

## Práctica 02

DOCENTE	CARRERA	CURSO
Vicente Machaca Arceda	Maestría en Ciencia de la Computación	Algoritmos y Estructura de Datos

PRÁCTICA	TEMA	DURACIÓN
02	–	–

### 1. Datos de integrantes

- Grupo: 2
- Integrantes:
  - EDER ALONSO AMPUERO ATAMARI
  - HOWARD FERNANDO ARANZAMENDI MORALES
  - JOSE EDISON PEREZ MAMANI
  - HENRRY IVAN ARIAS MAMANI

### 2. Url GITHUB

Repositorio Github: Práctica 2

### 3. Estructuras de Datos

#### 3.1. BTree

ZZZZZZZZZZ

#### 3.2. AVL

El Merge Sort es un algoritmo recursivo bastante eficiente para ordenar un array, que tiene un orden de complejidad  $O(n \log n)$  al igual que Quick Sort. fue desarrollado en 1945 por John Von Neumann.

El Merge Sort está basado en la técnica de diseño de algoritmos Divide y Vencerás, esta técnica consiste en dividir el problema a resolver en sub problemas del mismo tipo que a su vez se dividirán, mientras no sean suficientemente pequeños o triviales.

##### 3.2.1. Resultados del experimento

### 4. Conclusiones

- En las pruebas realizadas para el Algoritmo Quick Sort, se obtuvo tiempos de ejecución menores para el código desarrollado en lenguaje de programación Golang, y tiempos de mayor valor en la ejecución del código en lenguaje de programación Python.

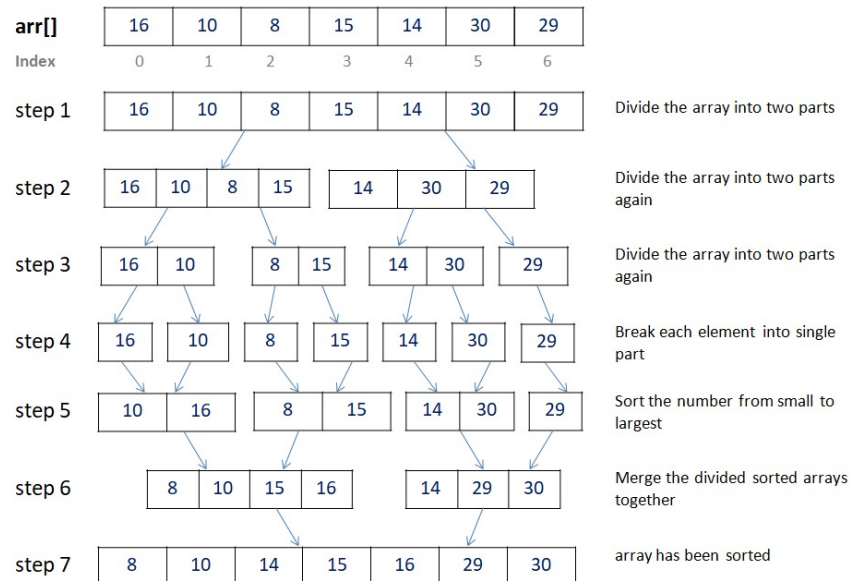


Figura 1: Estrategia que sigue algoritmo para ordenar una secuencia S de n elementos

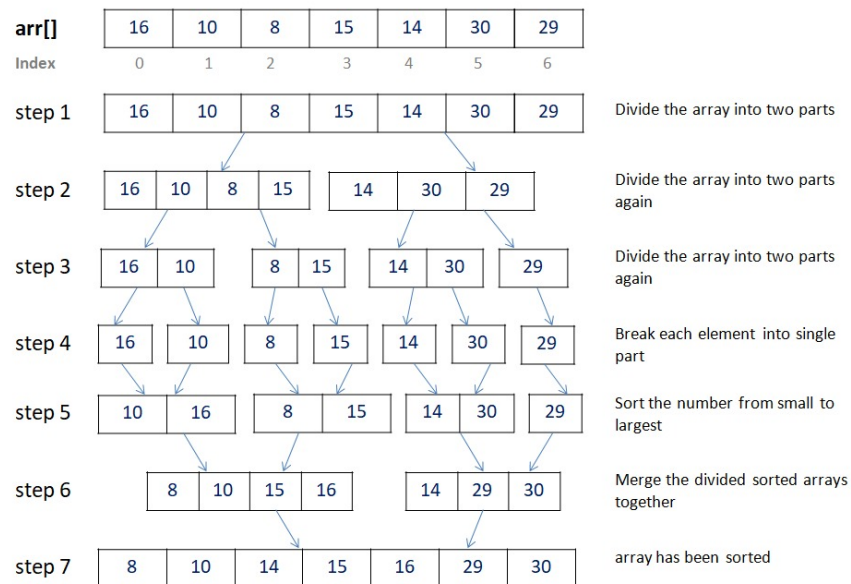


Figura 2: Estrategia que sigue algoritmo para ordenar una secuencia S de n elementos

- Se observó que, para tamaños de entrada menores a 10 000 datos, los tiempos de ejecución son inconsistentes, al ejecutar el código del algoritmo Quick Sort en lenguaje Golang, presentándose en reiteradas oportunidades valores de cero.

- En la ejecución del algoritmo Quick Sort, los valores de desviación estándar son mayores al ejecutar el código en lenguaje Python, y presentan valores menores al usar el código en C++. Para las pruebas realizadas en Python se observa que los valores de desviación estándar van en aumento respecto al tamaño de la entrada, en el caso del lenguaje Golang y C++, estos valores se incrementan desde 100 000 y 20 000 datos, respectivamente.
- Los programas en lenguaje compilado (C++) presenta un mejor desempeño en cuanto se presenta mayor cantidad de datos que un programa en lenguaje interpretado (Python)

## 5. Referencias