

## **UD1. PRÁCTICA 1: Definir procesos que se executen en paralelo**

### **ENUNCIADO:**

Nesta práctica imos facer uso da clase ProcessBuilder de Java para a execución de procesos.

### **OBXECTIVOS:**

- Coñecer o funcionamento básico de ProcessBuilder
- Implementar unha pequena aplicación que execute procesos.

### **DURACIÓN ESTIMADA:**

- 100 minutos

### **RECURSOS NECESARIOS:**

- Máquina virtual con Debian ou Ubuntu
- Eclipse

### **ENTREGABLE:**

A través do proxecto de Gitlab, que cada alumno terá para este módulo, deberedes crear unha carpeta chamada "UD1-Ex1" que conterá o proxecto desta práctica, que será un proxecto maven..

### **ENUNCIADO:**

Tomando como referencia o código da aplicación da UD subida ao gitlab.com, implementar unha pequena aplicación que recibirá un número e o sumará co seu consecutivo, sumaralle 1 e ese resultado sumarao ao anterior resultado, así tantas veces como se indique. Polo tanto a función que imos invocar como un proceso independente desde a nosa aplicación Java terá os seguintes parámetros:

- num\_base: Número base a partir do cal comeza a realizar a suma co seguinte consecutivo.
- num\_veces: Número de veces que fai a suma da sucesión numérica co seu consecutivo.
- time\_sleep: Tempo en milisegundos durante o cal a función fai unha parada entre cada execución.

Polo tanto teremos unha función:

$$F(\text{num\_base}, \text{num\_veces}, T) = (\text{numbase} + (\text{num\_base}+1)) + ((\text{num\_base}+1) + (\text{num\_base}+2)) + \dots +$$

$((\text{num\_base}+n) + (\text{num\_base}+n+1))$

Debemos ter en conta que entre cada unha das sumas debe facer unha parada de T milisegundos (sendo T o parámetro que se pasa á función). Entre cada unha das interaccións, deberemos sacar o cálculo realizado, así como o momento (data, como hora, minutos e segundos) a un ficheiro de texto por cada un dos procesos (por exemplo. "result1.txt" e "result2.txt" en función do proceso que o lance).

A nosa app debe lanzar desde o *main*, a execución dos dous procesos:

- Lanzar a execución do primeiro proceso facendo uso do *ProcessBuilder* e cos seguintes parámetros F(10, 20, 1500)
- Lanzar a execución do segundo proceso cos parámetros F(10, 20, 300) usando procesos independentes a través do *ProcessBuilder*

Debes reparar na execución de ambos procesos e a partires dos ficheiros xerados, comprobar si se están executando de xeito concorrente.