­­­

**Práctica\_2\_4 v2.** Divide y Vencerás

**GUIÓN DE LA PRÁCTICA**

**1.- Objetivos de la práctica**

Dados dos enteros ***a***, n>0, la potencia an puede calcularse haciendo n-1 multiplicaciones (lo que supone una complejidad lineal con respecto a n, que es el tamaño del problema), pero esto podría resultar muy costoso si n es grande. No obstante, utilizando un esquema de tipo Divide y Vencerás, podemos escribir una función que calcule an en un tiempo logarítmico ().

Planteado el problema de la potenciación de enteros, en esta práctica trataremos de analizar cuan efectiva puede llegar a ser la técnica Divide y Vencerás según el tamaño de n (exponente) comparando la ejecución del algoritmo implementado la solución trivial iterativa (a\*a\*a….n….\*a), una solución recursiva simple (an=a\* an-1) y otra aplicando la técnica Divide y Vencerás (an = an/2 \* an/2).

**2.- Actividades a realizar**

**Actividad 1**: Resolver el problema de la potenciación con el algoritmo trivial iterativo:

**función** potencia1 (a,n: entero) **devuelve** entero largo

**principio**

r:=1;

**para** i:=1 **hasta** n **hacer**

r:=r\*a

**fpara**;

**devuelve** r

**fin**

1. Implementar la función de potencia1 **incluyendo las OE (hacer desde cero con pseudocodigo o código en la sesión).**
2. Realizar un nuevo código que llame 10 veces (diez pruebas) a la función potencia1 para **medir las OE y el tiempo de ejecució**n generando aleatoriamente tanto la base (a) como el exponente (n) con una semilla dada (“42”) para una distribución uniforme en un rango de 1 a 100, es decir, tanto a como n tomarán aleatoriamente un valor entre 1 y 100. Visualizar por pantalla las siguientes variables (una línea para cada prueba):

* a: base de la potencia
* n: exponente
* resultado
* número de operaciones elementales
* tiempo de ejecución en milisegundos

1. Incluya una copia de pantalla en la **hoja de trabajo** y calcule la complejidad temporal de la solución propuesta.

**Actividad 2**: Resolver el problema de la potenciación con un algoritmo recursivo simple basado en las propiedades elementales de las potencias (an=a\* an-1) :

**función** potencia2 (a,n: entero) **devuelve** entero largo

**principio**

**si** (n==0)

devuelve 1

**sino**

devuelve (a\*potencia2(a, n-1));

**fsi**

**fin**

1. Implementar la función de potencia1 **incluyendo las OE (hacer desde cero con pseudocodigo o código en la sesión).**
2. Realizar un nuevo código que llame 10 veces (diez pruebas) a la función potencia1 para **medir las OE y el tiempo de ejecució**n generando aleatoriamente tanto la base (a) como el exponente (n) con una semilla dada (“42”) para una distribución uniforme en un rango de 1 a 50. Visualizar por pantalla las siguientes variables (una línea para cada prueba):
   * a: base de la potencia
   * n: exponente
   * resultado
   * número de operaciones elementales
   * tiempo de ejecución en milisegundos

1. Incluya una copia de pantalla en la **hoja de trabajo** y calcule la complejidad temporal de la solución propuesta.

**Actividad 3**:  Resolver el problema de la potenciación aplicando la técnica de Divide y Vencerás para reducir el tamaño del problema a la mitad (an = an/2 \* an/2):

**función** potencia3 (a,n: entero) **devuelve** entero largo

**principio**

**si** (n==0)

devuelve 1

**sino**

devuelve (potencia3(a, n/2)\* potencia3(a, n/2));

**fsi**

**fin**

1. Implementar la función de potencia1 **incluyendo las OE (hacer desde cero con pseudocodigo o código en la sesión).** Para aplicar divide y vencerás a este problema hemos de tener en cuenta que, aunque de manera general an = an/2 × an/2, como trabajamos con enteros, esto solo se cumplirá si n es par pero no para *n* impar. Debemos distinguir por tanto entre la solución trivial (n=0), cuando n sea par (an = an/2 × an/2) y cuando n sea impar (an = a(n-1)/2 × a(n-1)/2 × a)
2. Realizar un nuevo código que llame 10 veces (diez pruebas) a la función potencia1 para **medir las OE y el tiempo de ejecució**n generando aleatoriamente tanto la base (a) como el exponente (n) con una semilla dada (“42”) para una distribución uniforme en un rango de 1 a 50. Visualizar por pantalla las siguientes variables (una línea para cada prueba):
   * a: base de la potencia
   * n: exponente
   * resultado
   * número de operaciones elementales
   * tiempo de ejecución en milisegundos

1. Incluya una copia de pantalla en la **hoja de trabajo** y calcule la complejidad temporal de la solución propuesta.

**Actividad 4**: Modifique el código de la Actividad1\_2, Actividad2\_2 y Actividad3\_2 para que los resultados se escriban sobre un fichero de texto separado por puntos y comas (resultados.csv). Abra el fichero con una aplicación de hojas de cálculo:

1. Ordene los datos según el orden ascendente de n (exponente).
2. Calcule el número medio de OE según la implementación de potencia utilizada
3. Calcule la desviación estándar de los tiempos de ejecución según la implementación de potencia utilizada

**3.- Entrega fin sesión**

Los archivos con los programas correspondientes a las actividades se deberán entregar, a través de Moodle, al finalizar la sesión de prácticas, junto con la Hoja de Trabajo del estudiante. Todo el código entregado debe compilar y ejecutar, comente aquel código que está pendiente para la entrega final de la práctica para que no de errores de compilación.

Las líneas de código deben estar comentadas con su funcionalidad, tome como ejemplo los comentarios en el código proporcionado y las fórmulas utilizadas para el cálculo de las operaciones elementales en cada línea de código, así como las fórmulas teóricas utilizadas para el cálculo teórico.

**4.- Después de asistir al laboratorio**

* + Completar el código y el documento de trabajo que no haya dado tiempo a realizar en clase.
  + El código y documento entregado en esta entrega debe realizarse, aunque se haya terminado todo el trabajo en clase. El código debe ser una ampliación de lo realizado en clase no puede ser completamente nuevo.
  + Realizar la Entrega\_Practica\_2\_4 en un fichero comprimido con la hoja de trabajo:
    - Actividades 1-3:
      * Código con la implementación de las tres funciones de potenciación: practica\_2\_4.cpp
      * Hoja de Trabajo con lo solicitado en los apartados 3 de las tres actividades.
    - Actividad 4:
      * Resultados.csv
      * Hoja de cálculo (comparación.xls)