

Ejercicio 6: Influencia del proceso de compilación

Retome el ejercicio de ordenación mediante el algoritmo de la burbuja. Ahora replique dicho ejercicio pero previamente deberá compilar el programa indicándole al compilador que optimice el código. Esto se consigue así:

```
g++ -O3 ordenacion.cpp -o ordenacion_optimizado
```

Compare las curvas de eficiencia empírica para ver cómo mejora esto la eficiencia del programa.

- 1) Creamos el fichero 'ordenacion_optimizado.cpp' en el que incluimos tanto el programa completo como el algoritmo de ordenación de la burbuja.

- 2) Generamos el ejecutable mediante:

```
g++ -O3 ordenacion.cpp -o ordenacion_optimizado
```

- 3) Modificamos los campos del archivo 'ordenacion_optimizado.csh' con los datos que nos han pedido.

- 4) Para que el resultado de todas las ejecuciones se guarden en el archivo "ordenacion.dat", ejecutamos lo siguiente:

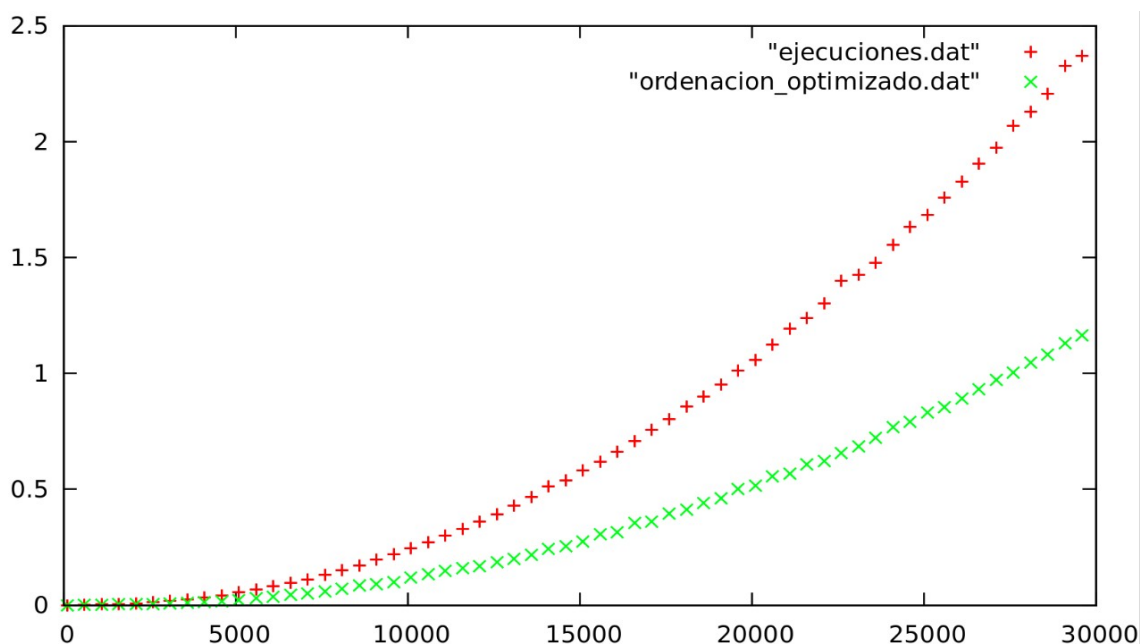
```
./ordenacion_optimizado.csh
```

- 5) Con la ayuda de gnuplot mostramos la eficiencia empírica y la comparamos con la del ejercicio1:

```
>gnuplot
```

```
>plot "ejecuciones.dat" , "ordenacion_optimizado.dat"
```

La gráfica resultante:



A pesar de ser el mismo código, cuando se compila de forma optimizada, se puede apreciar que es más eficiente que cuando se compila sin optimizar.

Características del ordenador

Fabricante	Lenovo
Procesador	Intel Core i7 4710HQ
Sistema Operativo	Ubuntu
Versión SO	14.04
RAM	8192MB
CPU	64 bits