

Name

ID:

COMP 103

Mid Trimester Test

17 August 2006

Time: 90 minutes

Marks: 90

Answer all questions.

Non programmable calculators permitted

Unannotated foreign language dictionaries permitted

1. Collection Types	16	<input type="text"/>
2. ArrayList	12	<input type="text"/>
3. Sorting	30	<input type="text"/>
4. Linked Lists	17	<input type="text"/>
5. Using Collections	15	<input type="text"/>
	<hr/>	
	90	<input type="text"/>

Spare page

Question 1. Collection Types.

[16 Marks]

(a) [4 marks] Which collection type would be a good choice for representing the collection of people waiting to be served at a supermarket checkout? Explain why.

(b) [4 marks] Which collection type would be a good choice for storing information about which employees are currently inside a building? Explain why.

(c) [4 marks] Which collection type would be a good choice for storing the information needed to retrieve automobile ownership records based on number plate? Explain why.

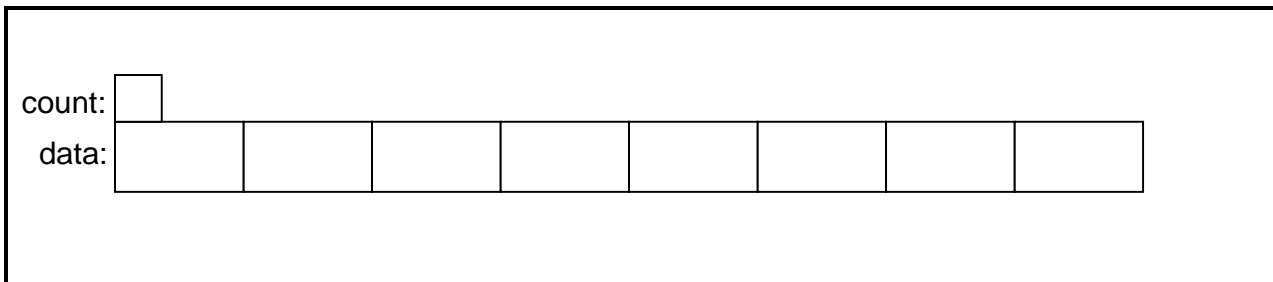
(d) [4 marks] Name a collection type that has constrained access and state the ways in which the access is constrained.

Question 2 ArrayList

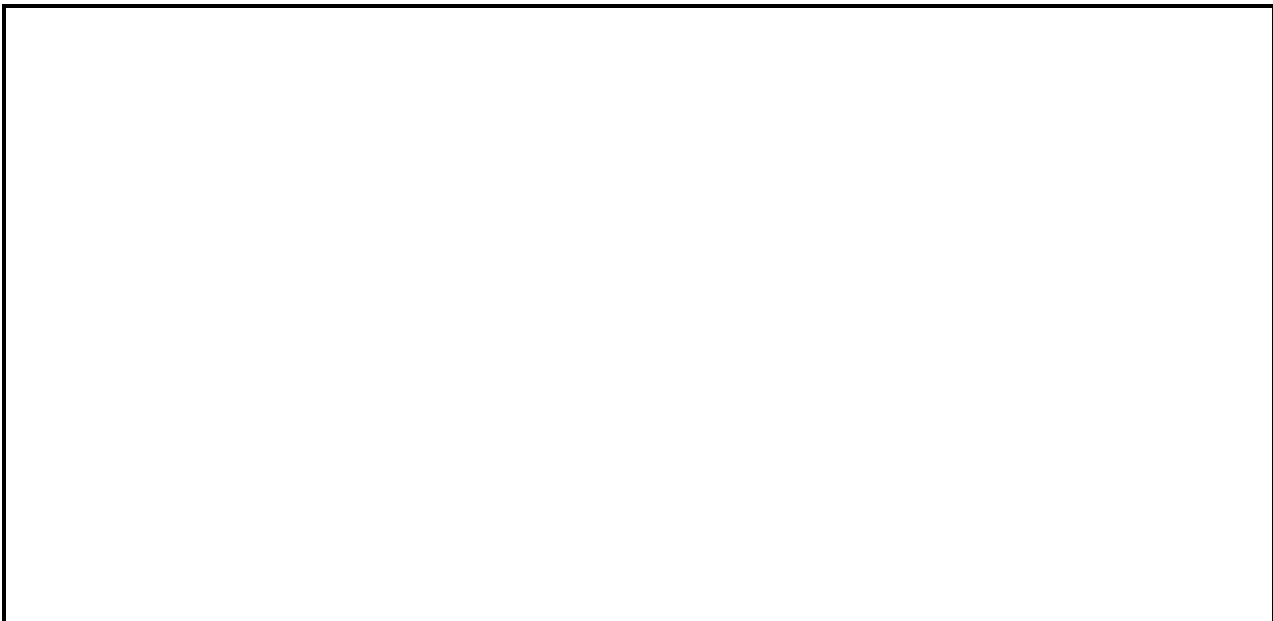
[12 Marks]

