

***Instituto Politécnico Nacional***  
***Escuela Superior de Cómputo***



## ***Desarrollo de Sistemas Distribuidos***

### ***Tarea 4. Implementación de un token-ring***

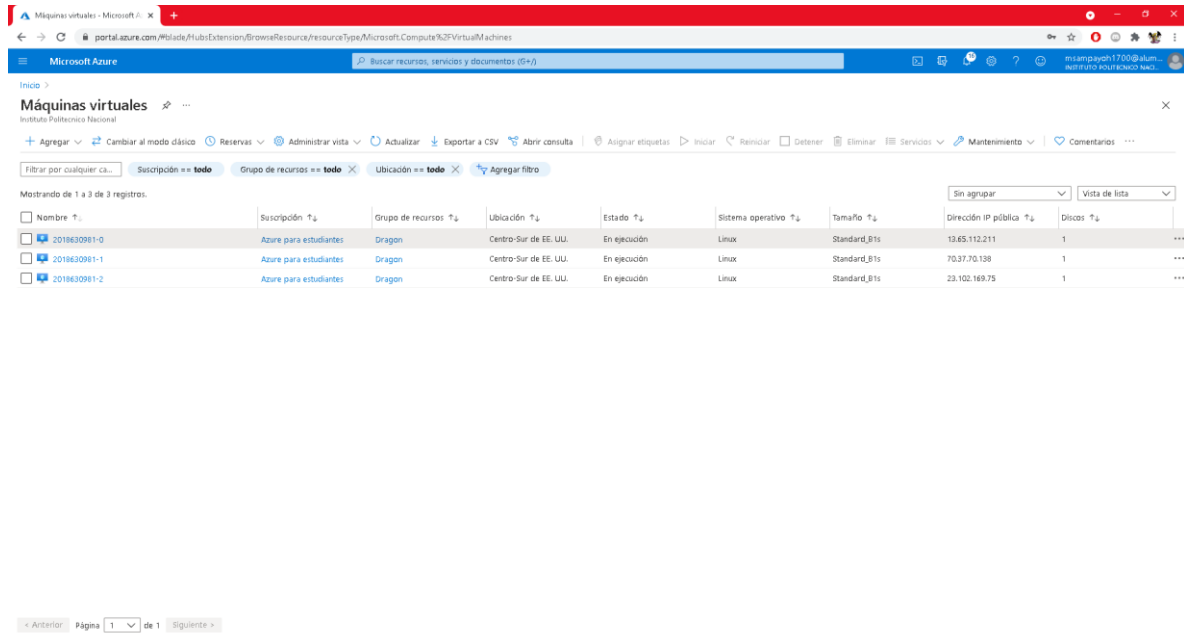
***Nombre:*** Sampayo Hernández Mauro

***Grupo:*** 4CV1

***Profesor:*** Pineda Guerrero Carlos

## Creación de las máquinas virtuales:

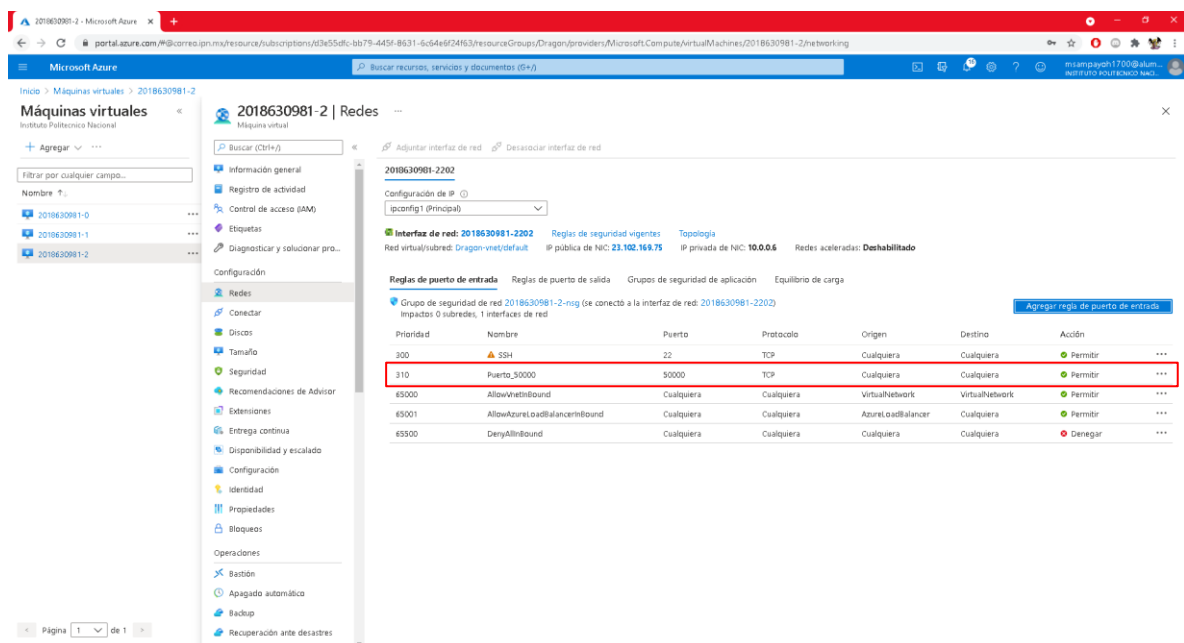
Primero se realiza la creación de 3 máquinas virtuales de Ubuntu en Azure, cada una con 1 CPU, 1 GB de RAM y disco HDD estándar.



