

UNIVERSITY OF LONDON

GOLDSMITHS COLLEGE

B. Sc. Examination 2012

DEPARTMENT OF COMPUTING

IS52028B Principles and Applications of Programming

IS52014C Programming with Object Orientated Data

Structures and Algorithms

(Music Computing Only.)

Duration: 1 hour 30 minutes, 11 May 14.30

---

*There are two questions in this paper. You should answer both 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 50 marks available on this paper.*

*No calculators should be used.*

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

## Question 1

- (a) i. A prime number is a positive integer, larger than 1, that can only be divided, without remainder, by itself and 1. Provide an English language procedure for testing if a given integer  $p > 1$  is prime. [2]
- ii. Draw a flow chart for the procedure described in part a.(i). [3]
- iii. Write a *procedural* Java program `Prime` which implements the procedure of part a.(ii). Your program should test program arguments for primality e.g if your program is invoked with the command line instruction `java Prime 12 97 2`, your program responds with

```
12 is not a prime
97 is prime
2 is prime
```

You may wish to use the following information taken from the Java api:

```
class java.lang.Integer
```

```
public static int parseInt(String s)
```

```
    Parses the string argument as a signed decimal integer.
```

[10]

- (b) i. What is the string pool and how does it differ from the garbage collectable heap? Your answer should consider advantages and disadvantages. [5]
- ii. Consider the class `StringTest` below.

```
package exam;
```

```
public class StringTest {

    public static void main(String [] args) {

        String s = "42";
        String t = "42";
        String u = new String("42");

        System.out.println(s == t);
        System.out.println(s.equals(t));
        System.out.println(s == u);
        System.out.println(s.equals(u));
    }
}
```

Its output is:

```
true  
true  
false  
true
```

Explain these results.

[5]

## Question 2

- (a) i. Write a class `Broadcast` with a single method `public void userIn()` which, using Java input streams, endlessly reads in strings typed at the command line. [4]
- ii. Write an interface `Listener` with a single abstract method `public void inform(String s)`. [2]
- iii. Modify your `Broadcast` class so that it includes a method `public void register(Listener l)` which enables `Listener`'s to register their interest. Also, adapt the method you wrote in part a.(i) so that messages typed at the command line are distributed to all registered listeners. [6]
- (b) i. The JVM is able to run more than one call stack in order to give the appearance of multi-tasking. Explain what a call stack is and how the JVM accomplishes multi-tasking. Illustrate your answer with code snippets and with appropriate diagrams. [8]
- ii. The multi-tasking described in part b.(i) can create a problem known as deadlock. Explain what deadlock is, and provide an example of how it might occur. [5]