24/02/2005

Ejercicio 1.- (3 puntos) Analice los siguientes apartados y justifique en su caso las respuestas con ejemplos.

a) Si tenemos una aplicación donde los hilos se bloquean a menudo. ¿Es mejor implantarla a nivel de usuario o a nivel de núcleo?

b) La programación en grupo que ventajas e inconvenientes tiene. Cite alguna primitiva de programación en grupo en pvm para la sincronización de procesos.

c) Defina lo que es un puerto sobre una computadora en la web. Indique varios protocolos y sus números de puerto bien-conocidos, así como su misión y su tipo de codificación.

d) Ventajas e inconvenientes de los algoritmos de elección de coordinador en un SD.

e) Una cerradura giratoria (*spin lock)* es una variable Booleana cuyo acceso se realiza mediante una instrucción atómica *test-and-set* que se utiliza para resolver un problema de exclusión mutua. ¿Se podría utilizar un *spin lock* para obtener exclusión mutua entre hilos en una computadora con un solo procesador?

f) En una comunicación basada en petición-respuesta, el protocolo HacerOperación-ObtenerPetición-EnviarRespuesta el mensaje de respuesta actúa como reconocimiento mientras que en el protocolo Enviar-Recibir cada mensaje se reconoce de forma individual. Cuales son las ventajas de una sobre la otra

Ejercicio 2.- Analice el siguiente código y responda a la pregunta, justificando las respuestas.

¿Qué misión tienen los hilos que aparecen en el mismo?

```
//: c14:Daemons.java
// Daemonic behavior.
import java.io.*;
```

```
class Daemon extends Thread {
  private static final int SIZE = 10;
  private Thread[] t = new Thread[SIZE];
  public Daemon() {
    setDaemon(true);
    start();
  }
  public void run() {
    for(int i = 0; i < SIZE; i++)
      t[i] = new DaemonSpawn(i);
    for(int i = 0; i < SIZE; i++)
      System.out.println(
        "t[" + i + "].isDaemon() = "
        + t[i].isDaemon());
    while(true)
      yield();
  }
}
```

Ejercicio 3.- Analice el siguiente código, indicando su misión y el modelo de comunicación que representa en un SD.

a) Que misión tienen las clases de Java ServerSocket y DatagramSocket. ¿Cuales son sus diferencias?

```
Import  java.net.*;
import  java.io.*;
import  java.util.Date;

public class UDPDaytime Server extend UDPServer {
        protected static int defaultPort = 13;

        public static void respond(DatagramSocket ds, DatagramPacket dp) {

                DatagramPacket outgoing;

                Date now = new Date();
                String s= now.toString();
                byte[ ] data = new byte[s.length()];
                s.getBytes(0, s.length(), data, 0);
                try {
                        outgoing = new DatagramPacket(data, data.length, dp.getAddress( ), dp.getPort( ));
                ds.send(outgoing);
                }
                catch (IOException e) {
                        System.err.println(e);
                }
            }
        }
```