

UNIVERSITY OF LONDON

GOLDSMITHS COLLEGE

Department of Computing

B. Sc. Examination 2014

IS52025A

Internet and Distributed Programming

Duration: 2 hours 15 minutes

Date and time:

There are five questions in this paper. You should answer no more than THREE questions. 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.

**THIS PAPER MUST NOT BE REMOVED
FROM THE EXAMINATION ROOM**

Question 1

(a) Briefly explain what the following program does:

```
public class seb5
{
    public static void main(String[] args) throws Exception
    {
        Class.forName("com.mysql.jdbc.Driver");
        Connection c=
        DriverManager.getConnection("jdbc:mysql://localhost/bla","mas01sd","sebastian");
        Statement st = c.createStatement();
        st.executeUpdate("INSERT INTO one VALUES('" + args[0] + "','" + args[1] + "')");
        ResultSet resultSet = st.executeQuery("SELECT * from one");
        while (resultSet.next())
        {
            for (int i=1;i<3;i++)System.out.print(resultSet.getString(i) + " ");
            System.out.println();
        }
    }
}
```

[10]

(b) Given the class Pair

```
class Pair
{
    String first;
    String second;
    Pair (String f,String s)
    {
        first=f;second=s;
    }
}
```

Write a method, `makeSetFromTable` which takes a `ResultSet` resulting from a query like

```
ResultSet resultSet = st.executeQuery("SELECT * from one");
```

and returns a `HashSet` of Pairs, each pair corresponding to a row of the table (which we assume has two String fields).

[8]

(c) Write a function `makeTableFromSet` which takes a `HashSet` of Pairs and inserts each pair one at a time into a table.

[7]

Question 2

(a) Consider the following programs *s2* and *c2*:

```
import java.io.*;
import java.net.*;
class s2
{
    public static void main(String[] argv) throws Exception
    {
        ServerSocket s = new ServerSocket(5000);
        Socket t = s.accept();
        System.out.println("hello");
    }
}
```

```
import java.io.*;
import java.net.*;
class c2
{
    public static void main(String[] argv) throws Exception
    {
        Socket s = new Socket("localhost",5000);
        System.out.println("goodbye");
    }
}
```

- i. If we just run *c2* and not *s2*, which of the following will happen? (choose only one)
1. An exception will be thrown
 2. Only *hello* will be printed.
 3. Only *goodbye* will be printed.
 4. *hello* and *goodbye* will both be printed.
 5. none of the above.

ii. If we just run *s2* and not *c2*, which of the following will happen? (choose only one)

1. An exception will be thrown
2. Only *hello* will be printed.
3. Only *goodbye* will be printed.
4. *hello* and *goodbye* will both be printed.
5. none of the above.

iii. If we just run *c2* and *s2* on the same machine, which of the following will happen? (choose only one)

1. An exception will be thrown
2. Only *hello* will be printed.
3. Only *goodbye* will be printed.
4. *hello* and *goodbye* will both be printed.
5. none of the above.

iv. If we just run *c2* and *s2* on different machines, which of the following will happen? (choose only one)

1. An exception will be thrown
2. Only *hello* will be printed.
3. Only *goodbye* will be printed.
4. *hello* and *goodbye* will both be printed.
5. none of the above.

[12]

(b) Write a complete single threaded server that listens on port 8011 for characters, converts them to upper case and sends them back to the client.

[13]

Question 3

(a) Consider the following Java code, with 4 missing fragments:

```
import java.io.*;
import java.net.*;

class client
{
    public static void main(String[] argv) throws Exception
    {
        Socket s = /*missing 1*/;
        OutputStreamWriter toServer = /*missing 2*/;
        InputStream keyboard = /*missing 3 */;
        InputStream fromServer = /* missing 4 */;
        int c;
        while(true)
        {
            /* missing 5*/

        }
    }
}
```

Complete the missing fragments so that the above program acts as a client which sends data one character at a time to a server running on the localhost listening at port 8000. It sends whatever is typed on the console one character at a time to the server and prints out on the console whatever characters it receives back from the server.

[16]

(b) Consider the following Java program:

```
class p
{
    void f()
    { while (true) System.out.println("hello");}

    void g()
    { while (true) System.out.println("goodybye");}
}

class t1 extends Thread
{ p x;
  t1(p y)
  {x=y;}

  public void run()
  {x.g();}
}

class t2 extends Thread
{ p x;
  t2(p y)
  {x=y;}

  public void run()
  {x.f();}
}

class z
{
    public static void main(String[] argv)
    {
        p it= new p();
        new t2(it).start();
        new t1(it).start();
    }
}
```

- i. Explain what is output when it is executed.
- ii. What would happen if we declared the methods `f()` and `g()` as synchronized?
- iii. What common problem in concurrent programming is this an example of?

[9]

Question 4

(a) Briefly describe the purpose of the following method:

```
static HashSet<String> links (String url)
{
    HashSet<String> a= new HashSet<String>();
    try{org.jsoup.Connection z=Jsoup.connect(url);
        Document doc = z.get();
        Elements links = doc.select("a[href]");
        for (Element link : links) a.add(link.attr("abs:href"));

    }
    catch (Exception e)
    {
        System.out.println(e);
    }
    return a;
}
```

[10]

(b) Using the above method, write a method

```
static HashSet<String> links (String url, int n)
which finds all the links from url at all depths from 1 to n inclusive.
```

[15]

Question 5

- (a) Given the following class definition:

```
import java.io.*;
public class Person implements Serializable
{
    String name;
    int age;

    public Person (String n, int a)
    {
        age=a;name=n;
    }

    public String toString()
    {
        return name+" "+age;
    }
}
```

Write a complete single-threaded client that repeatedly reads names and ages from the console, constructs Person objects from them, and sends these Person objects to a server listening on port 5000 on "localhost". It doesn't listen out for messages from the server.

[10]

- (b) Write a complete single-threaded 'Object' server that listens on port 5000 for Objects and prints them out on the console if they are Person Objects. It doesn't send messages back to its client.

[15]