

**KENYA METHODIST UNIVERSITY**

**END OF TRIMESTER EXAMINATION APRIL 2008**

**FACULTY : SCIENCE AND SOCIAL STUDIES**  
**DEPARTMENT : COMPUTER INFORMATION SCIENCE**  
**COURSE CODE : COMP 211**  
**COURSE TITLE : OBJECT ORIENTED PROGRAMMING**

---

**Total Marks (60)**

**TIME: 2 ½ HOURS**

**Instructions**

*Answer ALL questions in section A, and ANY ONE question in section B.*

**SECTION A – Answer all questions (30 marks)**

1. Define the following terms
  - a. Class
  - b. Method overriding
  - c. Exception. (2 marks)
2. Differentiate between an argument and a parameter. (2 marks)
3. Give two benefits of handling exceptions yourself. (2 marks)
4. What is the importance of method overriding? (2 marks)
5. Give two uses of the keyword **super** in inheritance. (2 marks)
6. How does polymorphism promote extensibility? (2 marks)
7. Discuss the relative merits of using **protected** access vs. **private** access in superclasses. (2 marks)
8. Describe the two ways in which JAVA creates a **Thread**. (4 marks)
9. Write three Java statements that add a variable x to 1. (3 marks)
10. Describe the following JAVA buzzwords:
  - a. Robust
  - b. Architecture-neutral (4 marks)
11. Write a recursive program that computes the factorial of numbers 1-8. (5 marks)

**SECTION B – Answer ANY ONE question**

**Question 1 – 30 marks**

1. Outline the basics for a typical Java environment. (10 marks)
2. Computers are playing an increasing role in education. Write a program that will help an elementary school student learn multiplication. Use **Math.random** to produce two positive one-digit integers. The program should then display a question, such as

### How much is 6 times 7?

The student then types the answer. Next the program checks the student's answer. If it is correct, display "**Very good!**" and ask another multiplication question. If the answer is wrong, display "**No, please try again**", and lets the student try the same question repeatedly until the student finally gets it right. A separate method should be used to generate each new question. This method should be called once the application begins execution and each time the user answers the question correctly.

(20 marks)

### Question 2 – 30 marks

1. Give two uses of the keyword **this** in Java. (4 marks)
2. Name and explain the three object oriented principles. (6 marks)
3. Write a program that prints a table of the binary and octal equivalents of the decimal numbers 10-20. (10 marks)
4. A palindrome is a number or a text phrase that reads the same backwards as forward. For example, each of the following five-digit integers are palindromes: 12321, 55555, 45554 and 11611. Write an application that reads in a five-digit integer and determines whether or not it is a palindrome. If the number is not five digits long, display an error message indicating the problem to the user and then allows the user to enter a new value. (10 marks)

### Question 3 – 30 marks

1. (*Airline Reservation System*) A small airline has just purchased a computer for its new automated reservations system. You have been asked to program the new system. You are to write an application to assign seats on each flight of the airline's only plane (capacity: 10 seats). Your program should display the following alternatives:

**Please type 1 for "smoking"**

**Please type 2 for "nonsmoking"**

If the person types 1, your program should assign a seat in the smoking section (seats 1-5). If the person types 2, your program should assign a seat in the nonsmoking section (seats 6-10). Your program should then print a boarding pass indicating the person's seat number and whether it is in the smoking or nonsmoking section of the plane.

Use a single-subscripted array of primitive type **boolean** to represent the seating chart of the plane. Initialize all elements of the array to **false** to indicate that all seats are empty. As each seat is assigned, set the corresponding elements of the array to **true** to indicate that the seat is no longer available.

Your program should, of course, never assign a seat that has already been assigned. When the smoking section is full, your program should ask the person if it is acceptable to be placed in the nonsmoking section (and vice versa). If yes, make the appropriate seat assignment. If no, print the message "**Next flight leaves in 3 hours**". (30 marks)