

ALGORITMOS GREEDY

Práctica 3

Pablo Alaminos Morón
Celia Botella López
Jose Luis de la Rosa Morillas
David Pérez Núñez
Antonio Rodríguez Alaminos

ÍNDICE

- DEFINICIÓN DE ALGORITMO GREEDY
- DESCRIPCIÓN DEL PROBLEMA
- ALGORITMOS DE RESOLUCIÓN
 - CERCANÍA
 - INSERCIÓN
 - PROPIO
- Comparación de algoritmos

DEFINICIÓN

Es una estrategia de búsqueda por la cual se sigue una heurística consistente en elegir la opción óptima en cada paso local con el fin de llegar a una solución general óptima.

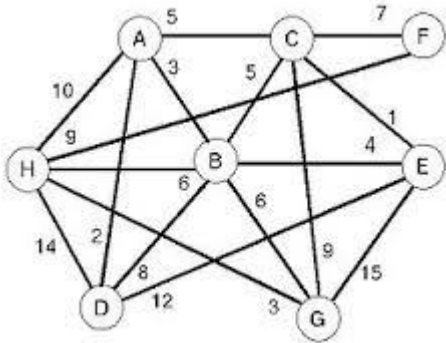
CARACTERÍSTICAS:

Un algoritmo Greedy o voraz se caracteriza por:

- La construcción de la solución por etapas.
- En cada momento elige un movimiento de acuerdo con un criterio de selección.
- No vuelve a considerar los movimientos ya seleccionados ni puede volver a modificarlos en posteriores etapas.
- Utiliza una función objetivo o criterio de optimalidad.

DESCRIPCIÓN DEL PROBLEMA

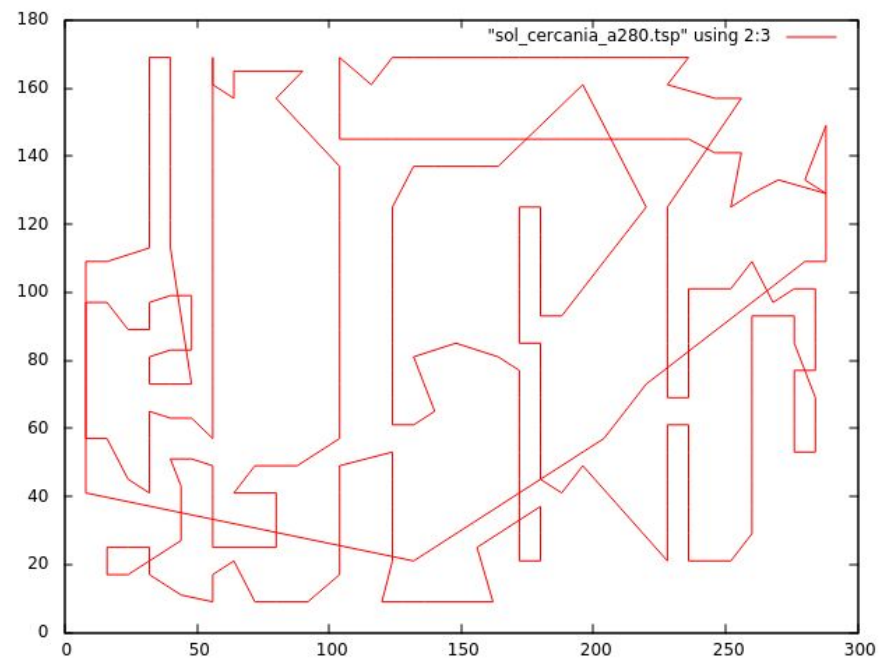
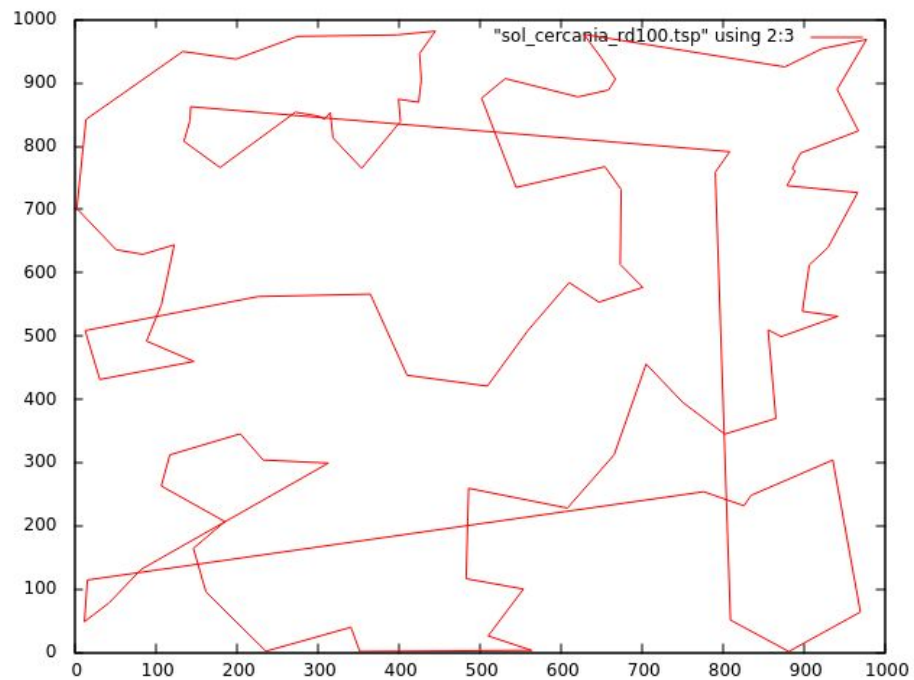
El problema del viajante de comercio consiste en dado un conjunto de ciudades y las distancias entre ellas determinar cuál es el camino más corto que pase por todas ellas y vuelva al punto de inicio, o lo que es lo mismo, un circuito hamiltoniano minimal.