The screenshot shows the 'Máquinas virtuales' (Virtual Machines) page in the Microsoft Azure portal. It displays a table with 3 virtual machines. The columns are: Nombre, Suscripción, Grupo de recursos, Ubicación, Estado, Sistema operativo, Tamaño, Dirección IP pública, and Discos. The first three rows represent VMs 0, 1, and 2, all created by 'Azure para estudiantes' in the 'Dragon' resource group, located in 'Centro-Sur de EE. UU.', and are in 'En ejecución' (Running) state. They all use 'Linux' as the operating system and 'Standard\_B1s' as the size. The public IP addresses are 13.65.112.211, 70.37.70.138, and 23.102.169.75 respectively. Each VM has 1 disk.

Nombre	Suscripción	Grupo de recursos	Ubicación	Estado	Sistema operativo	Tamaño	Dirección IP pública	Discos
2018630981-0	Azure para estudiantes	Dragon	Centro-Sur de EE. UU.	En ejecución	Linux	Standard_B1s	13.65.112.211	1
2018630981-1	Azure para estudiantes	Dragon	Centro-Sur de EE. UU.	En ejecución	Linux	Standard_B1s	70.37.70.138	1
2018630981-2	Azure para estudiantes	Dragon	Centro-Sur de EE. UU.	En ejecución	Linux	Standard_B1s	23.102.169.75	1

Configuramos cada máquina virtual para que puedan recibir conexiones a través del puerto 50000, abriendo dicho puerto en cada máquina virtual y seleccionando el protocolo de TCP para dicho puerto. A continuación, se muestra dicho puerto configurado en la máquina virtual del nodo dos.

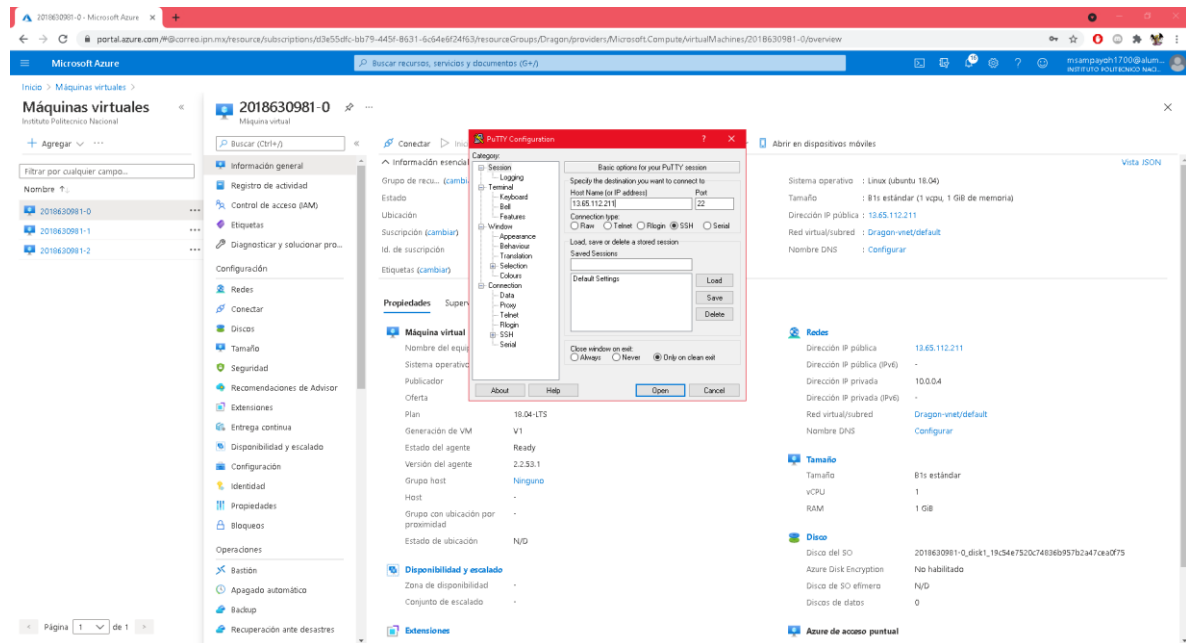


The screenshot shows the 'Redes' (Networks) page for the virtual machine '2018630981-2'. It displays the 'Configuración de IP' (IP Configuration) section, showing the 'ipconfig1 (Principal)' interface. Below this, the 'Reglas de seguridad' (Security Rules) section is visible, showing a list of rules. The rule 'Grupo de seguridad de red 2018630981-2-nsg' is selected, and its 'Reglas de puerto de entrada' (Inbound Port Rules) are shown. The rule 'Puerto\_50000' is highlighted, showing it is configured for TCP on port 50000, allowing traffic from any source to any destination.

