

Prácticas de Algorítmica.
3º de Grado en Ingeniería Informática.
Curso 2016-2017

Práctica 1.

Objetivos.

Con esta práctica se pretende que el alumno se familiarice con el cálculo de tiempos de ejecución de un determinado algoritmo en función del tamaño del ejemplar y hacer una estimación empírica de esos tiempos en función de dicho tamaño. Para ello, el alumno deberá implementar un programa en C++ donde se calculen los tiempos de ejecución de dos de los siguientes algoritmos y posteriormente, utilizando un enfoque híbrido, calcular la complejidad computacional de esos métodos en función del tamaño del ejemplar.

Enunciado.

Implementad en C++ un programa que permita obtener los tiempos de ejecución de 2 de los siguientes algoritmos y a partir de esos tiempos estimar su complejidad computacional usando un enfoque híbrido.

Se seleccionará uno de los siguientes algoritmos en función del último dígito del dni (suponemos que el dígito es d8):

1. Producto de matrices cuadradas (si $d8 \bmod 3 == 0$)
2. Determinante de una matriz cuadrada (si $d8 \bmod 3 == 1$)
3. Inversa de una matriz cuadrada. (si $d8 \bmod 3 == 2$)

Por último, todos los alumnos han de implementar la versión recursiva para obtener el término n-ésimo de la sucesión de Fibonacci.

El tamaño del ejemplar en el caso de las operaciones matriciales es el número de filas o de columnas y en el caso de la sucesión de Fibonacci es el número de término a calcular.

Para realizar las pruebas y calcular los tiempos, se procederá como sigue:

1. El usuario introducirá, en línea de comandos, el valor mínimo, el valor máximo, y el incremento del tamaño del ejemplar. Por ejemplo, en el caso de cualquiera de las operaciones matriciales, si el mínimo es 100, el máximo es 500, el incremento es 10, se probará primero con 100 elementos, después con 110 y así hasta llegar a 500. Para cada tamaño del ejemplar se medirá el tiempo empleado en su ejecución.
2. Las matrices serán de elementos de tipo double y se rellenarán aleatoriamente con valores comprendidos entre -10 y 10.
3. Una vez obtenidos y almacenados los tiempos de ambos algoritmos, éstos se ajustarán a una curva por mínimos cuadrados teniendo en cuenta que la variable independiente es el tamaño del ejemplar y que la variable dependiente es el tiempo de ejecución. Al final del enunciado se explica como realizar el ajuste.
 1. Para el caso de las operaciones matriciales se probarán polinomios de grado 3 y 4, seleccionando en cada caso el que mejor se ajuste. Por ejemplo $t = a_0 + a_1 * n + a_2 * n^2 + a_3 * n^3$.
 2. Para el caso de la sucesión de Fibonacci se probará un ajuste de la forma $t = a_0 + a_1 * 2^n$. Aunque este ajuste no es lineal, se puede linealizar haciendo el cambio de variable $z = 2^n$, en cuyo caso se podría realizar un ajuste lineal.
 3. Para ambos casos la ecuación de la curva obtenida nos proporciona la complejidad computacional de ese algoritmo usando un enfoque híbrido. Es híbrido porque a priori sabemos que un polinomio de grado 3 o 4 se ajusta bien a las operaciones matriciales y que una función tipo exponencial se ajusta al

algoritmo recursivo de Fibonacci.

4. Ahora, teniendo en cuenta las curvas de ajuste obtenidas en el paso 5, se calcularán los tiempos estimados a partir de dichas curvas para cada uno de los tamaños de ejemplar probados para obtenerlas. De esta forma, al final de este paso tendremos los tiempos reales de los algoritmos, que son los observados, y los tiempos estimados mediante los ajustes por mínimos cuadrados.
5. Una vez obtenidos los tiempos estimados se guardarán en un fichero de texto para poder representarlos posteriormente usando el programa **gnuplot** (se suministra un ejemplo de uso). Se guardará por columnas en el siguiente orden: tamaño del ejemplar, tiempo real u observado y tiempos estimados
6. Para finalizar, el programa ha de mostrar las ecuaciones de las curvas ajustadas por mínimos cuadrados y sus coeficientes de determinación y dar la posibilidad al usuario si quiere hacer una estimación de tiempos para un determinado valor del tamaño del ejemplar, en cuyo caso mostrará el tiempo de esa estimación en días. Esta opción ha de poder repetirse hasta que el usuario introduzca un tamaño de ejemplar igual a 0. Esto es útil cuando el tiempo es muy elevado para un tamaño de ejemplar relativamente grande. Por ejemplo el término 100 de la sucesión de Fibonacci tardaría en calcularse varios días, o incluso años, y mediante la curva ajustada podemos obtener de una forma muy fiable el tiempo que tardaría en calcularse.

- **Notas estadísticas y matemáticas:**

Para estimar los parámetros de un ajuste polinómico de orden m (en nuestro caso m vale 1 o 3 o 4) se puede usar el siguiente sistema de ecuaciones (<http://es.slideshare.net/diegoegas/regresion-polinomial-2512264>):

$$\begin{array}{ccccccccccc} a_0 \cdot n & + & a_1 \sum x_i & + & a_2 \sum x_i^2 & + & \dots & + & a_m \sum x_i^m & = & \sum y_i \\ a_0 \sum x_i & + & a_1 \sum x_i^2 & + & a_2 \sum x_i^3 & + & \dots & + & a_m \sum x_i^{m+1} & = & \sum x_i y_i \\ a_0 \sum x_i^2 & + & a_1 \sum x_i^3 & + & a_2 \sum x_i^4 & + & \dots & + & a_m \sum x_i^{m+2} & = & \sum x_i^2 y_i \\ \vdots & & \vdots & & \vdots & & & & \vdots & & \vdots \\ a_0 \sum x_i^m & + & a_1 \sum x_i^{m+1} & + & a_2 \sum x_i^{m+2} & + & \dots & + & a_m \sum x_i^{2m} & = & \sum x_i^m y_i \end{array}$$

En este sistema de ecuaciones las incógnitas son los valores de los a_i . Los valores x_i se corresponden con el tamaño del ejemplar y los valores y_i se corresponden con los tiempos observados.

Cada sumatorio tendrá tantos sumandos como valores de x_i hayan usado para el tamaño del ejemplar, por ejemplo si en Fibonacci se calculan los términos que van del 1 al 100, cada sumatorio tendrá 100 elementos.

Por otra parte, necesitamos tantas ecuaciones en el sistema como coeficientes tengamos que calcular para la curva, por ejemplo en un ajuste de un polinomio de tercer grado necesitamos 4 ecuaciones, ya que hay cuatro coeficientes en la curva.

Para resolver el sistema se puede usar la regla de Cramer, según la cual cada incógnita se obtiene como un cociente de dos determinantes.

El coeficiente de determinación será la varianza de la y estimada mediante el ajuste (tiempos estimados mediante el ajuste) dividida entre la varianza de la y observada (tiempos observados).

Fecha de comienzo: 22 de setiembre de 2016.

Fecha de Entrega: 13 de Octubre de 2016.