

# INDICACIONES PRÁCTICA

## 5

\* Nombre de Archivo Java:

***Mandar dentro del archivo comprimido la carpeta sistemaDistribuido***

\* Ejemplo Nombre Archivo Comprimido:

GutierrezAvilesLuisP5TSOAD0413B.zip

Nota: Al igual que en la práctica 1,2 ,3 y 4, hay que seguir las indicaciones del archivo .pdf y se debe de poner en todos los archivos .java que sean modificados su nombre completo, código y Número de práctica para la cual se hicieron estas correcciones.

Desarrollar paso a paso las 4 prácticas diferentes es un poco difícil por lo que sólo doy un poco de ayuda extra en caso de que lo requieran:

### A) **Almacenamiento en Buzones:**

En receiveVerdadero esto hay que hacer:

Tratas de Obtener Buzón

Si me regresa un null

{

    se registra proceso en TablaRecepcion

    se duerme al hilo // Se trata de un cliente

}

sino

## Taller De Sistemas Operativos Avanzados - 2013B - D04

---

```
si el buzón esta vacio
{
    se registra proceso en TablaRecepcion
    se duerme al hilo // Esto es lo que ya se hace en la práctica
2
}

sino
{
    Saco el primer elemento del buzón, que se maneja como
    Cola (LynkedListed)
    Hago la copia del arreglo que saco del buzón al arreglo del
    servidor
    Libero el espacio del buzón //El método poll hace esto de
    forma automática al momento de sacar el elemento
}
```

Dentro del run del micronucleo se tendría que poner algo así:

```
mientras espere datagramas
{
    recibe datos de la red
    saca los datos del mensaje recibido // ID origen, ID destino
    e IP origen

    Si el paquete es un AU
    {
        Se le avisa al cliente y se le despierta.
    }

    Si el paquete es un TA
    {
        Guardo solicitud en tabla de Mensajes a Retransmitir
        Espero 5 Segundo
        Envió de Nuevo la solicitud
    }
```

```
si el proceso existe en el micronucleo
{
    Si el proceso esta en la tabla de Recepcion
    {
        Aplica todo igual como en la práctica 2
    }

sino
    {
        Si hay espacio en el buzón
        {
            Registro al proceso origen en la Tabla de
Emision //Igual que en la práctica 2
            Guardo solicitud en el buzón //Haciendo copia a
un arreglo de bytes nuevo y almacenando este arreglo nuevo
            Recorro el final de la cola //El método offer hace
esto en forma automática al momento de meter la solicitud al
buzon
        }
        sino
        {
            Elaboro paquete TA
            Envio Paquete TA
        }
    }
}
sino
{
    Elaboro paquete Address Unknown
    Envió Address Unknown
}
```

B) **Almacenamiento Temporal en Núcleo:**

En receiveVerdadero esto hay que hacer:

## Taller De Sistemas Operativos Avanzados - 2013B - D04

---

si la tabla de Almacenamiento Temporal esta vacia ó No tiene Tabla de Almacenamiento Temporal (Ya que los clientes también convocan a Receive y no van a estar registrados en esta tabla)

```
{  
    se registra proceso en TablaRecepcion  
    se duerme al hilo // Esto es lo que ya haces en la práctica 2  
}
```

sino

```
{  
    Saco el elemento de la Tabla de Almacenamiento Temporal  
    Hago la copia del arreglo que saco de la tabla al arreglo del servidor  
    Borró el mensaje de la Tabla de Almacenamiento  
}
```

Dentro del run del micronucleo tendrías que poner algo así: mientras espere datagramas

```
{  
    recibe datos de la red  
    saca los datos del mensaje recibido // ID origen, ID destino e IP origen  
    Si el paquete es un AU  
    {  
        Se le avisa al cliente y se le despierta.  
    }  
}
```

Si el paquete es un TA

```
{  
    Guardo solicitud en tabla de Mensajes a Retransmitir  
    Espero 5 Segundo  
    Envió de Nuevo la solicitud  
}
```