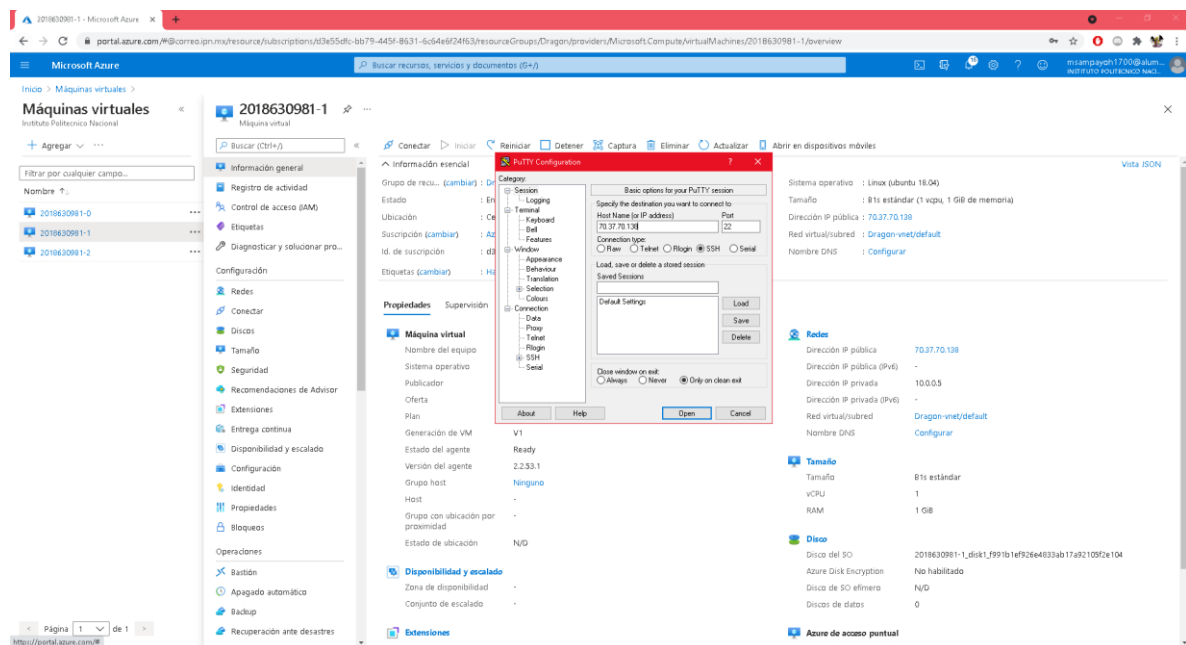
Prioridad	Nombre	Puerto	Protocolo	Origen	Destino	Acción
300	SSH	22	TCP	Cualquiera	Cualquiera	Permitir
310	Puerto_50000	50000	TCP	Cualquiera	Cualquiera	Permitir
45000	AllowVnetInbound	Cualquiera	Cualquiera	VirtualNetwork	VirtualNetwork	Permitir
45001	AllowAzureLoadBalancerInbound	Cualquiera	Cualquiera	AzureLoadBalancer	Cualquiera	Permitir
45500	DenyAllInbound	Cualquiera	Cualquiera	Cualquiera	Cualquiera	Denegar

Ejecutamos Putty.exe, para así acceder a cada una de las máquinas virtuales por medio de sus respectivos IPs.

