

OLLSCOIL NA hÉIREANN, GAILLIMH
NATIONAL UNIVERSITY OF IRELAND, GALWAY

SUMMER EXAMINATIONS, 2004

FINANCIAL MATHEMATICS AND ECONOMICS

CS209 - ALGORITHMS AND SCIENTIFIC COMPUTING

Dr. Dave Johnson

Prof. T. Hurley

Dr. A. McCluskey

Dr. P. O'Leary

Time allowed: *Two* hours.

Full marks for *three* questions.

1. (a) What does it mean to say that an algorithm *correctly computes* a certain function?

Show that $z + xy$ is a loop invariant for the algorithm below and prove that the algorithm correctly computes the function $f(m, n) = mn$, where m, n are strictly positive integers.

```

 $x := m; \quad y := n; \quad z := 0;$ 
while  $y \neq 0$  do
     $z := z + x$ 
     $y := y - 1$ 
 $p := z$ 

```

Discuss whether the condition 'strictly positive' is necessary for integers m, n in order for the algorithm to correctly compute their product.

- (b) Illustrate *clearly* the sorting algorithm Tournament sort by using it to arrange the numbers 3, 7, 6, 9, 5, 2, 4 in natural order.

Prove that Tournament sort has (worst-case) complexity $O(n \log n)$.
(Hint: Assume first that $n = 2^k$.)

2. (a) What is meant by an algorithm being (i) partially correct
(ii) correct?

Show that $\text{gcd}(x, y)$ is a loop invariant for the algorithm below and prove that the algorithm correctly computes the function $f(m, n) = \text{gcd}(m, n)$, where m, n are strictly positive integers.

```

x := m;  y := n;
while    x ≠ y    do
    if      x > y
    then    x := x - y
    else    y := y - x
f := z

```

If m and n are allowed to be non-negative integers, explain why the algorithm is partially correct, but *not* correct.

- (b) Given a binary tree of height h with k leaves, it is true that $k \leq 2^h$. Use this fact to prove that every comparison-based sorting algorithm has complexity at least $O(n \log n)$.

3. (a) The algorithm below is supposed to compute the quotient q and remainder r when m is divided by n where m and n are positive integers. Find a loop invariant and prove that the algorithm is correct.

```

x := m;  y := n;  z := 0;
while    x ≥ y    do
    x := x - y;
    z := z + 1;
q := z;
r := x

```

- (b) Illustrate the sorting algorithm Mergesort by using it to arrange the numbers 3, 7, 6, 9, 5, 2, 4 in natural order.

Prove that Mergesort has (worst-case) complexity $O(n \log n)$.

4. (a) Define the following terms with respect to a given algorithm:
- (i) complexity
 - (ii) worst-case complexity
 - (iii) average complexity
- What is meant by saying " $c(n)$ is $O(n \log n)$ "?
- (b) Illustrate the sorting algorithm Tree sort by using it to sort the sequence of letters

C O N F I G U R E

in alphabetical order.

Show that Tree sort has *average* complexity $O(n \log n)$.

5. (a) Explain the purpose and illustrate the use of the Maple commands
- (i) sort, (ii) coeff, (iii) diff, (iv) Diff.
- (b) Explain the differences between and illustrate the meaning in Maple of
- (i) sets, (ii) lists.
- (c) Explain the meaning of the use of global, local commands in a Maple procedure. The Bessel function of the first kind $J_n(x)$ are defined by

$$x J_{n+1}(x) = 2n J_n(x) - x J_{n-1}(x).$$

If $J_0(z) = 1$ and $J_1(z) = z$ write an efficient Maple procedure to evaluate $J_{20}(z)$.