INSTITUTE OF TECHNOLOGY BLANCHARDSTOWN

Year	Year 2
Semester	Repeat Paper
Date of Examination	Monday 27th August 2012
Time of Examination	10am – 12pm

Programme Title	HIGHER CERTIFICATE IN SCIENCE IN COMPUTING IN INFORMATION TECHNOLOGY
Programme Code	BN002
Programme Title	BACHELOR OF SCIENCE IN COMPUTING IN INFORMATION TECHNOLOGY
Programme Code	BN013
Programme Title	BACHELOR OF SCIENCE (HONOURS) IN COMPUTING
Programme Code	BN104
Module Title	Advanced Programming
Banner Module Code	COMP H2030

Internal Examiner(s): *Dr. Luke Raeside*

External Examiner(s): *Dr. Richard Studdert*
Mr. Michael Barrett

Instructions to candidates:

1) **To ensure that you take the correct examination, please check that the module and programme which you are following is listed in the tables above.**
2) **Answer any FOUR questions.**
3) **All questions carry equal marks (25 marks).**

DO NOT TURN OVER THIS PAGE UNTIL YOU ARE TOLD TO DO SO

Question 1

a) Briefly outline THREE advantages of using an **IDE** (Integrated Development Environment) in software development. List ONE **IDE** used in industry.

[7 marks]

b) Define the function of the keyword **static** in Java.

[3 marks]

c) Explain the function of **Javadoc** tags. List AND define the function of any TWO **Javadoc** tags commonly used with Java methods.

[6 marks]

d) Explain the effect of applying the following access modifiers to a class feature:

i. **protected**
ii. **default** (also known as **package** access)
iii. **private**

[9 marks]

[Total 25 marks]

Question 2

a) List THREE media control methods found in the **AudioClip** interface. Outline the function of any TWO of the methods listed.

[5 marks]

b) Write a sample Java conditional statement to demonstrate how the **instanceof** comparator is used.

[5 marks]

c) Define the term **inner class** in the context of Java programming.

[3 marks]

d) Explain the term **anonymous inner class** in the context of Java programming. Write a Java code example to demonstrate the definition of an **anonymous inner class**.

[9 marks]

e) Briefly explain the concept of **thread** programming.

[3 marks]

[Total 25 marks]

Question 3

a) Define **class reflection**. Outline the function of TWO reflective methods available in the class **Class**.

[6 marks]

b) Write a Java class definition called **School** that models a school with attributes location, educationLevel (e.g., primary or secondary) and principalName. Provide an appropriate non-blank **constructor** for **School** objects. Write a Java code statement that demonstrates the creation of an object of type **School** using arbitrary values.

[10 marks]

c) Define the function of a **default constructor** in object-oriented programming. Demonstrate how to code a **custom default constructor** by using appropriate Java code statements (use the example from part (b) above if necessary).

[7 marks]

d) List the features that define a **method signature.**

[2 marks]

[Total 25 marks]

Question 4

a) Write a sample **try – catch – finally** block using Java code. Describe briefly the function of the **finally** block in Java.

[6 marks]

b) Explain the difference between **handling** and **declaring** an exception in Java. Use Java code examples to aid your explanation.

[8 marks]

c) Describe briefly what is meant by an **unchecked** exception and a **checked** exception in Java. List and outline the function of ONE commonly encountered **unchecked** exception in Java.

[6 marks]

d) Discuss briefly the role that **inheritance** plays in the application of **polymorphism** in **object oriented programming**. Use an intuitive example to enhance your response.

[5 marks]

[Total 25 marks]

Question 5

a) Discuss THREE Java language features that support **internationalization**. Ensure that you explain precisely the function EACH of the features play in making Java an internationalized programming language.

[12 marks]

b) List TWO **locale sensitive** classes in Java. Describe the function and usage of EACH of the locale sensitive classes listed.

[8 marks]

c) Outline the function of the **ResourceBundle** class in Java. Explain clearly the function of the **getBundle()** method found in the **ResourceBundle** class.

[5 marks]

[Total 25 marks]