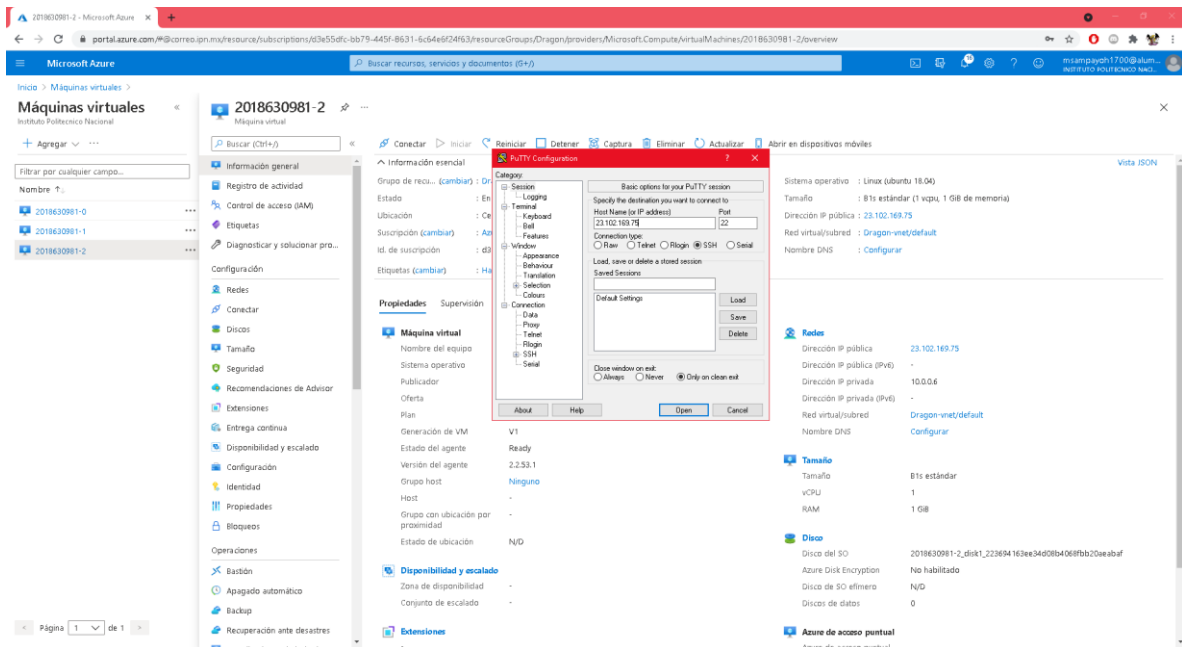
Aquí se muestra el acceso a la máquina virtual para el nodo 0:



Aquí se muestra el acceso a la máquina virtual para el nodo 1:

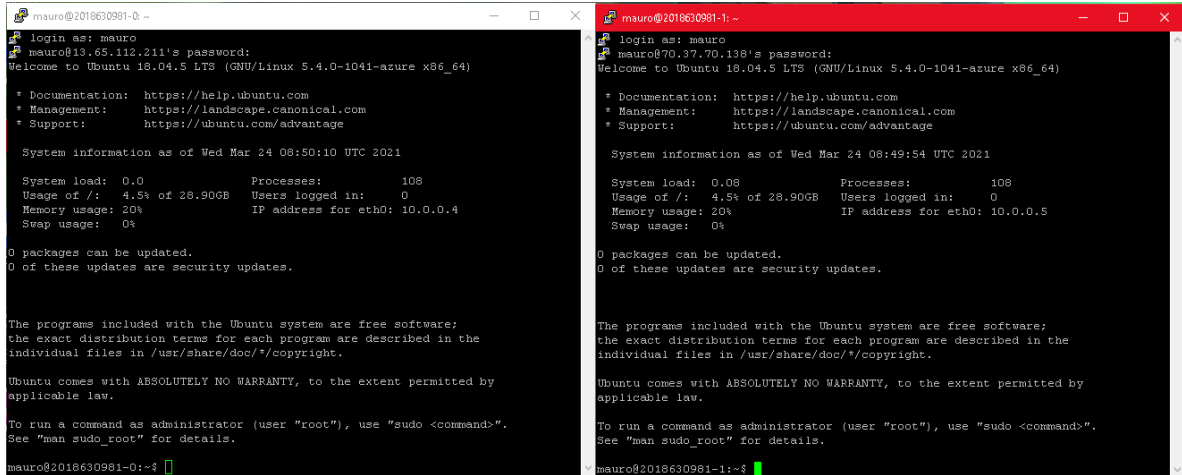


Finalmente se muestra el acceso a la máquina virtual para el nodo 2:



## Compilación del programa:

Una vez iniciamos cada una de las máquinas virtuales, accedemos a ellas por medio del usuario y la contraseña que configuramos previamente al momento de realizar la creación de las mismas



```
mauro@2018630981-2:~$  
login as: mauro  
mauro@23.102.149.75's password:  
Welcome to Ubuntu 18.04.5 LTS (GNU/Linux 5.4.0-1041-azure x86_64)  
  
 * Documentation:  https://help.ubuntu.com  
 * Management:    https://landscape.canonical.com  
 * Support:        https://ubuntu.com/advantage  
  
System information as of Wed Mar 24 08:50:01 UTC 2021  
  
System load:  0.0      Processes:      108  
Usage of /:   4.5% of 28.90GB    Users logged in:  0  
Memory usage: 20%      IP address for eth0: 10.0.0.6  
Swap usage:   0%  
  
0 packages can be updated.  
0 of these updates are security updates.  
  
The programs included with the Ubuntu system are free software;  
the exact distribution terms for each program are described in the  
individual files in /usr/share/doc/*/*copyright.  
  
Ubuntu comes with ABSOLUTELY NO WARRANTY, to the extent permitted by  
applicable law.  
  
To run a command as administrator (user "root"), use "sudo <command>".  
See "man sudo_root" for details.  
  
mauro@2018630981-2:~$
```

