GX3084

OLLSCOIL NA hÉIREANN, GAILLIMH
NATIONAL UNIVERSITY OF IRELAND, GALWAY

SEMESTER I EXAMINATIONS 2002/2003

SECOND YEAR COMPUTER SCIENCE [CS201]
OPERATING SYSTEMS and C++ [CS208]

Dr Dave Johnson
Prof. T. Hurley
Dr Niall Madden

Time allowed: **Two** hours.
Attempt **FOUR** questions.

1. (a) What is a computer *Operating System* (OS)? List three typical computer operations that an Operating System is responsible for, and three that user programs are responsible for.
   (b) What is meant by a *multiprogramming system* and a *multitasking system.* Explain the main goal of each system and how the operating system attempts to achieve that goal.
   What components must a multiprogramming system have that you would not find in a non-multiprogramming system?
   What OS components must a system have in order to be multitasking?

2. (a) Explain what a *process* is—in particular distinguish between a process and a program.
   (b) With respect to Synchronization of Cooperating Processes, explain, with an example, what is meant by a *race condition.*
   How can a (binary) Semaphore be used to avoid a race condition?

3. (a) Explain the problem of (memory) fragmentation caused by *contiguous memory allocation.* Include a description of *external* and *internal fragmentation.*
   (b) Explain how **paging** is used as a method of *noncontiguous* memory allocation.
   Why would you not expect any external fragmentation?
   How much memory would you expect to be lost due to internal fragmentation? Why?

4. (a) Compare and contrast Windows 2000 and Linux with respect to **two** of the topics listed below.
      (i) Memory management.
      (ii) Licensing.
      (iii) The kernel
   (b) With respect to Object Oriented Programming, write a short note on **two** of the following topics:
      (i) Encapsulation
      (ii) Inheritance.
      (iii) Polymorphism

5. *(This question refers to any standard Unix/Linux shell, e.g., the* bash *shell)*

(a) What is meant by **Input/Output redirection**, and how is it achieved? In the course of your answer, explain what is meant by *standard input*, *standard output*, and *standard error*. Give examples of how to

(i) redirect the output of a program to a file, overwriting its previous contents.

(ii) append the output of a program to a file that already exists.

(iii) redirect the input so that it is taken from a file.

(iv) pipe the output of one program directly to the input of another.

(b) Consider the following shell script:

```
#! /bin/bash
NUMBERS=$(seq 1 $1)
ANS=1
for i in $NUMBERS
do
  let ANS=$[ ANS*i ]
done
echo Answer: $1! is $ANS
```

(i) Explain the purpose of this program, and describe what each line does. Explain the usage of the positional parameter `$1`.

(ii) Write down a version of this script modified so that the user is prompted input if none is given, and an error is reported if more than one input is provided.