si el proceso existe en el micronucleo

```
{
```

## Taller De Sistemas Operativos Avanzados - 2013B - D04

---

```
Si el proceso esta en la tabla de Recepcion
{
    Aplica todo igual como en la práctica 2
}

sino
{
    Si no hay otro mensaje guardado para el ID
Destino en la Tabla de Almacenamiento Temporal
    {
        Guardo solicitud en la Tabla de Almacenamiento
Temporal
        Registro al proceso origen en la Tabla de Emision
//Como lo hacías en la práctica 2
        Inicio hilo Cronometro que dure 10 Segundos
        Al terminar el cronometro revisar si el mensaje
sigue en la Tabla de Almacenamiento Temporal

        Si Sigue
        {
            Elimino Mensajes de la Tabla Temporal
            Elaboro paquete TA
            Envio Paquete TA
        }
        Si no sigue
        {
            No hago nada ya que suponemos que el
servidor tomó el mensaje
        }

    }
    Si no se encuentra en la Tabla de Recepción
    {
        Elaboro paquete TA
    }
}
```

```
        Envio Paquete TA
    }
}
}
Si no existe el proceso destino
{
    Si no hay otro mensaje guardado para el ID Destino en
    la Tabla de Almacenamiento Temporal
    {
        Guardo solicitud en la Tabla de Almacenamiento
        Temporal
        Inicio hilo Cronometro que dure 10 Segundos
        Al terminar el cronometro revisar si el mensaje sigue
        en la Tabla de Almacenamiento Temporal

        Si Sigue
        {
            Elimino Mensajes de la Tabla Temporal
            Elaboro paquete Address Unknown
            Envió Address Unknown
        }
        si no sigue el mensaje
        {
            No hago nada ya que suponemos que el servidor
            tomó el mensaje
        }
    }
    Si no esta vacía la tabla de almacenamiento temporal
    {
        Elaboro paquete Address Unknown
        Envió Address Unknown
    }
}
```

## C) Servidor de Nombres:

## Taller De Sistemas Operativos Avanzados - 2013B - D04

---

Se pide utilizar dos tipos de sendVerdadero, uno donde el primer parametro es un entero (Es el que se hizo en la práctica 2) y otro donde el primer parametro es un String (Este es el send que se encargará de tomar los datos del Servidor de Nombres).

Al iniciar el sistema se deberá de abrir una ventana extra aparte del Micronucleo que será la que represente al servidor de Nombres. Esta ventana consta de dos areas de texto la primera será para listar a todos los servidores que se encuentra registrados y la segunda para escribir en ella todos los eventos relacionados con el servidor de nombres.

Este servidor de nombres es un proceso más que podrá atender tres tipos de solicitudes: Alta, Baja y Búsqueda (Muy similar al programa Conector), las dos primeras son invocadas por los servidores y la tercera por los clientes.

En `procesoServidor.java` del paquete `sistemaDistribuido.sistema.clienteServidor.modousoario` deberás de hacer dos pequeñas correcciones:

A) Antes del ciclo `while(Continuar)` tendrá que hacer un señalamiento al nucleo para que invoque al método `registrar` del programa conector, le tendría que mandar como datos el Nombre del Servidor y una ASA es decir una clase que herede de `ParMaquinaProceso` que tenga el ID y el IP del servidor. El servidor de nombres le contestará con un entero con el que sabrá en que posición fue registrado dentro del Servidor de Nombres, este mismo número servirá para hacer el deregistro.

B) Después del Ciclo `While(Continuar)`, es decir, al momento de cerrar la ventana del servidor deberás hacer otro señalamiento al nucleo para que invoque el método `baja` del Servidor de Nombres, es suficiente si este método lo invocas con el número de registro que le hayan dado al servidor.

## Taller De Sistemas Operativos Avanzados - 2013B - D04

---

OJO, el send del servidor no es modificado, se deja tal cual como en la práctica 2, es decir invoca al send con el ID del cliente como primer parametro.

En procesoCliente.java del mismo paquete se tendrá que hacer tan sólo la modificación de quitar el 248 que estaba en las prácticas anteriores y poner el nombre del servidor que quiera le atienda, lógicamente pondras el mismo nombre que utilizaste en procesoServidor.java. Esto hará que se invoque al otro método Send del micronucleo.

En Micronucleo.java del paquete sistemaDistribuido.sistema.clienteServidor.modosMonitor deberás pues corregir el método sendVerdadero nuevo, incluso tiene el comentario de que se utiliza para la práctica 5 para quienes les tocó Buzones. En estos momentos esta vacío por lo que deberas llenarlo de código.

¿Qué es lo que tiene que hacer este método?

Es simple, tiene que utilizar el primer parametro, es decir, el nombre del servidor para hacer un señalamiento al nucle e invocar al método busqueda del servidor de nombres, el servidor de nombres contestará con el ASA del servidor, es decir el IP y el ID del servidor en caso de que se encuentre algún servidor así llamado o con null si el servidor no se encuentra.

En caso de recibir null simplemente se le avisa al cliente que no puede ser atendida su solicitud y se le despierta.