Algoritmos de resolución

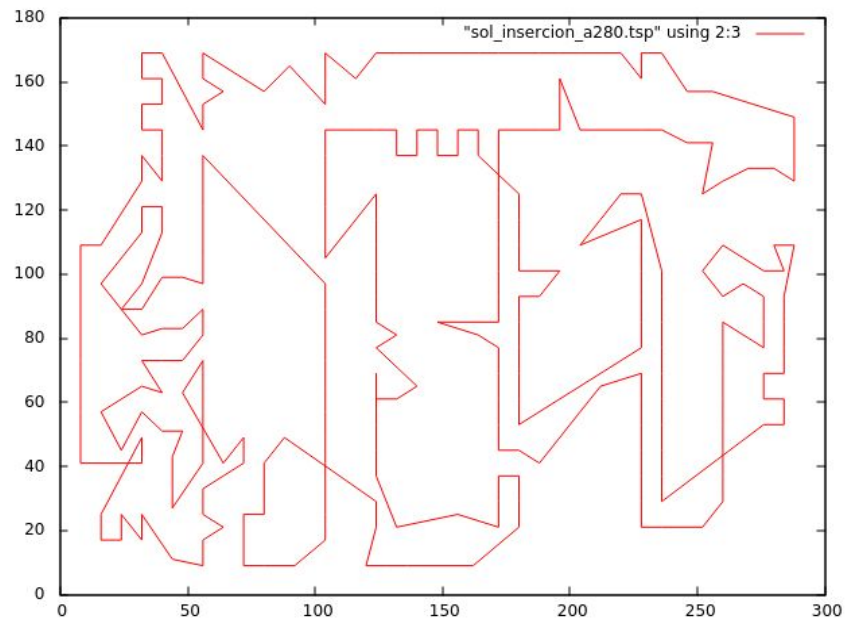
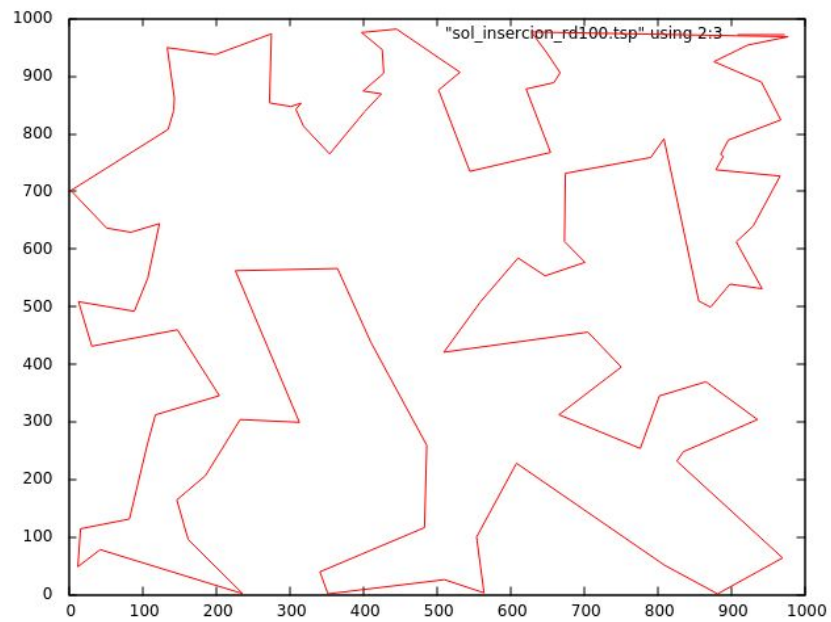
Algoritmo de Cercanía

