Trabajo Integrador: Algoritmos de Búsqueda y Ordenamiento en Python

Datos Generales

• Título del trabajo: Algoritmos de Búsqueda y Ordenamiento en Python

• Alumnos:

Integrante 1: Santagada, Leandro Nahuel – DNI 38258056

Integrante 2: Sedron, Claudio Andrés – DNI 24929288

• Materia: Programación I

• Profesor/a: Nicolás Quirós

• Fecha de Entrega: 20/06/2025

Índice

1. Introducción

Este trabajo explora dos técnicas fundamentales en programación: los algoritmos de búsqueda (lineal y binaria) y métodos de ordenamiento (burbuja y quick sort). Se seleccionó este tema por su importancia crítica en optimizar el acceso y procesamiento de información. El objetivo es analizar e implementar estos algoritmos en Python, destacando su relevancia en aplicaciones prácticas.

2. Marco Teórico

La búsqueda lineal compara secuencialmente cada elemento de una lista hasta hallar una coincidencia, ideal para listas pequeñas o desordenadas. La búsqueda binaria, mucho más eficiente para listas grandes, requiere previamente ordenar los datos. El método burbuja ordena comparando pares adyacentes, sencillo pero lento. Quick sort usa recursividad y partición con pivote, siendo altamente eficiente en la mayoría de los casos.

3. Caso Práctico

El caso práctico implica el desarrollo de un programa interactivo en Python que permite elegir algoritmos para ordenar una lista y realizar búsquedas sobre ella. Se utilizan dos listas: una pequeña para ejemplificar claramente cada algoritmo y otra de 1000 elementos aleatorios para comparar rendimientos. El código se acompaña de comentarios explicativos y medición de tiempos con el módulo time.

4. Metodología Utilizada

Se realizó una investigación previa, seguida por el desarrollo colaborativo del código mediante Github, utilizando Python y VSCode. El código se probó en diversos escenarios, validando su rendimiento y corrigiendo errores. La medición de tiempos permitió evaluar la eficiencia real de cada algoritmo.

5. Resultados Obtenidos

Los resultados mostraron la superioridad de quick sort y búsqueda binaria en términos de eficiencia, especialmente en listas grandes. El método burbuja y la búsqueda lineal demostraron utilidad principalmente en contextos educativos o pequeñas escalas, resaltando la importancia de elegir algoritmos apropiados según el tamaño y tipo de datos.

6. Conclusiones

Este proyecto facilitó un entendimiento profundo de algoritmos esenciales, su implementación y la importancia de seleccionar adecuadamente según la tarea requerida. Se aprendieron conceptos clave como eficiencia algorítmica y buenas prácticas de programación, además de destacar la colaboración efectiva en grupo.

7. Bibliografía

Python – Nivel 31 – Reto 3 – Ordenamiento de burbuja [Video]. YouTube. Publicado hace 5.2 años.

https://www.youtube.com/watch?v=4397M63NEEc

Algoritmo de Quicksort en Python [Video]. YouTube. Publicado hace 4.8 años.

https://www.youtube.com/watch?v=O7gwvLy695c

Amorín, D. (2023). Ordenamiento de Burbuja en Python. DiegoAmorin.com.

https://diegoamorin.com/ordenamiento-burbuja-python/

Tutorias.co. (s.f.). arrays – python (búsqueda binaria).

https://tutorias.co/arrays-python-busqueda-binaria/

Tutorias.co. (s.f.). arrays – python (búsqueda secuencial de un dato).

https://tutorias.co/arrays-python-busqueda-secuencial/

Anexos

- Código completo (main.py y lista.py).

- Video explicativo subido a YouTube.

https://youtu.be/XmIyV2C3k_g

- Grafico comparativo entre algoritmos.