En caso de recibir un ASA con datos pues se procede a hacer lo mismo que en el send de la práctica 2, es decir, llenas los encabezados, los primeros 4 bytes con el ID origen, los segundos 4 bytes con el ID destino. Se procede a la creación de un datagramPacket utilizando el arreglo de bytes que se recibe como parametro y la IP que se recibió del servidor de nombres, se procede a obtener el DatagramSocket y hacer el envío del DatagramPacket anteriormente creado.



Eso son las correcciones principales que tendrías que hacer.

Deberás de crear por lo menos 2 archivos.java nuevos donde el primero será la ventana del Servidor de Nombres, coloca este archivo de preferencia en las carpetas de los paquetes visual, si se te complica no hay problema que lo ubiques dentro de otro paquete que te sea más conveniente.

El segundo archivo a crear es el relacionado con los métodos del servidor de Nombres, es decir, la creación del procesoServidor de nombres. En este archivo podrás manejar una HashTable para guardar a los servidores que se registren y poder realizar las búsquedas, trata que este archivo sea un proceso como lo son el ProcesoCliente.java y el ProcesoServidor.java puedes revisar el código de estos 2 para que veas como se hace la creación de estos procesos, así mismo revisa el código de sus ventana para que puedas asignar este proceso de Servidor de Nombres a la ventana correspondiente.

### c) **Transmisión Rala:**

No debes de tomar ningún dato de la interfaz gráfica del micronúcleo.

Al dar el botonazo al "Solicitar" se sigue utilizando el mismo sendVerdadero de la práctica 2. Antes si no se encontraban los datos en la tabla de emisión se sacaban los datos de la interfaz, ahora si no se encuentran los datos en la tabla de Emisión debes de buscarlo en la tabla de direccionamiento de procesos remotos, si tampoco ahí encuentras nada pues mandas un paquete LSA y esperas 5 segundos, vuelves a

## Taller De Sistemas Operativos Avanzados - 2013B - D04

---

revisar la tabla de direccionamiento de procesos remotos, si aún sigues sin datos vuelves a mandar el LSA y espera otros 5 segundos (Sólo mandas hasta 3 intentos de LSA), en caso de que no encuentres datos en tres intentos despiertas al cliente y le avisas que no hay servidor.

En caso de encontrar algún dato en la tabla de direccionamiento de procesos remotos pues de ahí sacas los datos de la IP y de la ID destino y haces el envío de forma regular como en la práctica 2.

Hay dos tablas de direccionamiento, una en la que guardas los datos de todos los servidores que mandan un FSA, esa es la tabla de direccionamiento de procesos remotos y otra donde se registran los servidores al iniciar su servicio haciendo un señalamiento al núcleo esta se llama tabla de direccionamiento de procesos locales. De la primera tabla tienes que eliminar alguna entrada cuando recibas un AU, es decir sacaste los datos de la tabla de direccionamiento y mandaste tu mensaje pero resulta que el servidor ya no se encontraba y te regresaron un AU, eliminas pues esos datos, obtienes otros datos de otro servidor y re-envías la solicitud.

En la segunda tabla eliminas los datos al momento de cerrar el servidor, el servidor después del ciclo "while(continuar)" deberá de hacer otro señalamiento al núcleo pidiéndole lo borre de la tabla de direccionamiento de procesos locales ya que no dará servicio. En ninguno de los dos casos es necesario que se haga de forma gráfica la inserción o eliminación de datos, basta con que tengan las funcionalidades para poner y quitar datos.

Para no complicar tanto la práctica pueden mandar el LSA sólo a LocalHost sin problemas, no es necesario que hagas un Broadcast por toda la red.

En la tabla de direccionamiento de procesos remotos la búsqueda se hace por número de servicio (supongamos que seguimos utilizando el 248) y todos los servidores que abras

## Taller De Sistemas Operativos Avanzados - 2013B - D04

---

dan ese mismo servicio, tienen una ID distinta pero dan el mismo servicio. Si se usa una HashTable y guardas el número de servicio como <key> al llegar los datos de otro servidor se sobre-escribirían los datos ya que manejan el mismo número de servicio, por eso se refiere que debes programar de forma que sea posible obtener todos los servidores posibles para un mismo dest (para un mismo número de servicio). Para esto se recomienda el uso de la LinkedList, en donde puedes almacenar los datos, no importa si repites la misma <key> estos datos no se sobre-escribieran.

### Criterios de Evaluación

- Cumplir con las “Reglas de Operación y Evaluación del Taller de Sistemas Operativos Avanzados”.
- Fecha de asignación: 06 de Noviembre del 2013
- Fecha planeación de entrega: **26 de Noviembre**  
**(UNICA OPORTUNIDAD)**
- Observación: Ninguna
- Calificación en base a cobertura de requerimientos y fecha de entrega en horas clase