Dado un vector de nodos, tomamos el primer elemento de este como inicio y lo introduciremos en el vector solución. Recursivamente, tomaremos el nodo más cercano al actual que no esté en la solución, lo introducimos en la solución y lo convertimos en el nodo actual. Repetimos el proceso $n-1$ veces y finalmente se vuelve a introducir el primer nodo. Sin embargo, esta resolución del problema no garantiza que el camino tenga longitud mínima.



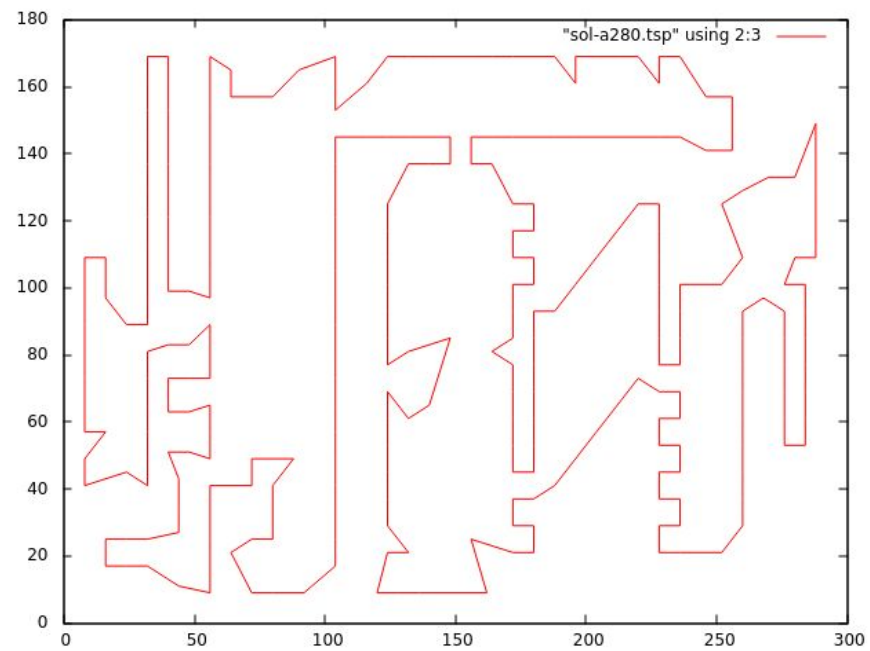
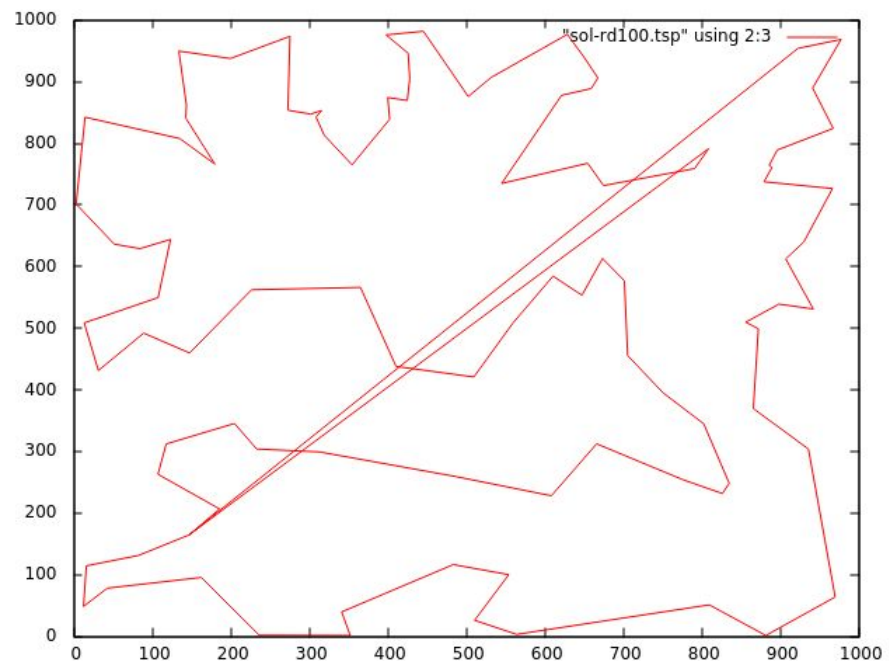
Algoritmo de Inserción

