CMPUT 115 Section A1
Final Exam

December 19, 2000
Instructor: Jonathan Sillito

Instructions:

- This is a closed book, no notes exam.
- Put all of your answers in the space provided.
- Be sure to write your student id number on each internal page.
- Please do not open the exam until you are instructed to do so.
- Good luck.

First Name:

Last Name:

Student id: _____

1. [1 Mark] What is the difference between the **size** and **capacity** of a Vector?

2. [1 Mark] Name **one** benefit of using Arrays rather than Vectors.

3. [1 Mark] Name **one** benefit of using Vectors rather than Arrays.

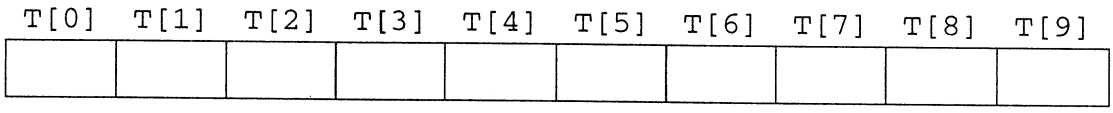
Question 4 refers to the following recursive method. Recall that `%` is the remainder operator, as an example `12%5` is 2.

```
public static int g( int n )
{
  if ( n <= 1 ) return 5;
  else if ( n%2 == 0 ) return g( n/2 ) + g( n-1 );
  else return g( n-1 );
}
```

4. [2 Marks] What value would be returned by the call **`g(4)`**?

5. [2 Marks] Suppose you have a hash table implemented using an array (`T`) of size 10, and suppose the hash table is storing Character objects using a hash function that maps each letter to its alphabet code (a = A = 0, b = B = 1, etc) modulo 10. Collisions are handled using linear-probing with a jump size of 1.

   Use the diagram below to describe the state of the table after each of the letters: i, d, c, M, and B are inserted.



Questions 6 and 7 refer to the following method.

```
public static void f( int[] A )
{
  for ( int i = 0; i < (A.length-1); i++ )
     for ( int j = i+1; j < A.length; j++ )
        if ( A[i] < A[j] ) {
           int t = A[j];
           A[j] = A[i];
           A[i] = t;
        }
}
```

6. [2 Marks] Assume that the array `A` has length 6 and the values in the array (before a call the method `f()`) are:

A[0]	A[1]	A[2]	A[3]	A[4]	A[5]
-6	5	9	-3	8	1

What will the array `A` look like after the method call **`f(A)`** returns? Put your answer in the following table:

A[0]	A[1]	A[2]	A[3]	A[4]	A[5]

7. [2 Marks] What is the maximum number of times the code inside the **`if`** statement (in the method **`f()`** above) could be executed? Express your answer in terms of `n` where `n` is the length of the array.

Student id: _____

8. [3 Mark] Below are two code fragments that differ only in the part that is in bold. Circle the fragment that would take **longest** to execute. (Hint: notice the order in which the elements are added.) Explain your choice in the space below the code fragments.

A.
```
OrderedStructure os = new OrderedList();
for ( int i = 10000; i >= 0; i-- )
  os.add( new Integer(i) );
```

B.
```
OrderedStructure os = new OrderedVector();
for ( int i = 10000; i >= 0; i-- )
  os.add( new Integer(i) );
```

9. [2 Marks] Under what conditions would you use an `OrderedVector` over an `OrderedList`?

10. [2 Marks] Draw the tree that would result from the following code.

```
BinaryTreeNode root = new BinaryTreeNode("A");
BinaryTreeNode x = new BinaryTreeNode("D");
x.setLeft( new BinaryTreeNode("F") );
x.setRight( new BinaryTreeNode("G") );
root.setRight( x );
root.setLeft( new BinaryTreeNode("E") );
```

Question 11 refers to the following method.

```
public static void visitTree( BinaryTreeNode node )
{
  if ( node == null ) return;
  visitTree( node.right() );
  visitTree( node.left() );
  System.out.println( node.value() );
}
```

11. [2 Marks] Assume that the above method was called with the root of the tree constructed in question 10, what would the output be? In other words what output would be produced as a result of the call **`visitTree(root)`**, where root is as in question 10?

12. [1 Mark] One way to sort a collection of (Comparable) objects is to insert each of them into a binary search tree. Approximately, what is the *average* time complexity of this sorting method? The following possible answers are expressed as functions of n, where n is the number of elements to be sorted.

    A. n^2

    B. n

    C. n(n-1)/2

    D. $n \log_2 n$

13. [1 Mark] If the sorting method described in question 12 was used on a collection of elements, what type of traversal of the tree would return the elements in sorted order?

    A. Inorder

    B. Preorder

    C. Postorder

    D. Levelorder

14. [3 Marks] Write a **recursive** method that returns the number of leaves in a tree.

```
public static int leaves( BinaryTreeNode root )
// post: returns the number of leaves in the tree rooted
//       at "root"
{
```

15. [4 Marks] Implement the **`get()`** method of the **`Hashtable`** class.

```
public class Hashtable implements Dictionary
{
  protected static Association reserved =
      new Association("reserved",null);
  protected Association data[];
  protected int count;
  protected int capacity;

  protected int locate(Object key)
  // pre: key is non-null
  // post: returns ideal index of key in table
  {
    // You may assume this has been implemented and
    // can be used in your get() method.
  }

  public Object get(Object key)
  // pre: key is non-null Object
  // post: returns value associated with key, or null
  {
```

16. [2 Marks] Briefly describe how an ordered vector could be used to implement something similar to the `Dictionary` interface?

17. [2 Marks] How would the `Dictionary` implementation described in question 16 be inferior to the hash table implementation in the structure package?

18. [2 Marks] In the structure package, the `SkewHeap` class does not have an `elements()` method (and does not have an associated Iterator class). Why is this?