(a) [3 Marks] Draw a diagram showing the result of inserting the following three items into an ArrayList that initially contains ["Hum", "Dance", "Bee"]

insert "Buzz" at index 2,
insert "Hive" at index 2,
insert "Net" at index 0.



(b) [4 Marks] Draw a diagram showing how `ensureCapacity` increases the capacity of an ArrayList when the array becomes full. Label the steps on your diagram to make the sequence clear.



(c) [5 Marks] Consider the following `clear` method for the `ArrayList` class that removes all the items in an `ArrayList`. It uses the `remove` method, which removes the item at a given index). Assume that the `count` field stores the number of items in the `ArrayList`.

```
public void clear(){  
    int max = count;  
    for (int j = 0; j < max; j++)  
        remove(j);  
}
```

What is the cost of this method, expressed in terms of $n =$ the initial size of the list? Justify your answer.

Question 3 Sorting

[30 Marks]

(a) [8 marks] For each of the following sort algorithms, state its average case cost and worst case cost (using big-O notation), whether it is stable, and whether it is in-place.

	Average case	Worst case	Stable?	In-place?
Insertion Sort:				
MergeSort:				
QuickSort:				
Radix Sort:				

(b) [4 marks] Show how Bubble Sort sorts the following list of six items by showing the state of the list after each swap.

[A , B , D , F , E , C]

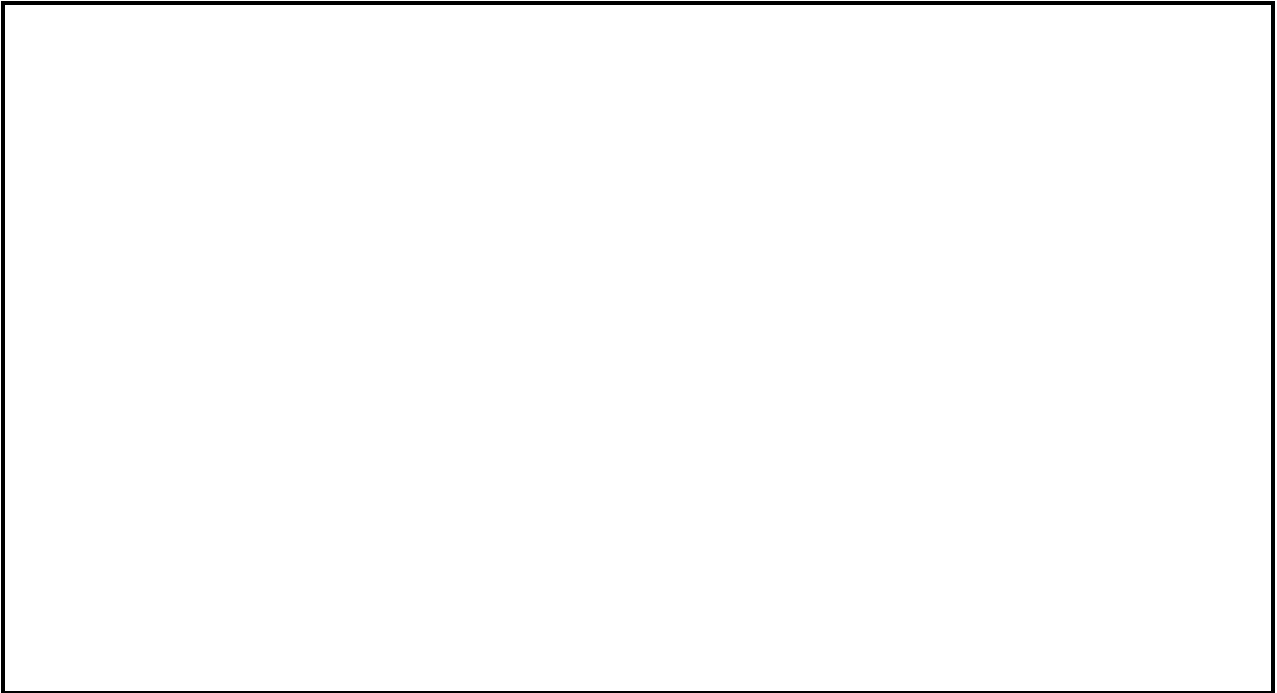
ID:

