

TRABAJO PRÁCTICO N ° 6

Árboles generales

Departamento de Ciencias e Ingeniería de la Computación - U.N.S.

Ejercicio 1:

- Programa el *TDA Árbol* respetando la interfaz *Tree<E>* provista por la cátedra. El árbol general deberá contar con una estructura de colección de hijos y referencia al padre, donde los rótulos de los nodos son de tipo genérico *E*. Para implementar la colección de hijos de un nodo utilice la clase lista programada en el TP N° 4.
- Corra el correspondiente test JUnit para verificar la correctitud de su implementación.
- Calcule el orden del tiempo de ejecución de cada una de las operaciones de la clase árbol. Justifique adecuadamente.

Ejercicio 2:

- Agregue un método a la clase *arbol* definida anteriormente cuya signature sea:

```
public void eliminarUltimoHijo(Position<E> p) throws InvalidPositionException, InvalidOperationException.
```

Este método deberá eliminar del árbol receptor del mensaje a la posición *p* siempre que *p* sea el último hijo (de izq a der) de su padre. La raíz no se considera último hijo, en este caso el método deberá lanzar *InvalidOperationException*.
- Calcule el tiempo de ejecución de su solución e indique su orden.

Ejercicio 3: Programe un método con la siguiente signature: `public Map<Character, Integer> cantidadRepeticiones(Tree<Character> t)`. Este método deberá retornar un mapeo con cada uno de los caracteres del árbol y la cantidad de veces que aparece cada carácter en el árbol. Resuelva este problema utilizando un recorrido en preorden.

Ejercicio 4: Dado un árbol *a* de Strings y un String *s*, programe un método tal que retorne un Iterable con las posiciones del árbol en las que aparece el String *s*. Para resolver este problema implemente un recorrido en postorden.

Ejercicio 5:

- Escriba un método tal que dado un árbol genérico *a* y un elemento *e*, elimine de *a* todas las apariciones de *e*. Compare los elementos por equivalencia. El método debe retornar la cantidad de eliminaciones realizadas.
- Calcule el tiempo de ejecución de su solución e indique su orden.

Ejercicio 6:

- Dado un árbol de enteros *a* y un entero *n*, escriba un método que determine si *n* pertenece al árbol *a*. Para resolver este método utilice el iterador del árbol.
- Calcule el tiempo de ejecución de su solución e indique su orden.