

**UNIVERSITY OF LONDON**

**GOLDSMITHS COLLEGE**

**B. Sc. Examination 2005**

**COMPUTING AND INFORMATION SYSTEMS**

**IS53022A (CIS337)**

**Internet Programming in Java**

**Duration: 2 1/4 hours**

**Date and time:**

---

*Full marks will be awarded for complete answers to a total of three questions. Each question carries 25 marks. The marks for each part of a question are indicated at the end of the part in [.] brackets.*

*There are 75 marks available on this paper.*

*Electronic calculators must not be programmed prior to the examination. Calculators which display graphics, text or algebraic equations are not allowed.*

*Your attention is drawn to the list of method prototypes in the Appendix at the end of this paper.*

## Question 1

(a) Java does not have an unsigned byte data type. Write a code fragment that converts signed bytes to unsigned bytes.

[1 mark]

(b) Write a method with the prototype

```
public static String toDottedQuad( byte[] address )
```

which takes an array of four signed bytes and returns a dotted quad IP address in the form "12.34.56.78".

[4 marks]

(c) Write a method with the prototype

```
public static String getUserInput()
```

which reads in a line from the standard input.

[5 marks]

(d) Now write a Java program that emulates the unix utility **nslookup**. You should make full use of the methods from parts (b)-(d) and you do not need to write them out again. This utility has two modes: interactive and command line. In command line mode, **nslookup** will print the host names or IP addresses of IP addresses or host names entered at the command line. The utility enters interactive mode if no host name or IP address is given; it will then continue to print the corresponding IP address and host name to host names and IP addresses that have been entered at the standard input. An example of three sessions with **nslookup** is given below. [15 marks]

```
>java nslookup www.gold.ac.uk  
158.223.1.1
```

```
>java nslookup 158.223.1.1  
scorpio.gold.ac.uk
```

```
>java nslookup  
Enter names and IP addresses. Enter exit to quit.  
scorpio.gold.ac.uk  
158.223.1.1  
127.0.0.1  
localhost  
exit
```

## Question 2

(a) How does Java help programmers handle the complexity of internet data transmission?

[2 marks]

(b) What basic functionality is offered by the `java.net.Socket` class.

[4 marks]

(c) RFC862 specifies an echo protocol. What is this specification for TCP?

[3 marks]

(d) Suppose that a socket object has instantiated using the statement

```
Socket socket = new Socket(hostname, port);
```

Show how a `BufferedReader` object can be constructed from the socket's `InputStream`

[2 marks]

(e) Show how a `PrintWriter` object can be constructed from the socket's `OutputStream`.

[2 marks]

(f) Write, using your answers to parts (d), (e) and (f), an echo client. This client should send data typed at the command line to an echo server, and should display on the terminal any data sent back from the server.

[12 marks]

### Question 3

- (a) Explain how a non-threaded server might handle incoming requests. [4 marks]
- (b) Explain how a threaded server (for example, written in Java) might handle incoming requests. [4 marks]
- (c) Discuss the relative merits of threaded and non-threaded servers. [5 marks]
- (d) Explain how threads are created and manipulated in Java. You should use code samples to illustrate your answer. [12 marks]

### Question 4

- (a) What is the primary function of the java class **ServerSocket**? [2 marks]
- (b) Can **ServerSocket** be used for data reception and transmission? [2 marks]
- (c) Describe the lifecycle of a java-implemented server. [7 marks]
- (d) RFC867 specifies a daytime protocol. What is this specification for TCP? [4 marks]
- (e) Write java code for a daytime server. [10 marks]

### Question 5

- (a) Someone in your development team has handed you a Java program which has not

been commented. Your job is to work out what the program does and include comments at appropriate places. Comment 1 should be a brief statement on the functionality of the program; the remaining comments refer to the following line of code. The source code for this program, `Unknown.java` is printed on the next page.

[19 marks]

(b) Suggest an improvement that you would make to this code.

[6 marks]

```

// comment 1 [6 marks]
public class Unknown {

    // comment 2 [1 mark]
    public final static int DEFAULT_PORT = 9;

    // comment 3 [1 mark]
    public final static int MAX_PACKET_SIZE = 65507;

    public static void main(String[] args) {

        // comment 4 [1 mark]
        int port = DEFAULT_PORT;
        byte[] buffer = new byte[MAX_PACKET_SIZE];

        // comment 5 [1 mark]
        if ( args.length > 0 )
            port = Integer.parseInt ( args[0] );

        // comment 6 [1 mark]
        DatagramSocket server =
            new DatagramSocket(port);

        // comment 7 [1 mark]
        DatagramPacket packet =
            new DatagramPacket(buffer, buffer.length);

        // comment 8 [1 mark]
        while (true) {

            // comment 9 [2 marks]
            server.receive(packet);

            // comment 10 [2 marks]
            String s = new String(
                packet.getData(), 0,
                packet.getLength() );

            // comment 11 [2 marks]
            System.out.println( packet.getAddress() +
                " at port " + packet.getPort()+ " says " + s);
        }
    }
}

```

## APPENDIX

### java.net.InetAddress

```
public String getHostName()
public byte[] getAddress()
public String getHostAddress()
public static InetAddress getByName(String host)
    throws UnknownHostException
public static InetAddress[] getAllByName(String host)
    throws UnknownHostException
public static InetAddress getByAddress(byte[] addr)
    throws UnknownHostException
public static InetAddress getLocalHost()
    throws UnknownHostException
```

### java.net.Socket

```
public Socket(String host, int port)
    throws UnknownHostException, IOException
public Socket(InetAddress address, int port) throws IOException
public InetAddress getInetAddress()
public InetAddress getLocalAddress()
public int getPort()
public int getLocalPort()
public SocketAddress getRemoteSocketAddress()
public SocketAddress getLocalSocketAddress()
public InputStream getInputStream() throws IOException
public OutputStream getOutputStream() throws IOException
public void close() throws IOException
public boolean isConnected()
public boolean isClosed()
```

### java.net.ServerSocket

```
public ServerSocket() throws IOException
public ServerSocket(int port) throws IOException
public InetAddress getInetAddress()
public int getLocalPort()
public SocketAddress getLocalSocketAddress()
public Socket accept() throws IOException
public void close() throws IOException
public boolean isBound()
public boolean isClosed()
```