(c) [7 marks] Show how Quicksort sorts the following list of 9 items by showing the state of the list after each call to partition

You may use the partition algorithm described in the lectures. You may also use another standard partition algorithm, but you must then describe it very briefly.

[J , H , G , V , U , K , L , H , U]

(d) [7 marks] Outline the Insertion sort algorithm using pseudo code.



(e) [4 marks] Explain why InsertionSort is more efficient than QuickSort or MergeSort if you know that the list is almost sorted.



Spare Page.

Question 4 Linked Lists

[17 Marks]

(a) [2 Marks] Suppose the variable `myList` contains a linked list of three items (“A”, “B”, and “C”). Draw a diagram of the linked list.

myList:

(b) [2 marks] Show the changes to the linked list in (a) after the following statement:

```
myList = new ListNode("X", myList.next);
```

myList:

(c) [2 marks] Show the changes to the *original* linked list in (a) after the following statement:

```
myList.next.next = new ListNode("Y", null)
```

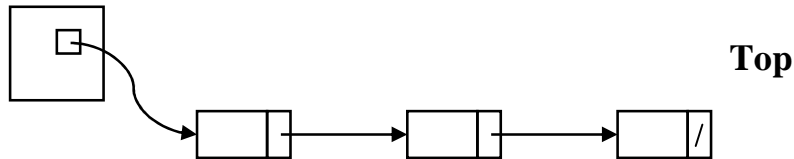
myList:

(d) [2 marks] Show the changes to the *original* linked list in (a) after the following statement:

```
myList.next.next.next = myList.next;
```

myList:

(e) [4 Marks] Suppose you implemented a Stack using a linked list with a header node as shown below, and chose to use the last node of the linked list as the top of the stack (where items are pushed and popped). Explain why this is a bad design.



(f) [5 Marks] Suppose you were implementing a Queue using a linked list and a header node containing two fields: front and back. Draw a diagram of how you would implement the queue, showing a queue with three elements. Explain why you chose this implementation.

Question 5. Using Collections

[15 Marks]

Most programming languages require that the syntax is properly nested. For example, an if statement cannot start inside a while loop but end outside the loop. A compiler must therefore check that every “opening” item has a matching “closing” item, ***and*** that the constructs are properly nested.

Consider a language that doesn't use brackets or braces, but uses the keywords **if** and **fi** to open and close an if statement, and the keywords **do** and **od** to open and close a loop. The program

```
“xxx do xxx xxx if yyy yyy fi xxx od zzz”
```

would be properly nested, but the programs

```
“xxx do xxx if yyy od xxx fi” and “xxx do yyy if yyy od zzz”
```

would not be properly nested.

Complete the following method that takes an argument containing a List of the words in a program and returns true if the program is correctly nested, and returns false otherwise. Assume that the only opening and closing words are **if**, **fi**, **do**, and **od**, and that all other words can be ignored.

```
public boolean checkNested(List<String> program){
```

```
}
```