

# Proyecto del Curso: Análisis de Algoritmos

Samir Cabrera Tabash  
Mynell Myers

**Instituto Tecnológico de Costa Rica**  
Curso: Análisis de Algoritmos  
Profesor: Eddy Ramírez Jiménez

# 1 Resumen Ejecutivo

Este proyecto tiene como objetivo abordar el Problema del Viajante de Comercio (TSP) mediante la implementación y comparación de tres enfoques algorítmicos: backtracking, programación dinámica y algoritmos genéticos. La tarea se centra en encontrar el recorrido más corto que visite cada ciudad exactamente una vez y regrese al punto de inicio.

faltan los resultados y las pruebas aca xd  
faltan los resultados y las pruebas aca xd  
faltan los resultados y las pruebas aca xd  
faltan los resultados y las pruebas aca xd  
faltan los resultados y las pruebas aca xd

# 2 Introducción

El presente proyecto aborda el fascinante y desafiante Problema del Viajante de Comercio (TSP), un clásico en la teoría de la computación y la optimización combinatoria. Este problema, formulado inicialmente en la década de 1800 y refinado a lo largo del tiempo, busca encontrar la ruta más corta que visite cada ciudad exactamente una vez y retorne al punto de inicio. Aunque su formulación matemática se estableció en la década de 1930, el TSP sigue siendo objeto de estudio en diversas disciplinas, incluyendo la informática, las matemáticas, la inteligencia artificial y la logística.

El TSP ha demostrado ser un desafío computacional significativo, y su complejidad se acentuó cuando en 1972, Richard Karp demostró que es un problema NP-completo. Esta clasificación implica que no existe un algoritmo eficiente conocido para encontrar la solución óptima en todos los casos, a menos que se resuelva la famosa conjetura  $P = NP$ . Ante la dificultad de resolver el TSP de manera exacta para instancias grandes, han surgido numerosas heurísticas y algoritmos de aproximación que exploran enfoques como algoritmos genéticos, simulated annealing y algoritmos hormiga.

El objetivo de este proyecto es explorar y comparar tres enfoques específicos para abordar el TSP: backtracking, programación dinámica y algoritmos genéticos. Cada uno de estos métodos presenta sus propias ventajas y limitaciones, y el análisis comparativo proporcionará una visión detallada de su rendimiento en términos de eficiencia y calidad de las soluciones.

A lo largo de este documento, se detallarán las implementaciones de cada enfoque, destacando las estrategias algorítmicas utilizadas y los desafíos encontrados durante el desarrollo.

Este proyecto no solo busca proporcionar soluciones algorítmicas efectivas para el TSP, sino también fomentar la comprensión profunda de las técnicas empleadas y su aplicación en la resolución de problemas prácticos. La resolución del TSP no solo tiene implicaciones teóricas, sino también aplicaciones prácticas significativas en la planificación de rutas, la logística y la optimización de recursos.

A través de este análisis detallado, esperamos no solo encontrar soluciones eficientes para el TSP, sino también contribuir al entendimiento y desarrollo continuo en el campo del análisis de algoritmos.

### **3 Marco Teórico**

### **4 Descripción de la Solución**

### **5 Resultados de Pruebas**

### **6 Conclusiones**

### **7 Aprendizajes**

La realización de este proyecto nos brindó una valiosa experiencia en la implementación y análisis de algoritmos para abordar el Problema del Viajante de Comercio (TSP). Utilizando Java como lenguaje de programación, hemos extraído aprendizajes significativos tanto a nivel individual como colectivo.

#### **7.1 Dominio de Java**

Programar en Java nos permitió fortalecer nuestras habilidades en este lenguaje orientado a objetos. Desde la gestión de clases y objetos hasta la manipulación de estructuras de datos, el proyecto consolidó nuestro dominio de Java y su aplicación en la resolución de problemas algorítmicos complejos.

#### **7.2 Implementación de Algoritmos**

La implementación de distintos enfoques para el TSP, como backtracking, programación dinámica y algoritmos genéticos, nos proporcionó una comprensión profunda de las estrategias algorítmicas específicas para cada caso. La experiencia práctica en la traducción de conceptos teóricos a código funcional enriqueció nuestra perspectiva sobre la eficiencia y la complejidad algorítmica.

#### **7.3 Paralelización y Eficiencia**

La integración de la paralelización en Java para ejecutar los algoritmos de manera concurrente nos introdujo en el mundo de la optimización de procesos. Aprendimos a gestionar la concurrencia de manera efectiva, mejorando la eficiencia de nuestros programas y comprendiendo la importancia de la adaptabilidad en entornos de ejecución paralela.

## **7.4 Benchmarking y Evaluación de Resultados**

La realización de pruebas exhaustivas y la evaluación comparativa de los resultados nos enseñaron la importancia del benchmarking en la validación de algoritmos. Aprendimos a interpretar y analizar datos de rendimiento, lo cual es esencial para la toma de decisiones informadas en el diseño y selección de algoritmos.

## **7.5 Colaboración en Equipo**

Trabajar en parejas nos brindó la oportunidad de compartir conocimientos, abordar desafíos de manera colaborativa y aprovechar la diversidad de habilidades individuales. La comunicación efectiva fue clave para el éxito del proyecto, destacando la importancia de la colaboración en equipo.

## **7.6 Aplicaciones Prácticas y Relevancia Teórica**

Comprender la aplicabilidad del TSP en contextos prácticos, como la planificación de rutas y la logística, nos conectó con las aplicaciones del análisis de algoritmos en situaciones del mundo real. Al mismo tiempo, la profundización en la teoría del TSP contribuyó a nuestro entendimiento de problemas computacionales complejos.

En resumen, la realización exitosa de este proyecto en Java no solo amplió nuestras habilidades técnicas, sino que también enriqueció nuestra comprensión de los principios fundamentales de la ciencia de la computación y la teoría de algoritmos. Estos aprendizajes se traducen en una base sólida para abordar futuros desafíos en el ámbito de la programación y el análisis de algoritmos.

# **8 Bibliografía**