

Prácticas de Algorítmica.
3º de Grado en Ingeniería Informática.
Curso 2023-2024.
Práctica 2

Objetivos.

Con esta práctica se pretende que el alumno implemente dos variantes de un algoritmo basado en la técnica divide y vencerás y que en base a los resultados obtenidos, concluya cual es el mejor.

Enunciado:

Implementad dos variantes del algoritmo del Quicksort para ordenar un conjunto de elementos de tipo entero, que se implementará usando un vector de la STL. El conjunto se rellenará aleatoriamente con valores enteros comprendidos entre 0 y 999999. Al igual que se hizo en la práctica 1, el usuario introducirá el tamaño mínimo, el máximo, el incremento y el número de repeticiones. Las variantes consistirán en lo siguiente:

- Variante del Quicksort que aparece en el tema 5.
 - En este caso, se implementará el algoritmo que aparece en los apuntes, seleccionando como pivote el elemento central, y se medirán los tiempos para cada valor del tamaño del vector y se irán mostrando por pantalla conforme se calculan.
- Variante del Quicksort usando la mediana para obtener el pivote.
 - En este caso, el usuario también introducirá lo siguiente:
 - Tamaño mínimo del vector a partir del cual se obtendrá el pivote por el método de la mediana. Es decir, por debajo de ese tamaño, se tomará el central como pivote, y por encima de ese tamaño, se usará la mediana,
 - Número de elementos del vector se han de tomar aleatoriamente para obtener la mediana. Este número ha de ser impar.
 - Para ordenar esos elementos aleatorios y obtener la mediana, se usará el método de selección.
 - Al igual que en la variante anterior, se medirán los tiempos para cada valor del tamaño del vector y se irán mostrando por pantalla conforme se calculan.

Un ejemplo de ejecución podría ser:

Para las dos variantes usar los siguientes datos de entrada: minSize= 100000, maxSize = 500000, incSize = 10000, numRep = 3,

Para la segunda variante usar numMediana = 5, numeroMinimo = 100.

De esta forma se generarán dos curvas de tiempos reales que se mostrarán en una sola gráfica.

Hay que hacer pruebas para ver cual es el número de elementos óptimo para calcular la mediana y el numero mínimo óptimo para ver cuando aplicar el método de la mediana. Hacer un pequeño informe con las pruebas realizadas.

Nota1: Los parámetros de entrada se introducirán en línea de comandos para poder realizar las pruebas más eficientemente.

Nota2: El programa principal no tendrá menú y tendrá una llamada a una función de medio nivel de prototipo void comparacionVariantesQuicksort(int nMin, int nMax, int incremento, int repeticiones, int nMediana, int nElementosMinimo);

La función que devolverá los tiempos de ambos métodos tendrá el siguiente prototipo:

void tiemposOrdenacionVariantesQuicksort(int nMin, int nMax, int incremento, int repeticiones, int nMediana, int nElementosMinimo, vector<double> &tiemposRealesQS1, vector<double> &tiemposRealesQS2, vector<double> &numeroElementos);

Fecha de comienzo: 10 de octubre de 2023

Fecha de Entrega: 31 de Octubre de 2023