20,32)=30
- tree.lwc(67,90)=70
- tree.lwc(69,99)=70
- tree.lwc(31,94)=50
- tree.lwc(15,25)=20
- tree.lwc(15,35)=30
- tree.lwc(15,32)=30
- tree.lwc(20,30)=30
- tree.lwc(67,70)=70
- tree.lwc(30,35)=30
```