

## PROYECTO DE SIMULACIÓN POR EVENTOS DISCRETOS (VALOR 20%) PROTOCOLO PARA EL ACCESO AL MEDIO EN UNA RED

En grupos de **no más de 2 personas** se debe programar en Java un modelo estocástico discreto de simulación del sistema explicado a continuación.

### I. DESCRIPCIÓN DEL SISTEMA A SIMULAR

Suponga que se tiene una **LAN** (Local Area Network) compuesta por tres computadoras **A, B, C**. La capa de red de cada máquina recibe archivos para su envío hacia máquinas fuera de la red. Los archivos recibidos por cada una de estas computadoras deben ser enviados al "Servidor Antivirus" para que descarte aquellos que lleven virus, el resto de los archivos son enviados por este servidor al "Router" del sistema para que se encargue de enviarlos a su destino.

Hay **una sola línea** de transmisión que deben compartir las **3 computadoras** para enviar sus archivos al **Servidor Antivirus**. **La transferencia de un archivo desde el Antivirus al Router, se hace inmediata, o sea, en tiempo "0".**

La manera de utilizar la única línea de transmisión por estas 3 máquinas para el envío de sus archivos al Antivirus sigue un **protocolo de polling o Token Ring<sup>1</sup>**

Este protocolo funciona de la siguiente manera:

1. En **cualquier momento dado** **solamente una de las máquinas de la red puede transmitir información** a través del medio (el cable).
2. El protocolo maneja un **token** que es **asignado durante un período de tiempo T** a cada máquina de manera secuencial.
3. Durante ese **tiempo T**, la máquina que tiene el *token* tiene **acceso exclusivo al cable o línea de transmisión**. **Durante este período debe transmitir los archivos que le dé tiempo de transmitir, siempre respetando la etiqueta de prioridad del archivo y prefiriendo transmitir los archivos más grandes primero.**
4. Si una máquina no tiene archivos que transmitir entonces **se libera el token de forma inmediata y se asigna a la siguiente máquina. Si después de enviar al menos un archivo, el remanente de tiempo es menor al archivo de menor tamaño en cola, el token se libera** y se asigna de inmediato a la siguiente máquina.

La máquina **A** recibe archivos con una distribución de su tiempo entre arribos **exponencial con una media de 5 segundos**.

La máquina **B** recibe archivos con la siguiente distribución para el tiempo entre arribos:

$$f(x) = x/40 \quad 8 \leq x \leq 12 \text{ segundos}$$

La máquina **C** recibe archivos **con una distribución entre arribos normal**, con una media de 4 segundos y una varianza de 0.01 segundos cuadrados o sea es  **$n(\mu=4, \sigma^2=0.01)$** .

Cada archivo recibido tiene una etiqueta de **prioridad (1 ó 2)**, en donde los archivos con prioridad 1 son más importantes y deben ser transmitidos de forma prioritaria. Cada archivo tiene una probabilidad de **0.25** de ser de **prioridad 1**.

---

<sup>1</sup> **Token Ring** fue uno de los primeros protocolos para el control del acceso al medio en la capa de enlace de datos, fue desarrollado por **IBM** y era el utilizado para realizar la comunicación en las primeras máquinas en red, actualmente fue estandarizado en el protocolo **IEEE 802.5**.

Suponga que cuando se recibe un archivo en una de las tres computadoras, éste se divide en **paquetes de 1024 k-bits**, si el último paquete tiene menos de **1024 k-bits**, entonces se completa con 0's para alcanzar dicha cantidad de bits.

El **tamaño de cada archivo (M)**, medido en número de paquetes de 1024 kb (k-bits), es uniforme **entre 1 y 64**. Es decir, tiene **distribución uniforme en {1, 2, 3, ..., 64}**.

El proceso de transferir a la línea **cada paquete** tarda, en cualquiera de las 3 computadoras, **1/2 segundo**. Una vez puesto en la línea el último paquete del archivo que está transmitiendo, el tiempo que tarda en llegar **el último bit** de ese paquete al ANTIVIRUS es de **1 segundo (tiempo de propagación)**. Es hasta ese momento que se considera que el archivo llegó a dicho servidor.

El **servidor Antivirus** tiene suficiente espacio para colocar en cola todos los paquetes que le vayan llegando antes de revisarlos y descartar el archivo o enviarlo al *Router*.

El proceso de revisión por virus de **cada archivo** puede fallar, en cuyo caso se comienza a revisar otra vez. Se ha estimado que la **revisión número "I"** del archivo, **tarda  $(M \cdot I)/8$  segundos**, donde **M es el tamaño del archivo**. La **probabilidad de fallar** en la revisión en un intento cualquiera es igual a **0.15**. El resultado de la revisión se obtiene al terminar el tiempo de procesamiento del archivo. Si un archivo no pasa la prueba **no pasa la prueba** porque se le encontró algún virus en 3 intentos, el archivo se descarta.

Finalmente, cuando un archivo pasa la revisión del antivirus, **es enviado al Router de la red** (vamos a asumir que **el tiempo que dura es en ser enviado y llegar al router despreciable y se contabiliza como 0**).