Dado un vector de nodos, partimos de un ciclo de tres nodos creado con las siguientes ciudades: la más al este, la más al oeste, y la más al norte. Partiendo de esto, vamos añadiendo más ciudades, las más cercanas a los lados, destruyendo los mismos lados en el proceso y creando otros enlaces entre los nodos afectados. La característica más atractiva de este algoritmo es que el camino que te aparece es el óptimo.

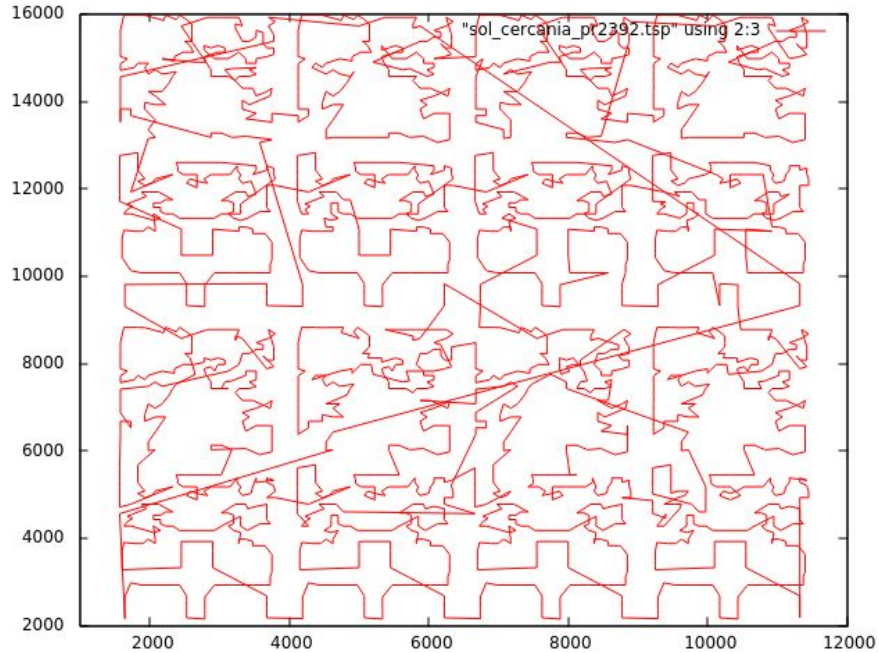


Algoritmo Propio

