

2.1.- Diseño del algoritmo

La práctica constará de un programa en java que resuelva el problema aplicando el esquema de Divide y Vencerás junto con una memoria de su implementación.

2.2.- Argumentos y parámetros

La práctica se invoca usando la siguiente sintaxis:

```
java skyline [-t][-h] [fichero entrada] [fichero salida]
```

o

```
java -jar skyline.jar [-t][-h] [fichero entrada] [fichero salida]
```

Los argumentos son los siguientes:

- **-t:** traza cada invocación recursiva de manera que se describa la parametrización de cada llamada recursiva.
- **-h:** muestra una ayuda y la sintaxis del comando.
- **fichero_entrada:** es el nombre del fichero del que se leen los datos, en este caso una ciudad representada por un conjunto de edificios $C = \{e_1, e_2, \dots, e_n\}$ y cada edificio representado por un rectángulo sobre un eje de coordenadas.
- **fichero_salida:** es el nombre del fichero que se creará para almacenar la salida con el *skyline*. Si el fichero ya existe, el comando dará un error. Si falta este argumento, el programa muestra el resultado por pantalla.

Por ejemplo:

```
$ java skyline -h <ENTER>
```

```
SINTAXIS: skyline [-t][-h] [fichero entrada] [fichero salida]
-t          Traza cada llamada recursiva y sus parámetros
-h          Muestra esta ayuda
[fichero entrada]  Conjunto de edificios de la ciudad
[fichero salida]   Secuencia que representan el skyline de la ciudad
```

2.3- Datos de entrada

El fichero de datos de entrada consta de una línea por cada edificio de la ciudad. La entrada termina cuando se llega al final del fichero.

Cada edificio se describe por tres valores enteros: x_1 , x_2 , h . Los valores x_1 y x_2 representan las posiciones inicial y final del edificio sobre el eje x. El valor h representa la altura. Se puede ver un ejemplo en el texto base.

2.4- Datos de salida

La salida es una secuencia de pares (x,y) , donde x indica la posición de la coordenada x en la que se produce en cambio de altura, e y la nueva altura. Véase ejemplo en el texto base.

2.5.- Implementación del algoritmo

El programa se desarrollará en Java siguiendo un diseño orientado a objetos. Los detalles del entorno recomendado se encuentran en la guía de la asignatura. Se valorará el diseño OO y la eficiencia del desarrollo.

3.- CUESTIONES TEÓRICAS DE LA PRÁCTICA

- 1) Analiza el coste computacional del algoritmo.
- 2) Describe alternativas al esquema utilizado, si las hay, y compara su coste con el de la solución realizada.