

## UNIDAD TEMÁTICA 4: Árboles Binarios I y Recursividad

### TRABAJO DE APLICACIÓN 5

#### Ejercicio #1

Un problema (o función matemática) definido de forma recursiva tiene una implementación recursiva casi directa. Sin embargo, diversos autores recomiendan que si es posible implementar una solución iterativa de complejidad equivalente, es conveniente usar esta última.

Por ejemplo, para implementar la función factorial se puede programar el siguiente algoritmo recursivo:

```
// función factorial recursiva
public static int factorial(int n) {
    SI (n == 0) devolver 1; // caso base
    SINO, devolver
        n * factorial(n- 1); // caso recursivo
}
```

A continuación se presenta un pseudocódigo que implementa la función factorial en forma iterativa, en el cual se han mezclado las sentencias en cualquier orden (**DEBEN SER ORDENADAS EN FORMA APROPIADA**).

```
factorialIterativo (n de tipo entero)
Comienzo
fin mientras
fact ← fact * n
mientras n > 1 hacer
devolver fact
fact ← n
n ← n - 1
fin
```

**En sub-equipos:**

**1. Sub-Equipo A:**

- a) Analiza el orden del tiempo de ejecución del algoritmo recursivo
- b) Implementa y prueba **con test cases** el algoritmo recursivo “**factorial\_recursivo**”

**2. Sub-Equipo B:**

- a) Analiza el orden del tiempo de ejecución del algoritmo iterativo
- b) **Genera el paquete UT4-TA5 en el repositorio**
- c) Implementa y prueba **con test cases** el algoritmo recursivo “**factorial\_iterativo**”

**Responder preguntas presentadas en pantalla**