Instalamos el jdk 8 para todas las maquinas virtuales por medio del comando “sudo apt install openjdk-8-jdk”, y comprobamos que la instalación se haya realizado correctamente por medio del comando “java --version”

```
mauro@2018630981-0:~$ java --version  
openjdk 11.0.10 2021-01-19  
OpenJDK Runtime Environment (build 11.0.10+9-Ubuntu-0ubuntu1.18.04)  
OpenJDK 64-Bit Server VM (build 11.0.10+9-Ubuntu-0ubuntu1.18.04, mixed mode, sharing)  
mauro@2018630981-0:~$  
  
mauro@2018630981-1:~$ java --version  
openjdk 11.0.10 2021-01-19  
OpenJDK Runtime Environment (build 11.0.10+9-Ubuntu-0ubuntu1.18.04)  
OpenJDK 64-Bit Server VM (build 11.0.10+9-Ubuntu-0ubuntu1.18.04, mixed mode, sharing)  
mauro@2018630981-1:~$
```

Obtenemos el archivo TokenRing.java, el cuál fue previamente subido en un repositorio de github por medio del comando “wget https://raw.githubusercontent.com/Zekyal/Diuse-o-de-Sistemas-Distribuidos/main/TokenRing.java” para realizar su compilación. Cabe destacar que el link utilizado en el comando “wget” nos lleva a la versión raw del archivo que se encuentra en el repositorio.

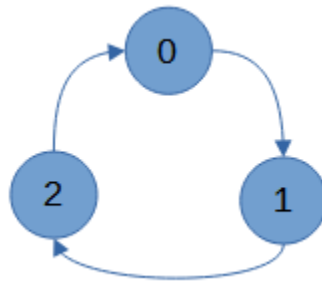
```
mauro@2018630981-0:~$ wget https://raw.githubusercontent.com/Zekyal/Diuse-o-de-Sistemas-Distribuidos/main/TokenRing.java  
--2021-03-24 09:54:56-- https://raw.githubusercontent.com/Zekyal/Diuse-o-de-Sistemas-Distribuidos/main/TokenRing.java  
Resolving raw.githubusercontent.com (raw.githubusercontent.com)... 185.199.110.133, 185.199.108.133, 185.199.109.133, ...  
Connecting to raw.githubusercontent.com (raw.githubusercontent.com)|185.199.110.133|:443... connected.  
HTTP request sent, awaiting response... 200 OK  
Length: 2461 (2.4K) [text/plain]  
Saving to: 'TokenRing.java'  
  
TokenRing.java 100%[=====] 2.40K --.-KB/s in 0s  
2021-03-24 09:54:56 (33.7 MB/s) - 'TokenRing.java' saved [2461/2461]  
mauro@2018630981-0:~$  
  
mauro@2018630981-1:~$ wget https://raw.githubusercontent.com/Zekyal/Diuse-o-de-Sistemas-Distribuidos/main/TokenRing.java  
--2021-03-24 09:55:38-- https://raw.githubusercontent.com/Zekyal/Diuse-o-de-Sistemas-Distribuidos/main/TokenRing.java  
Resolving raw.githubusercontent.com (raw.githubusercontent.com)... 185.199.110.133, 185.199.108.133, 185.199.109.133, ...  
Connecting to raw.githubusercontent.com (raw.githubusercontent.com)|185.199.110.133|:443... connected.  
HTTP request sent, awaiting response... 200 OK  
Length: 2461 (2.4K) [text/plain]  
Saving to: 'TokenRing.java'  
  
TokenRing.java 100%[=====] 2.40K --.-KB/s in 0s  
2021-03-24 09:55:38 (39.8 MB/s) - 'TokenRing.java' saved [2461/2461]  
mauro@2018630981-1:~$  
  
mauro@2018630981-2:~$ wget https://raw.githubusercontent.com/Zekyal/Diuse-o-de-Sistemas-Distribuidos/main/TokenRing.java  
--2021-03-24 09:56:02-- https://raw.githubusercontent.com/Zekyal/Diuse-o-de-Sistemas-Distribuidos/main/TokenRing.java  
Resolving raw.githubusercontent.com (raw.githubusercontent.com)... 185.199.110.133, 185.199.108.133, 185.199.109.133, ...  
Connecting to raw.githubusercontent.com (raw.githubusercontent.com)|185.199.109.133|:443... connected.  
HTTP request sent, awaiting response... 200 OK  
Length: 2461 (2.4K) [text/plain]  
Saving to: 'TokenRing.java'  
  
TokenRing.java 100%[=====] 2.40K --.-KB/s in 0s  
2021-03-24 09:56:02 (47.0 MB/s) - 'TokenRing.java' saved [2461/2461]  
mauro@2018630981-2:~$
```

Una vez cargado el archivo dentro de cada máquina virtual, realizamos la compilación de dicho archivo por medio del comando "javac TokenRing.java".

```
mauro@2018630981-0: ~  
mauro@2018630981-0:~$ javac TokenRing.java  
mauro@2018630981-0:~$ ls  
'TokenRing$Worker.class'  TokenRing.class  TokenRing.java  
mauro@2018630981-0:~$  
mauro@2018630981-1: ~  
mauro@2018630981-1:~$ javac TokenRing.java  
mauro@2018630981-1:~$ ls  
'TokenRing$Worker.class'  TokenRing.class  TokenRing.java  
mauro@2018630981-1:~$  
mauro@2018630981-2: ~  
mauro@2018630981-2:~$ javac TokenRing.java  
mauro@2018630981-2:~$ ls  
'TokenRing$Worker.class'  TokenRing.class  TokenRing.java  
mauro@2018630981-2:~$
```

