



Unidad Profesional Interdisciplinaria de Ingeniería campus Zacatecas

Problema TSP

Carrera: Sistemas Computacionales.

Grupo: 2CM2.

Alumnas: Mónica Yoselin Gallardo Galván

Profesor: Roberto Oswaldo Cruz Leija.

Materia: Análisis de Algoritmos

Introducción

Este documento muestra la propuesta de solución para el problema de TSP donde se describe el algoritmo. La idea de solución tengo que dar los créditos al compañero Osvaldo de Luna Herrera ya que el fue el principal aportador del método de solución, yo intente con varias simulaciones y ninguna me daba así que con su ayuda llegamos a esta, por mi parte aporte conocimientos en el lenguaje para plasmar la solución al código y a la vez, simplificaciones aplicadas.

Planteamiento del problema.

Un hombre de negocios debe recorrer n ciudades las cuales están conectadas por un camino con distancia no necesariamente igual. El problema redunda en no visitar una ciudad dos veces, pero si recorrer todas de la manera más optima (esto es, con la menor distancia recorrida) hasta regresar a la ciudad donde comenzó su viaje. La única información con la que se cuenta es, la ciudad donde comienza y la distancia que hay de una ciudad a cualquiera de las otras.

Solución Propuesta.

El modo de resolución de este problema fue empleando el concepto de permutación, este se examino en la materia de probabilidad, el cual es lo siguiente:

Permutación		Resultado
A ↗ B	C	ABC
A ↘ C	B	ACB
B ↗ A	C	BAC
B ↘ C	A	BCA
C ↗ A	B	CAB
C ↘ B	A	CBA

Se analizan las combinaciones posibles, con esto nos ayuda a identificar los caminos posibles a recorrer.

Para aplicar el concepto dinámico, hicimos lo siguiente:

Consideremos la siguiente matriz de caminos para observar el procesamiento del algoritmo procesamiento del algoritmo

	A	B	C	D
A	0	1	2	8
B	1	0	3	12
C	2	3	0	7
D	8	12	7	0

Elegimos una ciudad de donde partir, en este ejemplo elegiremos la C.

Lo primero es buscar los posibles caminos a partir de esta y a la vez, guardara la distancia, como se muestra en el ejemplo

CA	2
CB	3
CD	7

En seguida, aplicando el concepto de permutación, busca completar el camino con una tercera ciudad.

CAB	3
CAD	5
CBA	4
CBD	15
CDA	15
CDB	19

Ahora con un cálculo de 4 ciudades.

CABD	15
CADB	17
CBAD	12
CBDA	23
CDAB	16
CDBA	20

Para finalizar, calcula la distancia de regreso al lugar de inicio:

CABDC	22
CADBC	20
CBADC	19
CBDAC	25
CDABC	19
CDBAC	20

Así obteniendo que el camino mas fiable es **CBADC** o **CDABC** con una distancia total recorrida de 19.

Resultados

Si utilizamos el mismo ejemplo anterior tenemos que la matriz de distancias es la siguiente:

```
int [][] caminos = new int[][]{
    {0,    1,    2,    8},
    {1,    0,    3,   12},
    {2,    3,    0,    7},
    {8,   12,    7,    0}
};
```

Y vemos que como tenemos que elegir un resultado, es este caso nuestro algoritmo muestra la primera opción con el resultado mínimo **CBADC**

```
run:
Camino: 21032, valor=19
```

Conclusión.

El problema de la mochila puede ser analizado de diferentes maneras y con esto encontrar otros métodos de solución, no solo el empleado en este trabajo, una de estas, puede ser, analizando gráficamente con un plano, pero habría que ser más analíticos en la solución.

Y un posible error que podría encontrar en este modo de solución, es que, si se plantean un gran numero de ciudades puede tardar más en solucionarlo, pero esto no implica un resultado erróneo sino solo una demora.