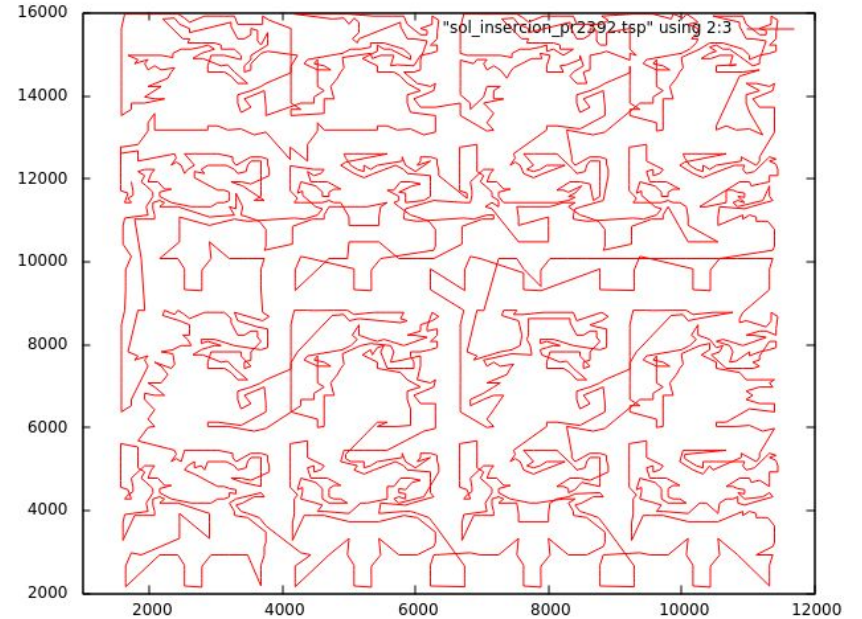
Dado nuestro vector de nodos, los introduciremos en una matriz $n \times n$ donde cada casilla i,j es la distancia entre el nodo i y el nodo j . A partir de aquí iremos introduciendo las distancias más pequeñas, teniendo en cuenta que cada nodo no puede aparecer más de dos veces. Repetimos el proceso n veces.



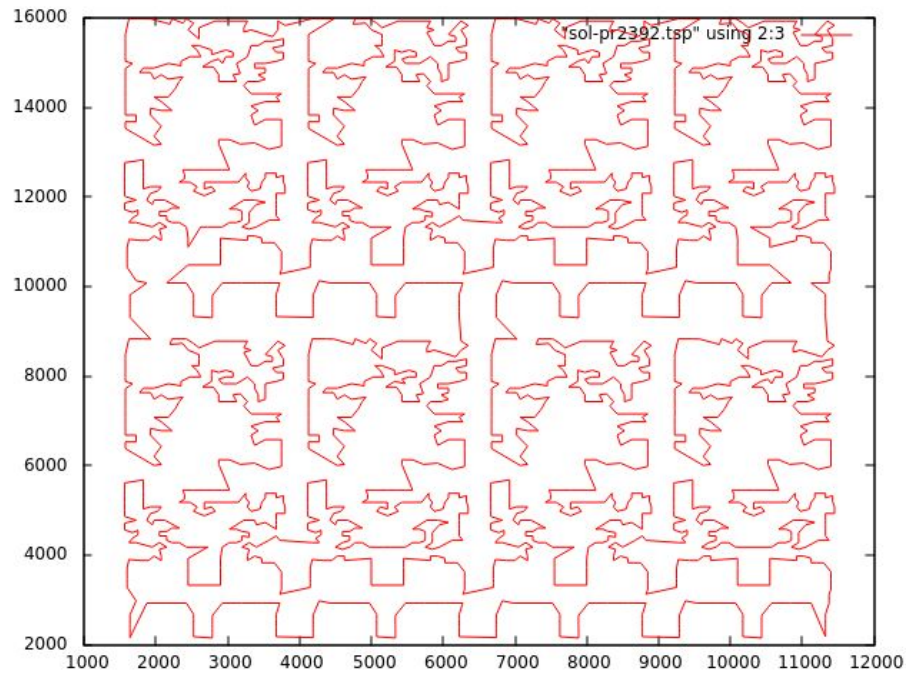
COMPARACIÓN DE GRÁFICAS



Algoritmo cercanía



Algoritmo de inserción



Algoritmo propio



¿Alguna pregunta?