### Ejecución del programa:

Realizamos la ejecución del programa pasándole 2 argumentos al programa, el primero que indicará el número de nodo y el segundo que indicará la ip de la máquina virtual que se usará como el nodo siguiente, de tal manera que la configuración del token ring quede de la siguiente manera:



El token a enviar será un número entero de 32 bits, el cual será inicializado en el nodo 0 inicializará con el valor de 1. Posteriormente el nodo 0 enviará el token al nodo 1, entonces el nodo 1 recibirá el token y lo enviará al nodo 2. El nodo 2 recibirá el token y lo enviará al nodo 0.

Las direcciones IP de cada nodo se enlistan a continuación:

- Dirección IP del nodo 0: 13.65.112.211
- Dirección IP del nodo 1: 70.37.70.138
- Dirección IP del nodo 2: 23.102.169.75

Se muestra a continuación la ejecución del programa en cada una de las máquinas virtuales:

```
mauro@2018630981-0: ~  
mauro@2018630981-0:~$ java TokenRing 0 70.37.70.138  
mauro@2018630981-1: ~  
mauro@2018630981-1:~$ java TokenRing 1 23.102.169.75  
mauro@2018630981-2: ~  
mauro@2018630981-2:~$ java TokenRing 2 13.65.112.211
```

Finalmente se muestra la salida de los 3 nodos una vez el nodo llega al valor de 1000 en su contador. Como se puede observar, al haber finalizado el nodo 0, y por consiguiente no haber ningún token a enviar para los nodos 1 y 2, estos generan un error.

```
mauro@2018630981-0: ~  
Nodo: 0 Contador: 940 Token: 1  
Nodo: 0 Contador: 941 Token: 1  
Nodo: 0 Contador: 942 Token: 1  
Nodo: 0 Contador: 943 Token: 1  
Nodo: 0 Contador: 944 Token: 1  
Nodo: 0 Contador: 945 Token: 1  
Nodo: 0 Contador: 946 Token: 1  
Nodo: 0 Contador: 947 Token: 1  
Nodo: 0 Contador: 948 Token: 1  
Nodo: 0 Contador: 949 Token: 1  
Nodo: 0 Contador: 950 Token: 1  
Nodo: 0 Contador: 951 Token: 1  
Nodo: 0 Contador: 952 Token: 1  
Nodo: 0 Contador: 953 Token: 1  
Nodo: 0 Contador: 954 Token: 1  
Nodo: 0 Contador: 955 Token: 1  
Nodo: 0 Contador: 956 Token: 1  
Nodo: 0 Contador: 957 Token: 1  
Nodo: 0 Contador: 958 Token: 1  
Nodo: 0 Contador: 959 Token: 1  
Nodo: 0 Contador: 960 Token: 1  
Nodo: 0 Contador: 961 Token: 1  
Nodo: 0 Contador: 962 Token: 1  
Nodo: 0 Contador: 963 Token: 1  
Nodo: 0 Contador: 964 Token: 1  
Nodo: 0 Contador: 965 Token: 1  
Nodo: 0 Contador: 966 Token: 1  
Nodo: 0 Contador: 967 Token: 1  
Nodo: 0 Contador: 968 Token: 1  
Nodo: 0 Contador: 969 Token: 1  
Nodo: 0 Contador: 970 Token: 1  
Nodo: 0 Contador: 971 Token: 1  
Nodo: 0 Contador: 972 Token: 1  
Nodo: 0 Contador: 973 Token: 1  
Nodo: 0 Contador: 974 Token: 1  
Nodo: 0 Contador: 975 Token: 1  
Nodo: 0 Contador: 976 Token: 1  
Nodo: 0 Contador: 977 Token: 1  
Nodo: 0 Contador: 978 Token: 1  
Nodo: 0 Contador: 979 Token: 1  
Nodo: 0 Contador: 980 Token: 1  
Nodo: 0 Contador: 981 Token: 1  
Nodo: 0 Contador: 982 Token: 1  
Nodo: 0 Contador: 983 Token: 1  
Nodo: 0 Contador: 984 Token: 1  
Nodo: 0 Contador: 985 Token: 1  
Nodo: 0 Contador: 986 Token: 1  
Nodo: 0 Contador: 987 Token: 1  
Nodo: 0 Contador: 988 Token: 1  
Nodo: 0 Contador: 989 Token: 1  
Nodo: 0 Contador: 990 Token: 1  
Nodo: 0 Contador: 991 Token: 1  
Nodo: 0 Contador: 992 Token: 1  
Nodo: 0 Contador: 993 Token: 1  
Nodo: 0 Contador: 994 Token: 1  
Nodo: 0 Contador: 995 Token: 1  
Nodo: 0 Contador: 996 Token: 1  
Nodo: 0 Contador: 997 Token: 1  
Nodo: 0 Contador: 998 Token: 1  
Nodo: 0 Contador: 999 Token: 1  
Nodo: 0 Contador: 1000 Token: 1  
mauro@2018630981-0:~$  
mauro@2018630981-1: ~  
Nodo: 1 Contador: 942 Token: 1  
Nodo: 1 Contador: 943 Token: 1  
Nodo: 1 Contador: 944 Token: 1  
Nodo: 1 Contador: 945 Token: 1  
Nodo: 1 Contador: 946 Token: 1  
Nodo: 1 Contador: 947 Token: 1  
Nodo: 1 Contador: 948 Token: 1  
Nodo: 1 Contador: 949 Token: 1  
Nodo: 1 Contador: 950 Token: 1  
Nodo: 1 Contador: 951 Token: 1  
Nodo: 1 Contador: 952 Token: 1  
Nodo: 1 Contador: 953 Token: 1  
