NUI MAYNOOTH

Ollscoil na hÉireann Má Nuad

OLLSCOIL NA hÉIREANN MÁ NUAD

THE NATIONAL UNIVERSITY OF IRELAND MAYNOOTH

JANUARY 2012 EXAMINATION

CS210

Algorithms & Data Structures 1

Dr. M. McNeill, Dr. A. Winstanley, Dr. P. Maguire

Time allowed: 2 hours

Answer ***all*** questions

All questions carry equal marks

[20 marks]

1 (a) Design a method that takes in a person's full name as a single `String`, and extracts and returns their first name from that `String`. You can assume that the person's first and last names are always separated by a space. For example, if you inputted "John Smith" the method would return "John". [5 marks]

Provide a pseudocode or Java implementation of your algorithm and explain how it works.

(b) Design an algorithm that can solve the following problem: [5 marks]

What 4-digit number *abcd* satisfies the equation 4**abcd* = *dcba*?

Provide a pseudocode or Java implementation of your algorithm and explain how it works. (**Hint**: try all the possibilities for *a*, *b*, *c* and *d* – each represents a unique digit from 0 to 9)

(c) Design a method that takes in two `Strings` as parameters and returns the number of alphabetic letters that they share in common. For example, inputting the `Strings` "maynooth" and "nuim" would give a result of 2, because the letters *m* and *n* appear in both `Strings`. [5 marks]

Provide a pseudocode or Java implementation of your algorithm and explain how it works.

(d) Explain how the method below works. Show step by step how the values of `lowerbound` and `upperbound` would be updated if this method was used to search for the number 87 in the following array. (**Note**: the array is not quite ordered. Show how this causes the search to break down) [5 marks]

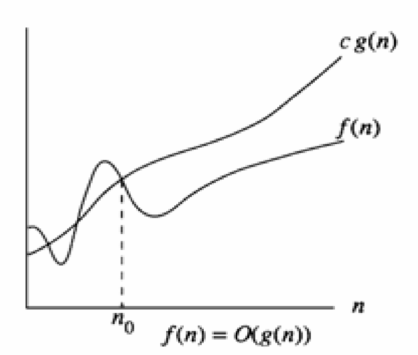
```
[16  24  45  52  57  61  63  69  89  87  91]
```

```
public int find(int searchKey){
   int lowerBound = 0;
   int upperBound = nElems-1;
   int check;
   while(true){
      check = (lowerBound + upperBound)/2;
      if(myArray[check]==searchKey){
         return check;
      }else if(lowerBound > upperBound){
         return -1;
      }else{
```

```
                if(myArray[check] < searchKey){
                   lowerBound = check + 1;
                }else{
                   upperBound = check - 1;
             }
          }
       }
```

[20 marks]

2 (a) Describe in your own words how Big O Notation represents algorithmic complexity. What does it tell you about an algorithm's performance? What does it not tell you? In your answer you should explain the significance of the diagram below: [4 marks]



(b) State the Big O complexity of the following real-world scenarios and explain your reasoning clearly: [4 marks]

i) Getting into an elevator and choosing the button for your floor, where *n* is the number of floors in the building
ii) Getting everybody at a party to shake hands with each other, where *n* is the number of people at the party
iii) Finding the tallest person in your class, where *n* is the number of people in your class
iv) Finding out how many people in your class share birthdays with somebody else in the class, where *n* is the number of people in your class

(c) Describe how the following sorting algorithms work and show clearly how they would sort the following numbers: [6 marks]

i) Insertion sort
ii) Merge sort

```
54    87    12    83    37    27
```

(d) Design a sorting algorithm which operates on an array of unsorted `Strings` and sorts them, first by length, and then alphabetically within `Strings` of the same length. For example, both “cat” and “zoo” would come before “table”, because they have a length of 3, and “cat” would come before “zoo” because of their dictionary ordering. [6 marks]

Provide a pseudocode or Java implementation of your algorithm and explain how it works.

[20 marks]

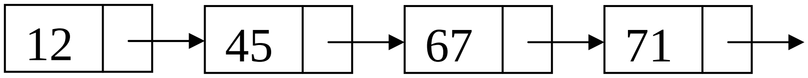
3 (a) Describe how the following data structures work, using diagrams as appropriate: [4 marks]

i) Priority Queue
ii) Deque

(b) Design a method that uses a `Stack` object to check whether or not a `String` is a palindrome (i.e. reads the same forwards or backwards). You can assume that a `Stack` class is available with push() and pop() methods. [4 marks]

Provide a pseudocode or Java implementation of your algorithm and explain how it works.

(c) Explain how the method below works. Describe step-by-step how it would insert the number 63 into the following linked list, showing how the variables `previous` and `current` are updated: [4 marks]



```
public void insert(Link newlink){

    Link previous = null;
    Link current = first;
    while(current != null && newlink.data >
current.data){
        previous = current;
        current = current.next;
    }
    if(previous==null){
        first = newlink;
    }else{
```

```
                previous.next = newlink;
                newlink.next = current;
            }
        }
```

(d) Design an algorithm for reversing the contents of a single-ended singly-linked list. Describe the general concept in your own words, using diagrams as appropriate. [4 marks]

(e) Write a recursive method that calculates the return on an investment earning compound interest. The method should take in three parameters, namely the initial sum invested, the interest rate applicable, and the number of years for which the sum is invested. For example, if you invest €1000 at 10% for two years, then at the end of year 1 you earn 10% of €1000, and at the end of year two you earn 10% of the total you had at the end of year one, yielding a total of €1,210 after two years. [4 marks]

Provide a pseudocode or Java implementation of your recursive algorithm and explain how it works.