

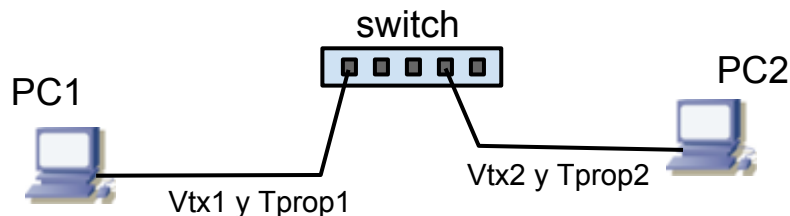
## Práctica 3 – Programando en Java

### 1- Objetivos

El objetivo de esta práctica es familiarizarse con la programación en Java. Para ello se realizará un programa que calcule tiempos de envío de paquetes en un escenario

### 2- Escenario

Tenemos una red formada por un conmutador Ethernet en el que hay conectados dos ordenadores. Se pretende realizar un programa que calcule el tiempo que empleara el envío de un paquete de la longitud determinada entre los dos y que genere los datos necesarios para graficar este tiempo de entrega de datos en función del tamaño que se desee enviar



### 3- Especificación [puntuacion 2% tiempo estimado 1h30m]

Realice un programa que calcule el tiempo que tarda un paquete en atravesar el sistema para un conjunto de tamaños a partir de los variables

p1Vtx - velocidad de transmisión del puerto 1 (en bps)

p1Tprop - tiempo de propagación en el cable conectado al puerto 1 (en s)

p2Vtx - velocidad de transmisión del puerto 2 (en bps)

p2Tprop - tiempo de propagación en el cable conectado al puerto 2 (en s)

tam1 - tamaño inicial del barrido (en bytes)

tam2 - tamaño final del barrido (en bytes)

tamdelta - incremento del tamaño en el barrido (en bytes)

#### Uso:

#### java Grafica

Imprime una línea por cada tamaño indicado en las variables desde tam1 hasta tam2 en incrementos tamdelta. En cada línea indica el tiempo que tardara un paquete de ese tamaño en ser entregado en el ordenador 2 desde que empezó a ser enviado por el ordenador 1. El formato de la línea sera

```
<tamaño del paquete en bytes> <tiempo en segundos empleado>
```

Las variables indicadas estaran definidas en el propio fichero Grafica.java

**Ejemplos:**

```
$ java Grafica
500 0.004402
750 0.006602
1000 0.008802
1250 0.011002
1500 0.013202
```

Pruebe el programa generando diferentes barridos con diferentes valores. Para dirigir la salida a un fichero por ejemplo caso1 puede hacer

```
$ java Grafica > caso1
```

Pruebe a obtener la grafica con el programa gnuplot.

```
$ gnuplot
> plot "caso1" using 1:2 with linespoints
```

**Checkpoint 1:**

Debe enseñar esto al profesor para que compruebe su funcionamiento y subirlo a la página de entrega antes de finalizar la sesión de practicas.

Valor del chekpoint 2%

#### 4- Versión avanzada [opcional +1% estimado 30m]

Realice un programa en java que cumpla con el siguiente patrón. Recuerde que las especificaciones no son negociables si el programa no cumple las especificaciones cuenta como no hecho.

**Uso:**

```
java Grafica <puerto1Vtx> <puerto1Tprop> <puerto2Vtz> <puerto2Tprop>
<tamaño> [<tamaño<max> <incremento>]
```

Si tamaño<max no está presente calcula el tiempo en segundos que emplea un paquete del tamaño indicado en atravesar el sistema desde que el ordenador 1 empieza a enviarlo por el puerto 1 hasta que el ordenador en el puerto 2 lo recibe completamente. El formato de salida será una unica linea con dos campos

```
<tamaño del paquete en bytes> <tiempo en segundos empleado>
```

Los parametros de entrada puerto1Vtx y puerto2Vtx se deben interpretar como valores en bps. Los parametros de entrada puerto1Tprop y puerto2Tprop deben interpretarse como valores en segundos. Los parametros de entrada tamaño y tamaño<sub>max</sub> si esta presente deben interpretarse en bytes.

Si tamaño<sub>max</sub> esta presenta el programa generara el resultado para cada tamaño desde el indicado hasta tamaño<sub>max</sub> en el incremento indicado

**Ejemplos:**

```
$ java Grafica 1000000 0.000001 1000000 0.000001 500  
500 0.004402
```

```
$ java Grafica 1000000 0.000001 1000000 0.000001 500 1500 250  
500 0.004402  
750 0.006602  
1000 0.008802  
1250 0.011002  
1500 0.013202
```

**Checkpoint 2:**

Debe enseñar esto al profesor para que compruebe su funcionamiento y subirlo a la página de entrega antes de finalizar la sesión de practicas.

El segundo checkpoint no puntuara nada si no se ha hecho el primero correctamente

Valor del checkpoint 1%