



Instituto Politécnico Nacional.

Unidad Profesional Interdisciplinaria de Ingeniería  
campus Zacatecas.

Ingeniería en Sistemas Computacionales

Análisis de algoritmos.

Profesor: Roberto Oswaldo Cruz Leija.

Grupo: 3CM1

“TSP (Travelling Salesman Problem): El  
Problema del Agente Viajero”

ALUMNA;

Alejandra Monserrath Esparza Ríos.

Zacatecas, Zac., a 07 de noviembre del 2019.

## **INTRODUCCION.**

El problema del agente viajero es uno de los problemas más famosos de la matemática computacional. Este problema en concreto ha sido muy estudiado por múltiples aplicaciones en la optimización de recursos, tanto en el campo empresarial como en la robótica (desplazamientos para un circuito impreso).

Este problema planteado como un viajante podría empezar y terminar en una ciudad concreta, pasando por todas las ciudades que están en su mapa una sola vez y por la mínima ruta posible. A priori la solución puede verse sencilla, solo se tendría que probar cuál de las posibles combinaciones de rutas sería la más óptima (fuerza bruta). Lo difícil sería cuando el número de ciudades es elevado ya que las posibles combinaciones aumentan de manera exponencial.

## **CARACTERISTICAS DEL PROBLEMA**

Dada una lista de ciudades y las distancias entre cada par de ellas. ¿Cuál es la ruta más corta posible que visita cada ciudad exactamente una vez y regresa a la ciudad origen? Esto conlleva: distancias, el inicio y a la vez siendo el mismo final, número de ciudades.

## **METODOLOGIA DE LA SOLUCION.**

Consiste en un agente de ventas que tiene “n” ciudades por visitar, comenzando y terminando en la misma ciudad, visitando solamente una vez cada ciudad, y haciendo que el recorrido sea el costo mínimo, este costo puede estar expresado en términos de tiempo o distancia, es decir recorre el mínimo de kilómetros o lleva a cabo un recorrido en el menor tiempo posible. El problema consiste en encontrar una ruta que, empezando y terminando en la misma ciudad, recorra sólo una vez las ciudades restantes y que a la vez esta ruta sea la mínima posible. Los costos son simétricos en el sentido de que viajar desde la ciudad X a la ciudad Y tiene el mismo costo que viajar desde la ciudad Y a la ciudad X. La condición de visitar todas las ciudades implica que el problema se reduce a decidir en qué orden las ciudades van a ser visitadas

## RESULTADOS.

```
TSP p = new TSP("zxywtr");
p.introducirRuta('z','x', 5);
p.introducirRuta('z','y', 78);
p.introducirRuta('z','w',15);
p.introducirRuta('z','t', 89);
p.introducirRuta('z','r', 3);
p.introducirRuta('x','y', 6);
p.introducirRuta('x','w', 67);
p.introducirRuta('x','t', 58);
p.introducirRuta('x','r', 30);
p.introducirRuta('y','w', 2);
p.introducirRuta('y','t', 9);
p.introducirRuta('y','r', 8);
p.introducirRuta('w','t', 56);
p.introducirRuta('w','r', 97);
p.introducirRuta('t','r', 1);

String r = p.buscarRuta('x', 'y');
```

```
6: x y
BUILD SUCCESSFUL (total time: 1 second)
TSP g = new TSP("abcdef");
g.introducirRuta('a','b', 5);
g.introducirRuta('a','e', 78);
g.introducirRuta('a','f',15);
g.introducirRuta('b','c', 89);
g.introducirRuta('b','e', 3);
g.introducirRuta('c','d', 6);
g.introducirRuta('c','e', 67);
g.introducirRuta('c','f', 58);
g.introducirRuta('d','f', 30);
g.introducirRuta('e','f', 22);

String respuesta = g.buscarRuta('a', 'e');

run:
8: a b e
BUILD SUCCESSFUL (total time: 0 seconds)
```

## **CONCLUSIONES.**

Se podría decir que este tipo de problema se podría utilizar para acortar las distancias de un punto a otro y optimizar el tiempo.

Así por ejemplo si queremos llegar de un punto a otro pero queremos saber cuál es la ruta que mejor nos convendría para llegar más rápido y ahorrar tiempo, este sería una buena forma de solucionarlo.