

PRACTICA SUMADORES – PARTE 2 – COMUNICACIÓN ENTRE PROCESOS POR FICHEROS

Crearemos un programa que permita parametrizar el lanzamiento de sumadores, que vuelque el contenido de las sumas en ficheros y que permita al programa principal recuperar las sumas de los ficheros parciales.

```
package es.multiproceso;

public class Sumador {

    public static int sumar(int n1, int n2){
        int suma=0;
        if (n1>n2){
            int aux=n1;
            n1=n2;
            n2=aux;
        }
        for (int i=n1; i<=n2; i++){
            suma=suma+i;
        }
        return suma;
    }

    public static void main(String[] args){
        int n1=Integer.parseInt(args[0]);
        int n2=Integer.parseInt(args[1]);
        int suma=sumar(n1, n2);
        System.out.println(suma);
        System.out.flush();
    }
}
```

```

public class Main {

    static final int NUM_PROCESOS=4;
    static final String PREFIJO_FICHEROS="fich";

    public static void lanzarSumador(
        int n1, int n2,String fichResultados) throws
IOException{
        String comando;
        comando="es.multiproceso.Sumador";

        File directorioSumador;
        directorioSumador=new File("C:\\Users\\"+
            "alumno\\workspace\\"+
            "MultiProceso1\\bin\\");
        File fichResultado=new File(fichResultados);
        ProcessBuilder pb;
        pb=new ProcessBuilder("java",
            comando,
            String.valueOf(n1),
            String.valueOf(n2) );
        pb.directory(directorioSumador);
        pb.redirectOutput(fichResultado);
        pb.start();
    }

    public static int getResultadoFichero(
        String nombreFichero){

        int suma=0;
        try {
            FileInputStream fichero=
                new FileInputStream(
                    nombreFichero);

            InputStreamReader fir=
                new InputStreamReader(
                    fichero);

            BufferedReader br=new BufferedReader(fir);
            String linea=br.readLine();
            suma= new Integer(linea);
            return suma;
        } catch (FileNotFoundException e) {
            System.out.println(
                "No se pudo abrir "+nombreFichero);
        } catch (IOException e) {
            System.out.println(
                "No hay nada en "+nombreFichero);
        }
        return suma;
    }
}

```

```

public static int getSumaTotal(int numFicheros){
    int sumaTotal=0;
    for (int i=1; i<=NUM_PROCESOS;i++){
        sumaTotal+=getResultadoFichero(
            PREFIJO_FICHEROS+String.valueOf(i) );
    }
    return sumaTotal;
}

/* Recibe dos parámetros y hará
 * la suma de los valores comprendidos
 * entre ambos parametros
 */
public static void main(String[] args) throws IOException,
InterruptedException{
    int n1=Integer.parseInt(args[0]);
    int n2=Integer.parseInt(args[1]);
    int salto=( n2 / NUM_PROCESOS ) - 1;
    for (int i=1;i<=NUM_PROCESOS; i++){
        System.out.println("n1:"+n1);
        int resultadoSumaConSalto=n1+salto;
        System.out.println("n2:"+resultadoSumaConSalto);
        lanzarSumador(n1, n1+salto ,
            PREFIJO_FICHEROS+String.valueOf(i));
        n1=n1 + salto + 1;
        System.out.println("Suma lanzada...");
    }
    Thread.sleep(5000);
    int sumaTotal=getSumaTotal(NUM_PROCESOS);
    System.out.println("La suma total es:"+
        sumaTotal);
}
}

```