PRINCE OF SONGKLA UNIVERSITY

FACULTY OF ENGINEERING

Department of Computer Engineering

Midterm Examination: Semester 1 **Academic Year**: 2002-2003

Date: 4th August, 2002 **Time**: 13.30 – **15.30** (**2 hours**)

Subject Number: 240-424 **Room**: R201

Subject Title: Introduction to Java Programming

Lecturer: Aj. Andrew Davison

Exam Duration: 2 hours

This paper has 2 pages.

Authorized Materials:

- Writing instruments (e.g. pens, pencils).
- Books (e.g. dictionaries) and calculators are **not** permitted.

Instructions to Students:

- *Answer questions in English.* Perfect English is **not** required.
- Attempt all questions.
- Write your answers in an answer book.
- Start your answer to each question on a new page
- Clearly number your answers.
- Any unreadable parts will be considered wrong.
- When writing programs, use good layout, and short comments; marks will not be deducted for minor syntax errors.
- The marks for each part of a question are given in brackets (…).

Question 1 (45 marks; 45 minutes)

a) What are the differences between a class and an object? (15)

b) Explain *extends* in Java. (15)

c) What is a *polymorphic* data structure? (15)

Each answer should include diagrams and **small** code fragments where possible.

Question 2 (30 marks; 30 minutes)

a) Write a Java **applet** that starts by reading in a string from an input dialog box. The string should contain two integers separated by spaces (e.g. "23 5"). The applet should tokenise the string, extract the integers, add them, and display the result using its `paint()` method. (25)

b) What are the main *differences* between the `String` and `StringBuffer` classes? Your answer should include diagrams and **small** code fragments where possible. (5)

Question 3 (45 marks; 45 minutes)

a) Explain in words the *six* steps for creating a `JFrame` GUI in Java.
Note: do **not** include any Java code. (5)

b) Write a Java **application** which contains a single button labelled "Time me". When pressed, the button should print "<X> millisecs since the last time" to the DOS Window. <X> is the number of milliseconds since the last time the button was pressed.

However, when the button is *first* pressed, it should only print "Hello Tom Cruise" to the DOS Window.

A useful Java method is:

```
long System.currentTimeMillis();
```

which returns the current time in milliseconds as a long integer.

Comment your code to show where the six steps of part (a) appear. (35)

c) Draw an event model diagram for your program of part (b). (5)

--- End of Examination ---