El **Router** tiene disponibles **2 líneas de transmisión** los cuales son utilizados para poner los archivos en la red (uno por archivo). El tiempo que se dura en transferir a una línea un archivo es de **M/64 segundos**, donde **M es el tamaño del archivo**. Pasado ese tiempo, el archivo es considerado como enviado. **El router sin embargo, es uno barato, y no tiene buffers para almacenar archivos en espera cuando todos los hilos están ocupados**, por lo que **cuando el antivirus termina un archivo, debe revisar el status del router, para saber si tiene hilos disponibles**. En caso afirmativo, el archivo es pasado al router, en caso contrario, el archivo se mantiene en el antivirus hasta que haya uno libre.

## II. ESTADÍSTICAS Y PRESENTACIÓN EN PANTALLA DE LA SIMULACIÓN

### 1. Al inicio se le debe solicitar al usuario al menos:

- Número de veces que se va a correr la simulación
- Tiempo total en segundos para correr cada vez la simulación (Máximo de tiempo)
- Si desea ver la simulación correr en modo lento o no (Se implementa solo con un **delay de 1 segundo entre cada cambio de reloj**)
- El tiempo T durante el cuál a cada máquina se le asigna el token.

### 2. Mientras corre la simulación en pantalla debe presentarse (como mínimo):

- El reloj** del sistema.
- La **longitud de la cola** de las 3 máquinas (dividido por prioridad) y el antivirus.
- La máquina que tiene el token.
- El tiempo T durante el cual se asigna el token.
- La cantidad de hilos ocupados en el router.
- El tipo de **evento** que se está procesando.

### 3. Al final de cada corrida deben desplegarse las siguientes estadísticas:

- Tamaño promedio de la cola en las 3 máquinas y el antivirus.**
- Tiempo promedio de "permanencia de un archivo en el sistema"** (desde que llega a la máquina y hasta que sale del router) en general y por prioridad. (Note que los descartados no se toman en cuenta para esta estadística)
- Promedio de archivos enviados por cada turno del token.**
- Número promedio de revisiones del antivirus por archivo.**

### 4. Al final de la serie de corridas solicitada por el usuario deben también desplegarse las estadísticas anteriores, pero como un promedio de todas las de cada corrida y un **intervalo de confianza** para el **tiempo promedio de permanencia en el sistema** para cada "archivo".

### III. ANÁLISIS DEL MODELO IMPLEMENTADO

Una vez implementada la simulación, debe de estudiar el proceso y analizar el impacto del cambio de ciertas variables dentro del sistema modelado. Por ejemplo:

1. ¿Qué pasa si el tiempo de transmisión de los servidores A, B y C se incrementa?
2. ¿Qué pasa si el tamaño máximo de los archivos se duplica?
3. ¿Qué pasa si se añade memoria al router?

Incluya dentro de la documentación de su proyecto un análisis de cómo se podría mejorar la eficiencia del sistema (reduciendo el tiempo promedio que pasa un archivo en el sistema). Note que para esto probablemente deberá modificar su simulación para ver el resultado de cambiar valores de ciertos parámetros (o podría también incluirlos dentro de los parámetros generales de la simulación para que el usuario los determine).

**IV. DOCUMENTACIÓN MÍNIMA ESPERADA Y ENTREGA DEL PROYECTO:** La documentación del proyecto se deberá entregar en un documento realizado en LaTeX. Para esto cada grupo de proyecto utilizará un proyecto en sharelatex que será asignado por el profesor del curso. El documento deberá ser entregado impreso el día 1 de octubre al inicio de la clase. Además se debe entregar la documentación y el código fuente mediante la plataforma del curso a más tardar las 5pm del 1 de octubre. El código fuente del documento será revisado en sharelatex. El documento debe incluir:

- a) Un **manual de instalación** describiendo el ambiente necesario para correr el programa así como la forma de correrlo.
- b) Descripción del sistema a simular (este enunciado).
- c) Un diagrama de clases de la solución implementada.
- d) Un **diagrama de flujo o un diagrama que utilice pseudocódigo** simple describiendo claramente la manera en la que se hará la simulación del sistema. Deben usarse nombres descriptivos, o una lista describiendo cada "nombre" del diagrama.
- e) Una descripción de las distribuciones utilizadas y su mecanismo de generación aleatoria.
- f) Archivos con **código fuente** y con el ejecutable (**debe presentarse el ejecutable**) debe haber una clara y completa documentación interna.
- g) **Estadísticas** obtenidas: impresión de la pantalla con las estadísticas obtenidas para los valores de prueba: **correr la simulación 15 veces, correrlo por 5000 segundos cada vez.**
- h) El **análisis del sistema** indicado en el punto III (del sistema simulado, no el de su programa de simulación).
- i) Código fuente (todo lo no relacionado con el UI).
- j) Descripción de los **problemas NO RESUELTOS** al momento de enviar la simulación y una idea de cómo resolver cada uno.

### V. Calificación del proyecto:

Documentación:	20%	
Lógica y diseño de la simulación:	50%	(se revisa en doc., código y corrida)
Ejecución correcta del programa:	30%	

### VI. Elementos de evaluación:

- Calidad del código fuente.
- Calidad del diseño de la solución
- Todos los elementos descritos en el programa del curso.