Nodo: 1 Contador: 954 Token: 1  
Nodo: 1 Contador: 955 Token: 1  
Nodo: 1 Contador: 956 Token: 1  
Nodo: 1 Contador: 957 Token: 1  
Nodo: 1 Contador: 958 Token: 1  
Nodo: 1 Contador: 959 Token: 1  
Nodo: 1 Contador: 960 Token: 1  
Nodo: 1 Contador: 961 Token: 1  
Nodo: 1 Contador: 962 Token: 1  
Nodo: 1 Contador: 963 Token: 1  
Nodo: 1 Contador: 964 Token: 1  
Nodo: 1 Contador: 965 Token: 1  
Nodo: 1 Contador: 966 Token: 1  
Nodo: 1 Contador: 967 Token: 1  
Nodo: 1 Contador: 968 Token: 1  
Nodo: 1 Contador: 969 Token: 1  
Nodo: 1 Contador: 970 Token: 1  
Nodo: 1 Contador: 971 Token: 1  
Nodo: 1 Contador: 972 Token: 1  
Nodo: 1 Contador: 973 Token: 1  
Nodo: 1 Contador: 974 Token: 1  
Nodo: 1 Contador: 975 Token: 1  
Nodo: 1 Contador: 976 Token: 1  
Nodo: 1 Contador: 977 Token: 1  
Nodo: 1 Contador: 978 Token: 1  
Nodo: 1 Contador: 979 Token: 1  
Nodo: 1 Contador: 980 Token: 1  
Nodo: 1 Contador: 981 Token: 1  
Nodo: 1 Contador: 982 Token: 1  
Nodo: 1 Contador: 983 Token: 1  
Nodo: 1 Contador: 984 Token: 1  
Nodo: 1 Contador: 985 Token: 1  
Nodo: 1 Contador: 986 Token: 1  
Nodo: 1 Contador: 987 Token: 1  
Nodo: 1 Contador: 988 Token: 1  
Nodo: 1 Contador: 989 Token: 1  
Nodo: 1 Contador: 990 Token: 1  
Nodo: 1 Contador: 991 Token: 1  
Nodo: 1 Contador: 992 Token: 1  
Nodo: 1 Contador: 993 Token: 1  
Nodo: 1 Contador: 994 Token: 1  
Nodo: 1 Contador: 995 Token: 1  
Nodo: 1 Contador: 996 Token: 1  
Nodo: 1 Contador: 997 Token: 1  
Nodo: 1 Contador: 998 Token: 1  
Nodo: 1 Contador: 999 Token: 1  
Nodo: 1 Contador: 1000 Token: 1  
Exception in thread "main" java.io.EOFException  
at java.base/java.io.DataInputStream.readInt(DataInputStream.java:397)  
at TokenRing.main(TokenRing.java:74)  
mauro@2018630981-1:~$  
mauro@2018630981-2: ~  
Nodo: 2 Contador: 942 Token: 1  
Nodo: 2 Contador: 943 Token: 1  
Nodo: 2 Contador: 944 Token: 1  
Nodo: 2 Contador: 945 Token: 1  
Nodo: 2 Contador: 946 Token: 1  
Nodo: 2 Contador: 947 Token: 1  
Nodo: 2 Contador: 948 Token: 1  
Nodo: 2 Contador: 949 Token: 1  
Nodo: 2 Contador: 950 Token: 1  
Nodo: 2 Contador: 951 Token: 1  
Nodo: 2 Contador: 952 Token: 1  
Nodo: 2 Contador: 953 Token: 1  
Nodo: 2 Contador: 954 Token: 1  
Nodo: 2 Contador: 955 Token: 1  
Nodo: 2 Contador: 956 Token: 1  
Nodo: 2 Contador: 957 Token: 1  
Nodo: 2 Contador: 958 Token: 1  
Nodo: 2 Contador: 959 Token: 1  
Nodo: 2 Contador: 960 Token: 1  
Nodo: 2 Contador: 961 Token: 1  
Nodo: 2 Contador: 962 Token: 1  
Nodo: 2 Contador: 963 Token: 1  
Nodo: 2 Contador: 964 Token: 1  
Nodo: 2 Contador: 965 Token: 1  
Nodo: 2 Contador: 966 Token: 1  
Nodo: 2 Contador: 967 Token: 1  
Nodo: 2 Contador: 968 Token: 1  
Nodo: 2 Contador: 969 Token: 1  
Nodo: 2 Contador: 970 Token: 1  
Nodo: 2 Contador: 971 Token: 1  
Nodo: 2 Contador: 972 Token: 1  
Nodo: 2 Contador: 973 Token: 1  
Nodo: 2 Contador: 974 Token: 1  
Nodo: 2 Contador: 975 Token: 1  
Nodo: 2 Contador: 976 Token: 1  
Nodo: 2 Contador: 977 Token: 1  
Nodo: 2 Contador: 978 Token: 1  
Nodo: 2 Contador: 979 Token: 1  
Nodo: 2 Contador: 980 Token: 1  
Nodo: 2 Contador: 981 Token: 1  
Nodo: 2 Contador: 982 Token: 1  
Nodo: 2 Contador: 983 Token: 1  
Nodo: 2 Contador: 984 Token: 1  
Nodo: 2 Contador: 985 Token: 1  
Nodo: 2 Contador: 986 Token: 1  
Nodo: 2 Contador: 987 Token: 1  
Nodo: 2 Contador: 988 Token: 1  
Nodo: 2 Contador: 989 Token: 1  
Nodo: 2 Contador: 990 Token: 1  
Nodo: 2 Contador: 991 Token: 1  
Nodo: 2 Contador: 992 Token: 1  
Nodo: 2 Contador: 993 Token: 1  
Nodo: 2 Contador: 994 Token: 1  
Nodo: 2 Contador: 995 Token: 1  
Nodo: 2 Contador: 996 Token: 1  
Nodo: 2 Contador: 997 Token: 1  
Nodo: 2 Contador: 998 Token: 1  
Nodo: 2 Contador: 999 Token: 1  
Nodo: 2 Contador: 1000 Token: 1  
Exception in thread "main" java.io.EOFException  
at java.base/java.io.DataInputStream.readInt(DataInputStream.java:397)  
at TokenRing.main(TokenRing.java:74)  
mauro@2018630981-2:~$
```

