NEWCASTLE UNIVERSITY

SEMESTER 1 2016/2017

PROGRAMMING AND DATA STRUCTURES
PART B

Total time allowed (for Parts A and B) - 3 Hours

Instructions to candidates:

Please read the Instructions to Candidates on the separate sheet carefully.

Answer ALL questions.

Marks shown for sub-sections are indicative only.

[Turn Over]

Question B1

A note-taking application has been written to allow people to record notes and search for given notes by keyword. Some code has already been written for this application, shown below.

```
public class Note
{
 private String subject;
 private String contents;

 public Note(String topic, String words)
 {
    subject = topic;
    contents = words;
 }

 /**
   * search note for some words, return true or
   * false
   */
 public boolean includes(String text)
 {
    // to be completed
 }
}
```

The class `Note` records subject and contents for a new text note. The `includes` method is incomplete.

a) Complete the body of the `includes` method. The method should return **true** if either the subject or the text of the note contains the `text` parameter. [8 marks]

A separate class, `Journal`, will maintain a list of `Note` objects together with the name of the person owning the journal. The class has been partially written, as follows:

```
import java.util.ArrayList;

public class Journal
{

    private String name;
    private ArrayList<Note> notes;

    // constructor method to be completed

    public int notesContaining (String text)
    {
        // count notes containing text

    }
}
```

b) Write a constructor method for the `Journal` class. New `Journal` objects should have a name initialised using a parameter to the constructor method, and an empty list of notes. [5 marks]

c) Write a method, `addNote`, to add a note object to the `notes` ArrayList of a Journal Object. [4 marks]

d) The `notesContaining` method will search all notes in the list for the given text and return a count of the number of notes which contain the text either in the subject or the contents. Complete the body of the `notesContaining` method. [8 marks]

Question B2

Write a superclass `Worker` and subclasses `HourlyWorker` and `SalariedWorker`. Every worker has a name and salary rate (per hour). Write a method `computePay(int hours)` that computes the weekly pay for every worker. An hourly worker gets paid the hourly wage for the actual number of hours worked, if `hours` is at most `40`. If the hourly worker worked more than `40` hours, the excess is paid at double time. The salaried worker gets paid the hourly wage for `40` hours, no matter what the actual number of hours is. To test the developed classes write a driver class containing a method `weeklyReport` that takes as parameters a `Worker` object and an integer representing the number of hours this worker worked in a given week, and returns a statement containing: the worker's name, an information whether she/he is an hourly worker or salaried worker and the amount of money earned by the worker for that week.

Your solution should contain the following classes:

a) `Worker` class containing a constructor and necessary methods; [7 marks]

b) `HourlyWorker` class containing a constructor and necessary methods; [7 marks]

c) `SalariedWorker` class containing a constructor and necessary methods; [4 marks]

d) `Test` class containing `weeklyReport` method. You do not need to provide a `main` method in the `Test` class. [7 marks]

You should avoid code duplication in your solution.

END OF PART B