University of Glasgow

**Monday, 29 April 2019**
**9.30 am – 11.00 am**
**(1 hour 30 minutes)**

**DEGREES OF MSci, MEng, BEng, BSc, MA and MA (Social Sciences)**

# ALGORITHMS AND DATA STRUCTURES 2: COMPSCI2007

**Answer all questions**

**This examination paper is worth a total of 60 marks.**

**The use of calculators is not permitted in this examination.**

## INSTRUCTIONS TO INVIGILATORS: Please collect all exam question papers and exam answer scripts and retain for school to collect. Candidates must not remove exam question papers.

**1.** Consider the algorithm described by the following pseudocode:

```
1:  F(A, x, p, r)
2:    if p > r
3:      return −1
4:    q := (p + r/2)
5:    if A[q] = x
6:      return q
7:    else
8:      if A[q] > x
9:        return F(A, x, p, q − 1)
10:     else
11:       return F(A, x, q + 1, r)
```

The inputs of algorithm **F** are: a sorted array of integers $A$, an integer $x$, and two indices for $A$ $p, r$.

**(a)** Briefly explain what algorithm **F** implements. [5]

**(b)** What is the output of $\mathbf{F}(A, 13, 0, 7)$, where $A = [0,4,4,5,8,7,13,14]$? [2]

**(c)** Draw the recursion trace for $\mathbf{F}(A, 13, 0, 7)$. [3]

**(d)** Is **F** tail recursive? Justify your answer. [2]

**(e)** Is **F** linear recursive? Justify your answer. [2]

**(f)** Write the recurrence equation for the running time $T(n)$ of algorithm **F** (with $n = r - p + 1$). [2]

**(g)** Solve the recurrence equation in **1(f)** to compute the complexity of algorithm **F** using big-Oh notation. [4]

**2.** Rank the following functions by order of growth; that is, find an arrangement $f_1, f_2, \ldots, f_8$ of the functions satisfying: $f_{i-1} = O(f_i)$.

$$\log\log n \qquad 1 \qquad n^3 \qquad 2^n \qquad n \qquad \sqrt{n} \qquad n\log n \qquad \log n$$

[5]

**3.** **(a)** Describe using pseudocode or otherwise the merge sort algorithm for sorting a list of integers contained in an array. Illustrate your description by sorting the list: 6, 3, 1, 7, 4, 5, 2. Note that you do not need to describe the merge procedure in detail. [8]

**(b)** What is the time complexity of merge sort? [1]

**(c)** Assume the quicksort algorithm is implemented using a partitioning scheme that selects the middle element (i.e. element $A[q]$ in subarray $A[p..r]$ where $q = \lfloor (p+r)/2 \rfloor$) as the pivot. Show an input of length 8 exhibiting worst case complexity $O(n^2)$. Explain your answer. [4]

**4.** **(a)** Briefly describe the two primary operations of the queue ADT. [2]

**(b)** Briefly describe the linked list implementation of the queue ADT. What is the complexity of the primary operations? [4]

**5.** **(a)** What is a binary search tree? Does the order in which a set of values is inserted into a binary search tree matter? Justify your answer. [6]

**(b)** Give a recursive definition of the preorder, inorder and postorder traversals of a binary tree. [6]

**(c)** Define using pseudocode or otherwise the insert operation on binary search trees. [4]



END OF QUESTION PAPER