**P2: Gabriel García Martínez L6**

1. **Describe en el documento en que consiste este método de selección, cuándo funciona mal y cuando no y que efecto tiene en el tiempo de ejecución.**

El método QuickSortFatal es una variante del QuickSort en el cual la elección del pivote es el elemento primero de la partición. Tanto en inverso como en ordenado no funciona desde los 20000 elementos, incluidos, ya que se produce stack overflow. En aleatorio, el método funciona con una complejidad de O(nlog n).

1. **TRABAJO PEDIDO**

**Todas las mediciones han sido realizadas con un ordenador con las siguientes características:**

Intel(R) Core(TM) i5-8250U CPU @ 1.60GHz 1.80 GHz

8,00 GB (7,87 GB usable)

* Ordenación por Inserción directa:

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **n** | **t aleatorio (10-5s)** | **t ordenado (10-5s)** | **t inverso (10-5s)** | **límite (10-5s)** |
| **10000** | 5.300,00 | 0,49 | 9.200,00 | 100 |
| **20000** | 17.300,00 | 0,86 | 34.000,00 | 100 |
| **40000** | 18.600,00 | 1,52 | 33.100,00 | 100 |
| **80000** | 67.200,00 | 2,80 | 121.500,00 | 100 |
| **160000** | 271.100,00 | 5,11 | 496.200,00 | 100 |
| **320000** | 1.076.900,00 | 9,96 | 2.123.400,00 | 100 |
| **640000** | 4.497.000,00 | 21,93 | 8.903.600,00 | 100 |
| **1280000** | 19.025.300,00 | 67,00 | 36.574.300,00 | 100 |
| **2560000** | 80.897.200,00 | 146,00 | >20 min | 100 |
| **5120000** | >25 min | 252,00 |  | 100 |
| **10240000** |  | 489,00 |  | 100 |
| **20480000** |  | 959,00 |  | 100 |
| **40960000** |  | 1.871,00 |  | 100 |
| **81920000** |  | 3.745,00 |  | 100 |
| **163840000** |  | 8.065,00 |  | 100 |

* Ordenación por selección:

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **n** | **t ordenado (ms)** | **t aleatorio (ms)** | **t inverso (ms)** | **limite(ms)** |
| **10000** | 14,20 | 31,30 | 54,00 | 1 |
| **20000** | 52,10 | 128,00 | 207,00 | 1 |
| **40000** | 161,00 | 463,00 | 616,00 | 1 |
| **80000** | 682,00 | 2.003,00 | 2.375,00 | 1 |
| **160000** | 2.803,00 | 7.776,00 | 9.541,00 | 1 |
| **320000** | 11.310,00 | 31.119,00 | 38.391,00 | 1 |
| **640000** | 45.390,00 | 130.750,00 | 152.649,00 | 1 |
| **1280000** | 201.041,00 | 536.404,00 | 619.036,00 | 1 |
| **2560000** | 1.037.697,00 | 2.229.437,00 | >30 min | 1 |
| **5120000** | >25 min | >40 min |  | 1 |

* Ordenación por burbuja:

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **n** | **t inverso (ms)** | **t aleatorio(ms)** | **T ordenado (ms)** | **limite (ms)** |
| **10000** | 71 | 134 | 18,20 | 1 |
| **20000** | 255 | 578 | 45,50 | 1 |
| **40000** | 1014 | 2301 | 205,00 | 1 |
| **80000** | 4068 | 9566 | 1.119,00 | 1 |
| **160000** | 16400 | 39604 | 4.437,00 | 1 |
| **320000** | 64942 | 158507 | 17.850,00 | 1 |
| **640000** | 258151 | 680477 | 71.689,00 | 1 |
| **1280000** | 1105136 | >25 min | 321.456,00 | 1 |
| **2560000** | >25 min |  | 1.561.344,00 | 1 |
| **5120000** |  |  | >25 min | 1 |

* QuickSort mediana a 3:

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **n** | **t inverso(10-5s)** | **t ordenado(10-5s)** | **t aleatorio(10-5s)** | **límite (10-5s)** |
| **10.000** | 18,50 | 17,20 | 17,00 | 100 |
| **20.000** | 33,50 | 29,70 | 31,40 | 100 |
| **40.000** | 67,00 | 60,00 | 62,00 | 100 |
| **80.000** | 141,00 | 130,00 | 141,00 | 100 |
| **160.000** | 295,00 | 269,00 | 297,00 | 100 |
| **320.000** | 593,00 | 563,00 | 688,00 | 100 |
| **640.000** | 1.241,00 | 1.181,00 | 1.582,00 | 100 |
| **1.280.000** | 2.607,00 | 2.477,00 | 3.994,00 | 100 |
| **2.560.000** | 5.549,00 | 5.287,00 | 10.480,00 | 100 |
| **5.120.000** | 12.589,00 | 11.049,00 | 29.545,00 | 100 |
| **10.240.000** | 24.672,00 | 23.025,00 | 99.400,00 | 100 |
| **20.480.000** | 53.478,00 | 47.847,00 | 210.300,00 | 100 |
| **40.960.000** | 105.983,00 | 99.712,00 | 462.600,00 | 100 |
| **81.920.000** | 228.168,00 | 208.810,00 | 1.098.600,00 | 100 |
| **163.840.000** | 463.366,00 | 509.476,00 | 2.867.800,00 | 100 |