



Taller 4

Caso de estudio: versión serial y paralela de un Algoritmo Genético Simple aplicado al problema del TSP

Parte 1- Funciones preliminares.

1. Generar vectores de coordenadas

Escribir una función que genera dos vectores con las coordenadas en X y Y, correspondientes a un caso particular del problema del TSP. Las coordenadas corresponden a valores enteros.

Parámetros de entrada:

nCiudades: Número de ciudades para el problema del transporte.

xMin, **xMax**: Valores mínimo y máximo para las coordenadas en X.

yMin, **yMax**: Valores mínimo y máximo para las coordenadas en Y.

Salida de la función:

coordenadasX: Vector con los valores de las coordenadas en X.

coordenadasY: Vector con los valores de las coordenadas en Y.

2. Crear matriz de distancias

Escribir una función que genera la matriz de adyacencias del grafo determinado por los nodos cuyas coordenadas vienen dadas en los vectores **coordenadasX** y **coordenadasY**. Esta matriz contiene las distancias entre los nodos del grafo determinado por tales coordenadas.

Parámetros de entrada:

nCiudades: Número de ciudades para el problema del transporte.

coordenadasX: Vector con los valores de las coordenadas en X.

coordenadasY: Vector con los valores de las coordenadas en Y.

Salida de la función:

matrizDistancia: Matriz que contiene las distancias entre las ciudades dadas por las coordenadas de entradas. Es una matriz cuadrada, simétrica y de diagonal 0.

3. Calcular la longitud del circuito

Escribir una función que retorna la longitud del camino dado como parámetro de entrada que es igual a la distancia recorrida al iniciar en la ciudad camino(i) y regresar al mismo lugar.

Parámetros de entrada:

camino: vector que muestra el orden de las ciudades recorridas en un circuito

matrizDistancia: requerido para obtener la distancia entre ciudades.

Salida de la función



Longitud: valor real que corresponde a la longitud del recorrido correspondiente al circuito dado.

4. Generar individuo

Escribir una función que genera una permutación con el recorrido por n ciudades, sin repetir ciudad.

Parámetros de entrada:

nCiudades: número de ciudades bajo análisis.

Salida de la función:

camino: vector de enteros consistente en una permutación de n ciudades entre 1 y n .

5. Función de adaptación

Retorna el valor de la adaptación, que permitirá determinar quien es el individuo mejor adaptado. Se calcula como la longitud máxima del recorrido en la población actual menos el valor de la longitud del individuo bajo análisis: $\text{longMaxima} + 1 - \text{longitud}$

Parámetros de entrada:

longitud: Valor de la longitud del individuo

longMaxima: Distancia de mayor recorrido entre todos los individuos de la población

Salida de la función

aptitud: Valor que corresponde a la adaptación al problema del transporte. Es un entero positivo, a mayor valor, mejor adaptación

6. Generar población inicial

Genera la población inicial en un arreglo bidimensional (matriz). El número de filas corresponde al tamaño de la población. Cada fila es una solución posible al problema del TSP, siendo una combinación que indica el orden del recorrido sobre n ciudades, es decir un vector numérico con n valores aleatorios entre 1 y n , sin que se repita ninguna valor. Además de la población se deben generar dos vectores adicionales, el uno corresponde a la distancia del recorrido de cada individuo y el otro a su aptitud, obtenido de la función de adaptación.

Parámetros de entrada:

tamPob: entero que indica el tamaño de la población a generar

nCiudades: entero que indica el número de ciudades correspondiente al problema.

matrizDistancia: matriz de adyacencias correspondiente a las distancias entre las ciudades a modelar.

Salida de la función

poblacion: matriz con los individuos que conforman la población.

distancias: arreglo unidimensional (vector) correspondiente a la distancia que corresponde al recorrido de cada individuo en la población

aptitud: vector correspondiente a la aptitud o adaptación de cada individuo.