

| | |
|----------------|------------|
| NOMBRE | GIANPIETRO |
| APELIDO | PANICO |
| UO | 299008 |
| GRUPO | L.6 |
| PRATICA | 4 |

****El tiempo de toda las operaciones se midió en un procesador intel core i7-1185G7, y 16,0 GB de memoria. El tiempo se expresa siempre en milisegundos.**

Describir el heurístico empleado en el algoritmo.

Una función heruística nos permite saber si un candidato concreto elegido en el algoritmo es válido o no.

En este caso, la heurística selecciona los elementos del vector *enemigos* que son \leq de los elementos del vector *grupoDefensaRapida*.

El mejor caso se da cuando el valor de un elemento del array grupoDefensaRapida es igual a un valor del array enemigos. En este caso, al array *asignación* a la ciudad correspondiente se le asignará inmediatamente el índice del array *grupoDefensaRapida*.

Si no es así, se recorre todo el array buscando el mejor candidato, eligiéndolo tomando como punto de referencia la resta entre los valores de los dos arrays.

¿Qué complejidad tiene el algoritmo diseñado? ¿Se podría cambiar el diseño para mejorar la complejidad?

El algoritmo tiene una complejidad de $O(n^2)$. Podría mejorarse ordenando el vector *enemigos* en orden ascendente.

Análisis del tiempo:

limite = 1000000 de $n=10$ a $n=160$, limite = 1000 de 320 en Adelante. Asi que los valores de n desde 10 hasta 160 se han dividido por 1000. ($76/1000 = 0,76$ por eso es un valor <50)

| n | Tiempo |
|-------|--------|
| 10 | 0,223 |
| 20 | 0,550 |
| 40 | 2,461 |
| 80 | 6,258 |
| 160 | 34,151 |
| 320 | 195 |
| 640 | 858 |
| 1280 | 3605 |
| 2560 | 15393 |
| 5120 | 66348 |
| 10240 | 285052 |

