| | | | | |
|---|---|---|---|---|
| | Course Name: | Computer Architecture | Course Code: | EE204 |
| | Program: | BS (Computer Science) | Semester: | Fall2019 |
| | Duration: | 60 Minutes | Total Marks: | 30 |
| | Paper Date: | 08-11-2019 | Weight | 15 |
| | Exam Type: | Midterm II | Page(s): | 4 |

**Student : Name:_____ Roll No._____ Section: _____**

Instruction:
1. Attempt all question in the provided space. You can use rough sheets but it should not be attached.
2. If you think some information is missing, write assumption and solve accordingly.

## Question 1a [6](Multiple options can be selected in MCQs)

1. A hardware scheme in which variable number of instructions can be launched in a single cycle is called _____.

2. In _____ branch prediction, we need to evaluate previous branch results to predict current branch decision.

3. Typically during pipelined execution, the read operation on the register file and the memory is performed during _____
   A. First half of the cycle
   B. Later half of the cycle
   C. Any half of the cycle
   D. Throughout the cycle

4. CPU time can be reduced by reducing:
   A. Instruction count
   B. Clocks per instruction
   C. Clock cycle time
   D. Clock frequency

5. When compiler attempts to schedule instructions to avoid hazard; this approach is called
   A. Structuring
   B. Static scheduling
   C. Dynamic scheduling
   D. None of the above

6. Structural hazards can be avoided by
   E. adding stalls
   F. forwarding
   G. adding more hardware
   H. register renaming

## Question 1b [2+2]

1. **State whether the following techniques or components are associated primarily with a software- or hardware-based approach. In some cases, the answer may be both.**
   A. Branch prediction
   B. VLIW
   C. Superscalar
   D. Static scheduling
2. **Will a speedup of 20 times on 50% of a program result in an overall speedup of at least 2 times? Explain your answer.**

## Question 1c [4]

Consider the following code snippet's execution on a 5-stage pipelined processor. Hazard detection and full forwarding is implemented

```
Instruction 1:   LwR1, R2, 20
Instruction 2:   subR2, R3, R4
Instruction 3:   Add R1,R1,R2
Instruction 4:   StR1, R2, 40
Instruction 5:   Add R4,R5,R4
```

1. Is the following condition true for Instruction 1 and Instruction 3 at the start of 5th clock cycle?
   EX/Mem.RegisterRt = ID/EX.RegisterRs
   **True/False: _____**

2. Is the following condition true for Instruction 1 and Instruction 3 at the start of 5th clock cycle?
   MEM/WB.RegisterRt = ID/EX.RegisterRs
   **True/False: _____**

3. Is the following condition true for Instruction 2 and Instruction 4 at the start of 6th clock cycle?
   EX/MEM.RegisterRd = ID/EX.RegisterRs
   **True/False: _____**

4. Is the following condition true for Instruction 3 and Instruction 4 at the start of 6th clock cycle?
   EX/MEM.RegisterRd = ID/EX.RegisterRs
   **True/False: _____**

## Question 2 [8]

**CPI of different instruction types is given as below**

| Arithmetic | Load/Store | Branch |
|---|---|---|
| 2 | 5 | 3 |

**Assume the following instruction breakdown for a given program**

| | No. of instructions |
|---|---|
| Arithmatic | $400 \times 10^6$ |
| Load/Store | $300 \times 10^6$ |
| Branch | $300 \times 10^6$ |

a. **What is the execution time for this program if the clock rate is 5 GHz**

b. **Suppose new, more powerful arithmetic instructions are added to the instruction set so the number of arithmetic instructions in our program are reduced by 25% but clock cycle time is increased by 10%. State whether this new design is better or not?**

## Question 3 [8]

**Consider the following instruction sequence:**

**add R1,R2,R3 // (R1←R2+R3)**

**lw R3,100(R1) // (R3←Mem[100+R1])**

**add R4,R4,R3 // (R4←R4+R3)**

**sub R2,R4,R2 // (R2←R4-R2)**

**sw R2,20(R2) // (R2→Mem[20+R2])**

You are required to write the final values of registers (R1, R2, R3 and R4) in the following table by considering the following situations for 5-staged MIPS pipeline.

1) Without forwarding & without hazard detection unit (column 3).
2) With full forwarding & without hazard detection unit (column 4)

Assume that all memory is initialized to 1. The hazard detection unit is capable of generating appropriate number of stalls in the cases of forwarding and without forwarding to resolve hazards. The absence of hazard detection unit suggests that the processor will not be able to generate any stalls.

| 1 | 2 | 3 | 4 |
|---|---|---|---|
| **Register Name** | **Initial Values** | **1) Without forwarding & without hazard detection unit** | **2) With full forwarding but without hazard detection unit** |
| **R1** | **0** | | |
| **R2** | **5** | | |
| **R3** | **10** | | |
| **R4** | **15** | | |