

Office Use Only

--	--	--

**Monash University**  
**Semester One Examinations 2000**  
**Faculty Of Information Technology**

**EXAM CODES:** CSE3304

**TITLE OF PAPER:** COMPUTER ORGANISATION

**Part B**

**EXAM DURATION:** 180 minutes writing time for Part A and B

**READING TIME** 10 minutes

**THIS PAPER IS FOR STUDENTS STUDYING AT:( office use only - tick where applicable)**

Berwick  Clayton  Peninsula  Distance Education  Open Learning   
 Caulfield  Gippsland  Sunway  Enhancement Studies  Other (specify)

*Candidates must complete this section*

STUDENT ID \_\_\_\_\_ DESK NUMBER \_\_\_\_\_  
 SURNAME .....SIGNATURE.....  
 OTHER NAMES (in full) .....

Candidates are reminded that they should have no material on their desks unless their use has been specifically permitted by the following instructions.

**AUTHORISED MATERIALS**

**CALCULATORS** YES  NO

**OPEN BOOK** YES  NO

**SPECIFICALLY PERMITTED ITEMS** YES  NO

**INSTRUCTIONS TO CANDIDATES**

1. Print your name and ID number in the section above.
2. Answer all questions in the space provided.
3. Individual marks are indicated for each question.
4. Total marks for Part B of this examination are 100
5. Calculators are NOT allowed.
6. This paper must be returned.

OFFICE USE ONLY		
Q1	20	
Q2	14	
Q3	36	
Q4	30	
Total	100	



- (e) Now assume that there **is register renaming hardware**. Show the resulting sequence of operations. (2 marks)

F	D	I	E	W								R1 ← R2 + R3
												R1 ← R4 + R5

- (f) Assume that there **is by-pass hardware** is available. Show the resulting sequence of operations. (2 marks)

F	D	I	E	W								R1 ← R2 + R3
												R4 ← R1 + R5

- (g) Assuming **no** branch prediction hardware, and that the PC is updated during the W phase. Show the resulting sequence of operations. (2 marks)

F	D	I	E	W								90: Branch to 100
												100: R1 ← R2 + R3

- (h) Assuming there **is branch prediction hardware**, and that the PC is updated during the W phase. You can assume that the branch is predicted correctly. Show the resulting sequence of operations. (2 marks)

F	D	I	E	W								90: Branch to 100
												100: R1 ← R2 + R3

- (i) Imagine that the timing of the execute phase varies from 1 to 5 cycles, and that the exact timing is not known when an instruction is fetched. In the following example, the first instruction takes 4 cycles to perform the addition. Would this alter your answer to part (d)? (4 marks)

If so, how

F	D	I	E	E	E	E	W					R1 ← R2 + R3
												R1 ← R4 + R5

why




## Superscalar Execution

- Q3 Consider a pipelined machine that can issue one instruction per cycle, but can execute multiple instructions concurrently. Each instruction takes a different amount of time to execute – a number of different scenarios are shown in Table 1. (36 marks)

Instruction type	Scenario 1	Scenario 2	Scenario 3	Scenario 4
Load	1 cycle	5 cycles	5 cycles	5 cycles
FADD	1 cycle	3 cycles	5 cycles	5 cycles
FMUL	1 cycle	5 cycles	7 cycles	7 cycles
IADD	1 cycle	2 cycles	1 cycle	1 cycle
Store	1 cycle	1 cycle	2 cycles	5 cycles

Table 1 Instruction timings

In this question, you are asked to consider the program in Table 2, and show the instruction timing assuming no shelving hardware, shelving using Tomasulo's algorithm and shelving using Thornton's algorithm for each scenario. You can assume that the memory variables A, B and C are in different memory banks, so they can all be accessed concurrently. You should assume that a Store operation reserves the source register until the Store completes.

Load R1, A
Load R2, B
FADD R1, R2 → R3
Store R3, C
IADD R4, 1 → R4
FMUL R5, 4 → R3

Table 2 Program







## Branches

- Q4 The following loop for computing factorial(4) is executed on a pipelined machine. You should assume a 4 stage pipeline, consisting of Fetch (F), Decode (D), Execute (E) and Write (W). (30 marks)

When a control transfer occurs the PC is updated during the **Write** phase during a branch instruction. The BGT instruction branches if the predicate is greater than 0. Assume that the branch predicate ( $R1 > 0$ ) is known by the end of the **Execute** phase. In all other cases, you should also assume the machine **has by-pass hardware**, so the result of an instruction can be made available in time for the next instruction if necessary. This means that there should be no data dependencies which stall execution other than branch delays. However, you should also assume that a branch must be flushed from the pipeline before the target instruction can be fetched. This means that the Fetch of a branch target can only start after the Write phase of the branch.

Address	Instruction
100	$R1 \leftarrow 4$
101	$R2 \leftarrow 1$
102	$R3 \leftarrow 1$
103	$R3 \leftarrow R3 * R2$
104	$R1 \leftarrow R1 - 1$
105	$R2 \leftarrow R2 + 1$
106	BGT R1,103
107	$R4 \leftarrow 1$
108	$R5 \leftarrow 1$

Calculate the number of cycles require to implement the program:







