Computer Science Department

1997 Examinations

C1

QUESTION PAPER

DUMMY COVER

ORIGINAL

EXAMINERS COPY

EXAM Co-ordinator

Peter Rounce

Ext 7291

Answer ALL 4 questions

The use of Electronic Calculators is not permitted in this examination.

Q1 Answer parts a) **and** b) and **either** part c) **or** part d)

a) Study the function `mystery` below:

```
void mystery(int* a, int* b, int c){
    for (int i = 0; i < c; i++){
        int x = 0;
        for (int j = 0; j < c; j++){
            if (a[j] < a[i]){
                x += 1;
            }
        }
        b[x] = a[i];
    }
}
```

i) The following code snippet shows the declaration of two arrays and a call of the function `mystery`. What do the two arrays contain after the call to `mystery`?

```
int p[4];
p[0] = 3; p[1] = 2; p[2] = 0; p[3] = 1;
int q[4];
q[0] = 2; q[1] = 4; q[2] = 1; q[3] = 5;

mystery(p,q,4);
```

[3 marks]

ii) What do you think the purpose of the function is? There are certain kinds of parameters for which the function does not work correctly: describe these cases.

[3 marks]

iii) Write a function prototype for the function, renaming the function and its parameters with meaningful names. Put as many '`const`' modifiers for the parameters as is allowed. Write a short comment to go with the function prototype.

[3 marks]

[Question 1 continued over page]

[Question 1 continued]

b) Write a function `**count_occurences**' that takes two strings, and returns the number of times that the second string is contained in the first. For example, the call:

```
count_occurences("stop all the clocks", "all");
```

returns the value 1, and the call:

```
count_occurences("ababa","aba");
```

returns the value 2.

Minor errors in C++ syntax will not be penalised.

[8 marks]

Answer either question c) or question d)

c) *This is an alternative to question d)*

Novice programmers often find it hard to decide how to break up a program into functions. Suggest some guidelines to use in determining whether a piece of code should be made into a function.

[7 marks]

d) *This is an alternative to question c)*

Using global variables often seems convenient when trying to get code working in limited time. Why is the use of global variables seen as a bad thing despite this apparent convenience?

[7 marks]

Q.2 Answer any **two** of parts a) b) and c)

a) A computer has a colour "*bit-mapped*" display which is accessed directly across the bus. Describe the main components and operation of such a system. Your answer should explain the role of *pixels* and *video RAM* but should not concern itself with electrical details nor the physical operation of the Cathode Ray Tube (CRT).

[12 marks]

b) i) What is the meaning of "*relative addressing*" and what advantages does it confer?

[5 marks]

ii) Explain how relative addressing may be implemented as a special case of *indexed addressing*.

[3 marks]

iii) The M68000 processor employs "relative branch" instructions in which the offset is calculated from the address of the 16-bit word immediately following the op-code. In the program fragment below, given that loop and end correspond to addresses 0x40003000 and 0x40003014 respectively and that the 16-bit opcode for bra is 0x6000 what would be the encoding of the bra instruction below expressed in hexadecimal notation?

```
loop: ...
      ...
 end: bra loop
```

[4 marks]

[Question 2 continued overleaf]

[Question 2 continued]

c) *In this question you are asked to write a small program. This should be written in a representative assembly language. You will not be marked on the accuracy of your use of any particular language, however, you must restrict yourself to typical assembly language constructs and must make clear the meaning of any novel notations you introduce.*

A character string is stored as a byte-array of ASCII codes starting at location 0x40002000. The string is terminated by a zero byte. Write an assembly language program to search the array for a character whose ASCII code is stored in register D3. The program should terminate with 1 stored in register D0 if the character is present, 0 otherwise.

Q3. Answer any **two** of parts a) b) and c)

a) i) Explain the meaning of the following sentence: “The magnetic disc drive has 4 *surfaces*, 1000 *tracks* per surface and 50 *sectors* per track”.

[6 marks]

ii) Assuming 1 Kbytes per sector, what is the raw capacity of each *cylinder* of the disc in i)?

[2 marks]

iii) The disc in i) has an average seek time of 10 ms and rotates at 6000 r.p.m. Estimate the average rotational latency and explain why it is beneficial to store all the data from a file in a single cylinder.

[4 marks]

[Question 3 continued overleaf]

[Question 3 continued]

b) i) Explain the following Operating Systems terms; *process*, *pre-emption*, *time-slice*.

[6 marks]

ii) Multiple processes running on a multi-processed, pre-emptive operating system all access a single resource. The code for each process includes the code fragment below which is intended to ensure that just one process may access the resource at any time. In this code sleep() is a system call which asks the operating system to suspend the current process until some other process executes a wakeup() system call.

```
while (flag = 0) {
        sleep();
}
flag = 1;
access_shared_resource();
flag = 0;
wakeup();
```

Explain why this code does not perform as intended.

[6 marks]

c) i) Define *virtual memory* and briefly outline how *paging* may be used to implement it. Your answer should make reference to the *page table*, *the memory management unit* (*MMU*) and *page faults*.

[6 marks]

ii) A MMU typically has a small amount of memory which operates as a *cache*. Explain how such a cache may improve the performance of the memory management system.

[3 marks]

iii) A paged *virtual memory* system offers a *32-bit address space* and uses 2048-byte pages. How many virtual pages would there be (expressed as a power of 2) and how many bits would be allocated to the page addresses?

[3 marks]

Q4 Answer part (a) and then three parts from (b)-(f).

(a) Describe what is meant by a real-time computing system and explain how such a system differs from a conventional system such as a network of UNIX workstations or a personal computer.

[7 marks]

(b) Describe what is meant by the real-time executive in a real-time computing system and what kind of functions it typically carries out. Explain why it is desirable that the executive should be protected in a real-time computing system and indicate what steps are usually taken in order to do so.

[6 marks]

(c) Explain what is meant by (i) the interrupt, (ii) clock and (iii) base, priority levels in a real-time computing system.

[6 marks]

(d) Explain the difference between “hard” and “soft” real-time systems and describe what effect this has on the way tasks may be scheduled in the two types of system and on how errors are handled . Discuss how these systems differ from ordinary interactive system.

[6 marks]

(e) Discuss the reasons why the design, implementation and testing of real-time systems is usually considerably more difficult than for conventional computing systems.

[6 marks]

(f) Give three reasons why multiprocessor systems often used in building real-time computing systems and describe a multiprocessor architecture well suited to meeting each of the requirements you give.

[6 marks]

 END OF PAPER