University of Toronto
csc148—Introduction to Computer Science, Fall 1997

First Term Test — Horton's morning section

Duration: 50 minutes

Aids allowed: None

Make sure that your examination booklet has 6 pages (including this one). Write your answers in the spaces provided. Write legibly.

Family Name: _____ **Lecturer:** _____

First Name: _____ **Tutor:** _____

Student # : _____

1. _____ / 7
2. _____ / 4
3. _____ / 14
4. _____ / 10

Total _____ / 35

Question 1

[7 marks in total]

(a) [2 marks]
Which of the following is an ADT and which is a data structure? Circle the appropriate answer for each.

sorted array:	ADT	data structure
unsorted array:	ADT	data structure
ordered set:	ADT	data structure
circularly-linked list:	ADT	data structure

(b) [2 marks]
What two things must one specify in order to define an ADT?

(c) [3 marks]
Briefly explain what information-hiding is.

Why is information-hiding a good thing?

How do we achieve information-hiding in a Java program?

Question 2

[4 marks]

Consider the following method:

```
// Return true if s occurs in A, and false otherwise.
static boolean occursIn (String[] A, String s) {
    for (int i=0; i<A.length; i++) {
          // If find a match at position i, return true.
          if (                )
             return true;
    }
    // Didn't find a match anywhere, so return false.
    return false;
}
```

Consider the following two ways to fill in the missing condition:

1. A[i] == s
2. A[i].equals(s)
   Note that `equals` is a method in class `String` that returns true if two strings contain the same sequence of characters, and false otherwise.

(a) [3 marks]
Explain how the method's behaviour will differ depending on whether we use condition (1) or condition (2).

(b) [1 mark]
Suppose we were writing a program to read a newspaper story from a file, and check to see if any of the words in it are from a given vocabulary list, stored in an array of strings. Which version of this method would be appropriate to use? Circle one answer.

version with condition (1) version with condition (2)

Question 3

[14 marks in total]

In this question, you will use the `IntNode` class, given here:

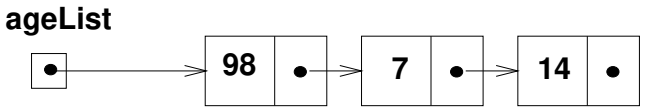
```
class IntNode {
   public int num;
   public IntNode next;

   // Constructor
   public IntNode(int n) {
      num = n;
      next = null;
   }
}
```

(a) [9 marks]
Complete the following Java method which attaches a new node containing the integer `n` to the end of the linked list referenced by `front`. Your method must also return a reference to the front of the revised list.

```
static IntNode append (IntNode front, int n) {
```

(b) [3 marks]
Using the `append()` method from part (a), write a fragment of Java code which will declare the variable `ageList` and create the linked list structure illustrated in the following diagram:



Don't worry about creating a method in which to put your code fragment. Assume that the append method is declared in the same class as your code fragment. Since it is declared `static`, it can be called directly, without naming a specific object whose `append` you are calling.

You can answer this question even if you have not completed part (a).

(c) [2 marks]
We chose to have the append method return a reference to the front of the revised list. Was this necessary? Circle the appropriate answer:

Yes No

Briefly explain your answer.

Question 4

[10 marks in total]

Suppose you are writing a program to keep track of inventory in a warehouse. The following ADT would be useful: An **Inventory Item** consists of an identification number, and a count of the number of units currently in stock. The following operations can be performed on an Inventory Item:

- record arrival: Record the fact that a given number of units of the Inventory Item have arrived at the warehouse.
- record shipment: Record the fact that a given number of units of the Inventory Item have been shipped out of the warehouse.
- number of units: Return the number of units of the Inventory Item that are currently in stock.

Write a Java class for implementing this ADT. Marks will be awarded for good design.