

Calcular el $T(n)$ y el big-Oh de :

```
public static String concatNro(int[] args, int n) {  
    String result = "";  
    int i = n;  
    while (i > 1) {  
        result += ", " +  
        i /= 2;  
    }  
    return result + concatNro(args, n/2);  
}
```

JUSTIFICACIÓN

Al analizar concatNro, me encontré con un problema que es clave. El algoritmo no tiene un caso base, es decir, que su complejidad temporal tiene a infinito ($T(n) = \infty$).

Lo que quiero decir, es que, concatNro se va a ejecutar un número ilimitado de veces. Al confirmar esto, podemos decir que no se puede aplicar la notación Big-Oh, porque no existe una función que logre anotar un crecimiento que tiende al infinito.

En conclusión, el algoritmo tiene propósito pero ahora mismo por cómo está construido no sirve para el propósito que fue encomendado.

PROPUESTA DE SOLUCIÓN

```
public static String concatNro(int[] args, int n)  
{  
    if (n <= 1)  
    {  
        return "";  
    }  
    String result = "";  
    int i = n;  
    while (i > 1)  
    {  
        result += ", " + i; i /= 2;  
    }  
    return result + concatNro(args, n/2);  
}
```