PAPER CODE NO.	EXAMINER : **MR PH Leng**	
COMP103	DEPARTMENT : Computer Science	Tel. No. 43673

SEPTEMBER 2000 EXAMINATIONS

Bachelor of Engineering : Year 1
Bachelor of Science : Year 1

COMPUTER SYSTEMS

TIME ALLOWED : Two Hours

INSTRUCTIONS TO CANDIDATES

Answer *ALL* questions in section A and answer *THREE* questions from section B

If you attempt to answer more than the required number of questions (in any section), the marks awarded for the excess questions will be discarded (starting with your lowest mark).

Section A (40 marks)

1. Explain what is meant by a processor **register**. Describe the roles played by the **Program Counter** and the **Instruction register** in the execution cycle of a machine-code instruction.

(6 marks)

2. Describe briefly how negative numbers are represented in twos complement binary notation. Illustrate your answer by showing the representation of the number -5 as an 8-bit signed binary number.

(5 marks)

3. Explain the difference between **immediate, direct** and **indirect** addressing modes. Give examples of each in the 68000 assembly language.

(5 marks)

4. Write a sequence of 68000 assembly-language instructions to perform the equivalent of the Java conditional statement:

```
if (p>=q)
   max=p;
else
    max=q;
```

(5 marks)

5. Distinguish between a **combinational** logic circuit and a **sequential** logic circuit. Explain the function of a **flip-flop** circuit.

(4 marks)

6. How are **status conditions** recorded in the 68000? Which of the following operations, carried out in twos complement (signed) arithmetic, will lead to (a) c**arry** status or (b) **overflow** status arising?

(i) addition: 125 + 8
(ii)subtraction: 2-3
(iii) addition: -5 + (-128) (5 marks)

7. Explain the difference between (a) a **program** and a **process**; (b) a **foreground** process and a **background** process. How is a background process initiated by a Unix command?

(5 marks)

8. Explain the difference between **absolute** and **relocatable** binary code. What is the advantage of writing code that is **dynamically** relocatable? Briefly discuss how this can be done in the 68000 assembly language.

(5 marks)

Section B

B1. (a) Write a 68000 assembly-language subroutine which will exchange the values of two variables, references to which are passed as parameters. Explain the means you are using to pass the parameters.
(NB: minor errors in the form of the instructions you write will not be penalised).
(10 marks)

(b) Write a UNIX shell script which will exchange the contents of two files, the names of which are passed as parameters. What action is required to make your shell script executable?
(10 marks)

B2. The piece of program below, in the 68000 assembly language, is written with the address of each instruction shown as a decimal number in the left-hand column.

(a) Draw up an execution history of the program, tabulating the changes in the values of the PC and the other registers used , for three complete cycles of the loop sequence.
(14 marks)

(b) Suppose that a breakpoint has been inserted in the program at address 2024. Tabulate the register values that would be displayed each time this breakpoint is reached as the program runs to completion.
(6 marks)

```
2000            move #0,d0
2004            lea array, a0
2008            move (a0)+, d1
2010  loop      move (a0)+, d2
2012            blt minus
2016            add d2, d0
2018            jmp endloop
2022  minus     sub d2, d0
2024  endloop   sub#1,d1
2028            bnz loop

3000  array     dc.w 10, 5, -3, 12, 27, -8, -2, 14, 20, 25, 7
```

B3. (a) Explain what is meant by an **elementary logic gate.** Draw and explain truth tables to represent the operations of **and, or** and **not** gates. Briefly explain why logic gates of this kind are significant in the implementation of digital computers.

(7 marks)

(b) By writing out an extended truth table, or otherwise, calculate the value of the logic function:

P= (**not** (A **and** B) **and** (A **or** B)) **or** C

(8 marks)

(c) Draw a logic circuit for the expression of part (b).

(5 marks)

B4 (a) Describe the main steps carried out by a typical High-Level Language compiler.

(10 marks)

(b) How is it possible for systems like UNIX to service a large number of users simultaneously, even when only one processor is involved? Identify the role played by interrupts in this respect.

(10 marks)