



IIC2115 – Programación como Herramienta para la Ingeniería (II/2022)

Ejercicio Capítulo 1b

Aspectos generales

- **Objetivos:** Aplicar los contenidos fundamentales de programación orientada a objetos para modelar un invernadero de frutos rojos.
- **Lugar de entrega:** lunes 22 de Agosto 2022 a las 22:00 hrs. en repositorio privado.
- **Formato de entrega:** archivo Python Notebook (**C1b.ipynb**) con el avance logrado durante la sesión. El archivo debe estar ubicado en la carpeta **C1b**. Utilice múltiples celdas de texto y código para facilitar el trabajo del cuerpo docente.

Descripción del problema

En esta ocasión se presentan dos desafíos que se deben resolver con estructuras de datos vistas en clases

Stacks: Balancear Ecuaciones

Se tiene una expresión aritmética como la siguiente:

$$2 + (4 - 5) - ((8/1 - 5) + 7) * 6$$

Se le pide que, usando la estructura de Datos de *Stack*, logre:

- Revisar si tiene balanceados sus paréntesis.
- Imprimir si está balanceado y en caso de no estarlo, en qué parte de la ecuación falta un paréntesis.
- Corregir los errores de la ecuación e imprimir esta nueva versión.

Tenga en cuenta que:

- La ecuación llega como un *string*.
- Se pueden usar diferentes símbolos para los paréntesis.
- Puede haber espacios entre los símbolos.

Hint: revise los Notebooks de ejemplo "01 - Estructuras de datos básicas.ipynb" y "03 - Stacks.ipynb" para métodos útiles de listas y la implementación de stacks respectivamente.

Árbol Filogenético

Se le ha entregado un árbol filogenético; estos tipos de árboles son diagrama que representa las relaciones evolutivas entre organismos, como la que se ve en la figura 1

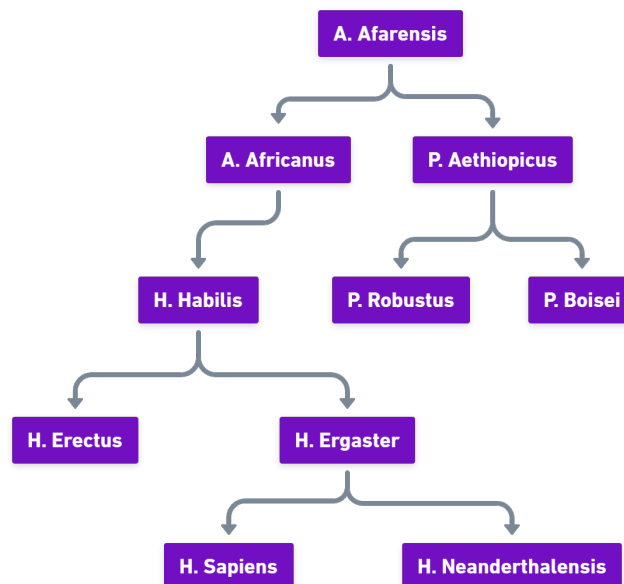


Figure 1: Árbol Filogenético Homínidos

Se le ha pedido encontrar las ramas evolutivas más largas. Para eso debe:

- Crear una clase Nodo que pueda guardar el nombre del organismo (ej: H. Sapiens) y los posibles hijos del nodo. Puede crear todos los métodos que estime necesarios.
- Crear una clase Árbol que represente la estructura de datos. Puede generar los atributos y métodos que quiera.

- Generar un árbol de ejemplo de mínimo 3 niveles y múltiples hijos.
- Armar un algoritmo que recorra el árbol y encuentre la o las ramas más largas, para luego imprimirlas.

Hint: revise el Notebook de ejemplo "06 - Árboles.ipynb" para la implementación de las clases y los algoritmos BFS y DFS.