

Práctica 4 – Programando en Java

1- Objetivos

El objetivo de esta práctica es familiarizarse con la programación usando arrays en Java. Para ello se realizará un programa que simule transmisiones en CSMA

2- Escenario

Tenemos una sistema CSMA p-persistente, en el que cuando voy a transmitir y encuentro el canal libre se decide si transmitir o no con una probabilidad p . En cuanto veo libre el canal con una probabilidad p transmito y con una probabilidad $1-p$ espero un tiempo Δt . Si espero que pase ese Δt microsegundos con una probabilidad p transmito y con $1-p$ vuelvo a esperar otros Δt y así sucesivamente. Eso quiere decir que a veces tardare mas y otras menos en transmitir. Queremos estudiar este tiempo. Para eso haremos un programa que haga experimentos, pruebe una y otra vez a transmitir y al final haga estadísticas.

3- Especificación [puntuacion 2% tiempo estimado 1h]

Realice un programa que haga una simulación del tiempo que tarda un paquete en empezar a transmitirse con el canal vacio en CSMA p-persistente.

Uso:

```
java SimulaCSMAp p deltat
```

Simula la transmision en CSMA p-persistente generando probabilidades para ver cuando se decide a transmitir imprimiendo cada paso para comprobar el funcionamiento

Ejemplos:

```
$ java SimulaCSMAp 0.3 0.000001
Intentando transmitir con persistencia p=0.3000
slot 1 ... espero
slot 2 ... espero
slot 3 ... espero
slot 4 ... espero
slot 5 ... transmito
conseguido en 5 intentos tiempo= 0.000004 s
```

Pruebe el programa con diferentes valores asegurándose que da resultados razonables.

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- Calcular estadísticas (avanzado) [+1% tiempo estimado 1h]

Realice un programa que haga la simulación anterior n veces calculando cuantas veces el resultado es 1 slot, 2 slots... para poder obtener el histograma. Como teóricamente podría esperar infinitos slots le diremos un número máximo de slots en que estamos interesados.

Uso:

```
java SimulaCSMAp p deltat n max
```

Simula n veces la transmisión en CSMA p -persistente generando el histograma de los max primeros slots (0 a $max-1$)

Ejemplos:

```
$ java SimulaN 0.3 0.000001 8 5
```

```
transmision en 1 intentos  
transmision en 2 intentos  
transmision en 1 intentos  
transmision en 1 intentos  
transmision en 1 intentos  
transmision en 3 intentos  
transmision en 3 intentos  
transmision en 1 intentos
```

Histograma:

```
0 slots -- 0 veces (0.00 %)  
1 slots -- 5 veces (62.50 %)  
2 slots -- 1 veces (12.50 %)  
3 slots -- 2 veces (25.00 %)  
4 slots -- 0 veces (0.00 %)  
Mas de 5 slots -- 0 veces (0.00 %)
```

Pruebe el programa con diferentes valores asegurándose que da resultados razonables. Incluyendo que haga el experimento miles de veces (puede quitar el imprimir cada caso y dejar solo el histograma final)

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 prácticas.

Valor del checkpoint +1%