Universidad del Valle de Guatemala Christopher Aju, 13171

Algoritmos y estructuras de Datos Pedro Joaquin

Hoja de Trabajo 6

HashSet

|  |  |
| --- | --- |
|  | Tiempo (µs) |
| Si subconjunto | 288.666 |
| No subconjunto | 379.755 |

TreeSet

|  |  |
| --- | --- |
|  | Tiempo (µs) |
| Si subconjunto | 472.983 |
| No subconjunto | 372.058 |

LinkedHashSet

|  |  |
| --- | --- |
|  | Tiempo (µs) |
| Si subconjunto | 382.321 |
| No subconjunto | 354.525 |

Resultados:

Para los casos donde los desarrolladores si eran subconjuntos la implementación más rápida es el tipo HashSet.

Para los casos donde los desarrolladores no eran subconjuntos la implementación más rápida es el tipo LinkedHashSet.

Complejidad de tiempo para implementación HashSet:

|  |  |
| --- | --- |
| Tipo de Instrucción | Tipo de Complejidad |
| If – else | O(1) |
| For -> n limitante | O(n) |
| If – else | O(1) |
| Set.contains -> peor caso | O(log(n)) |
| If – else | O(1) |

Complejidad del algoritmo: O(n log(n))

Link de repositorio GitHub: <https://github.com/PedroJoaquin/SET.git>