UNIVERSITY OF LAGOS
FACULTY OF ENGINEERING
DEPARTMENT OF ELECTRICAL AND ELECTRONICS ENGINEERING
B.Sc.(Hon.) DEGREE EXAMINATIONS
EEG 311: COMPUTER PROGRAMMING LANGUAGES II
FIRST SEMESTER, 2010/2011 ACADEMIC SESSION

Instructions: 1. Answer ALL questions
2. Build all your programs in a multifile organizational scheme
Time Allowed: 60 minutes

Question One

Write a program that has a function called WriteHellos and uses a default input parameter list in the prototype. The default parameters include an integer for the number of hellos on the line and an integer for the number of lines. The default function result shows five hellos on three lines. This call:

```
WriteHellos();
```

results in output like this:

```
hello hello hello hello hello
hello hello hello hello hello
hello hello hello hello hello
```

If either input value is zero, write "No Hellos for You." Your main function should call this program four times-once using the default values, once so that you see twelve hellos on three lines, once so that you see five hellos on eleven lines, and once showing the "No Hellos for You" message.

Question Two

Develop a class Date for representing a calendar. The class should provide a default constructor that initializes the date to September 14, 1752. Another constructor should initialize a Date object to a specific value using three integer parameters corresponding to the desired month, day, and year. The class should have three public inspectors and three public mutators that allow the month, day, and year to be accessed. The operators ++ and – should be overloaded so that when applied to a Date object, the object's new value is, respectively, the successive or preceding day. The subtraction operator should be overloaded such that the difference of two dates is the number of days between them. Also, overload the insertion and extraction operators for Date objects. Be sure that whatever form of insertion is produced, the result is of the correct form for an extraction to process. Also define auxiliary functions ToString(), which

returns a string version of its Date parameter, and DayOfWeek(), which returns the day of the week on which its Date parameter falls. The return type for DayOfWeek() should be an enumerated type whose symbolic constants are Sunday, Monday, Tuesday, Wednesday, Thursday, Friday, and Saturday.