THE BRITISH COMPUTER SOCIETY

THE BCS PROFESSIONAL EXAMINATION
Diploma

SYSTEMS SOFTWARE

26th April 2000 - 10:00a.m. - 12:00p.m.
Answer FOUR questions out of SIX. All questions carry equal marks.
Time: TWO hours.

The marks given in brackets are ***indicative*** *of the weight given to each part of the question.*

1. Most block-structured high-level programming languages permit a procedure to be declared within another procedure.

Consider such a case, in which a procedure named INNER is declared inside another named OUTER:
both are recursive;
both INNER and OUTER have parameters and variables;
a variable V, declared in OUTER, is also referred to by the instructions in INNER.

When OUTER has recursed, there will be an instance of V for each instance of OUTER. Assume that at such a time a reference to V from an instance of INNER will be to the V of the most recent instance of OUTER.

Describe how the run-time stack can be organised to support this language capability. **(15 marks)**

Include how parameters and other variables of both procedures would be incorporated. **(10 marks)**

2. How is virtual memory implemented for *either* a paged system *or* a segmented system? Describe the necessary hardware aspects and the data structures used, and the ways in which those structures are employed for address translation and for memory management. Illustrate your answer with examples. **(25 marks)**

3. *a)* Using a high-level programming language, write a sequence of statements using a single loop which examines the components of a one-dimensional array of integers with an arbitrary range of indexes N to M, summing the values at indexes divisible by four into a variable named SUMFOUR and summing the values at other indexes into a variable named SUMOTHER. **(5 marks)**

b) By writing assembler code for an actual or fictitious CPU, show how the compiler could optimise the test for "divisible by four". **(5 marks)**

c) Suppose that in the code you wrote for part *a)*, the compiler can determine that N=1 and that M is divisible by four. By re-writing the high-level code, and (if needed) by writing assembler code equivalent to key parts of it, explain how the compiler can optimise the loop as a whole to maximise speed. **(15 marks)**

4. Although the hardware interfaces of different peripheral devices differ, it is desirable for uniform software interfaces to be provided for them. Describe, with examples, typical hardware and software interfaces. **(10 marks)**

What mechanisms are provided to connect the two types of interface for a given hardware device? **(5 marks)**

Describe the activities typically carried out by such a mechanism. **(10 marks)**

5. Explain why an interpreted program will run much slower than a corresponding compiled program. **(25 marks)**

6. List and describe in outline the components which make up an operating system of your choice, which should be named. **(10 marks)**

   Using suitable examples, show how these components interact with each other to satisfy the users' requirements. **(15 marks)**