Finalmente, cabe mencionar que el programa no realizara ninguna acción si es que los nodos no están conectados entre sí.

## Código Fuente del programa:

```
import java.io.DataInputStream;  
import java.io.DataOutputStream;  
import java.net.ServerSocket;  
import java.net.Socket;  
  
public class TokenRing {  
    static DataInputStream entrada;  
    static DataOutputStream salida;  
    static boolean primera_vez = true;  
    static String ip;  
    static int nodo;  
    static int token;  
}
```

```

static int contador = 0;

static class Worker extends Thread
{
    public void run()
    {
        //Algoritmo 1
        try{
            ServerSocket servidor = new ServerSocket(50000);
            Socket conexion = servidor.accept();
            entrada = new DataInputStream(conexion.getInputStream());
        }catch(Exception e){
            e.printStackTrace();
        }
    }
}

public static void main(String[] args) throws Exception
{
    if (args.length != 2)
    {
        System.err.println("Se debe pasar como parametros el numero
del nodo y la IP del siguiente nodo");
        System.exit(1);
    }

    nodo = Integer.valueOf(args[0]); // el primer parametro es el
numero de nodo
    ip = args[1]; // el segundo parametro es la IP del siguiente
nodo en el anillo

    //Algoritmo 2
    Worker w = new Worker();
    w.start();
    Socket conexion = null;

    for(;;)
    try
    {
        conexion = new Socket(ip,50000);
        break;
    }
    catch (Exception e)
    {
        Thread.sleep(100);
    }

    salida = new DataOutputStream(conexion.getOutputStream());
    w.join();

    for(;;){
        if(nodo==0){
            if(primeravez==true){
                primera_vez = false;
                token = 1;
            }else{
                token = entrada.readInt();
            }
        }
    }
}

```



```

        contador++;
        System.out.println("Nodo: "+nodo+"  Contador:
"+contador+"  Token: "+token);

        if(contador==1000)
            break;
    }
    }else{
        token = entrada.readInt();
        contador++;
        System.out.println("Nodo: "+nodo+"  Contador:
"+contador+"  Token: "+token);
    }

    salida.writeInt(token);
}
}
}

```

### **Conclusión:**

El uso del algoritmo de token en anillo o token ring resulta bastante efectivo para llevar a cabo la exclusión mutua, pues además de que su implementación resulta ser bastante sencilla, es bastante rápida y ocupa pocos recursos siempre y cuando no se tenga un gran número de nodos y la conexión entre estos sea estable.