CSE/PC/B/T/226

BCSE 2nd Year 2nd Semester 2021 ONLINE Examination

Computer Architecture

Time: 11:00 AM to 2:00 PM (**3 hours**) **Full Marks: 70**

Instructions: Write your answers on plain paper in your own handwriting. Scan your answer script or take a snapshot. Send the resulting image file by Email to

asinha180314@gmail.com

GROUP-A

Answer 28 questions **28 × 2 = 56**

Choose the unique correct answer

1. Pipelining is efficient because

(a) it executes each instruction in less time
(b) it employs multiple execution units
(c) it overlaps the execution of several instructions
(d) it replaces the instructions in the user program by special internal instructions

For the rest of this Question Paper:

Unless otherwise mentioned, we will assume a 5-stage instruction pipeline: IF, ID, EX, MEM, WB.

2. A pipelined processor doesn't have separate instruction and data memories. i1, a load instruction, is initiated (occupies IF stage) in cycle-1. Subsequent instructions i2, i3, i4, i5 are initiated in cycles 2,3,4,5 respectively.

There is a hazard involving i1 and i4 in

(a) cycle-4 because both access memory
(b) cycle-5 because both access registers

For Q3-Q5

Consider the pipelined execution of the following program fragment:

```
------------------------
DADD   R1, R2, R3
DSUB   R4, R1, R5
AND    R6, R1, R7
OR     R8, R1, R9
XOR    R10, R1, R11
----------------------------
```

DADD is initiated (occupies IF stage) in cycle-1 and successive instructions are initiated in successive cycles.

3. Instruction DSUB
(a) gets the correct value of R1
(b) may not get the correct value of R1

4. Instruction AND
(a) gets the correct value of R1
(b) may not get the correct value of R1

5. Instruction OR
(a) gets the correct value of R1
(b) may not get the correct value of R1

For Q6-Q7

Consider the following program fragment:

```
----------------------
LD     R1, 0(R2)
DSUB   R4, R1, R5
AND    R6, R1, R7
OR     R8, R1, R9
-----------------------
```

The instructions are initiated in cycles 1, 2, 3, and 4 respectively.

6. The data hazard involving LD and DSUB
(a) can be resolved by forwarding
(b) cannot be resolved by forwarding

7. Suppose a branch instruction is initiated in cycle-1. The earliest cycle in which its successor can perform the EX stage is
(a) cycle-4
(b) cycle-5
(c) cycle-6
(d) cycle-7

For Q8-Q10:

The latencies of FP operations are given in the following table:

Instruction producing result	Instruction using result	Latency in clock cycles
FP alu op	Another FP alu op	3
FP alu op	Store double	2
Load double	FP alu op	1
Load double	Store double	0

The last column is the number of intervening clock cycles needed to avoid a stall.

Now consider the following MIPS code:

```
---------------------------------------------
Loop:  L.D         F0, 0(R1)
       ADD.D       F4, F0, F2
       S.D         F4, 0(R1)
       DADDUI      R1, R1, #-8
       BNE         R1, R2, Loop
-----------------------------------------
```

8. The number of stalls between L.D and ADD.D is

(a) 1
(b) 2
(c) 3
(d) 4

9.The number of stalls between ADD.D and S.D is

(a) 1
(b) 2
(c) 3
(d) 4

10. The number of stalls between DADDUI and BNE is

(a) 1
(b) 2
(c) 3
(d) 4

For Q11-Q12

Suppose the DADDUI is moved up and and placed between L.D and ADD.D.

11. A change of operands is required in the following instruction:

(a) ADD.D
(b) S.D
(c) DADDUI
(d) all of the above

12. In this scheduled code, the number of stalls between L.D and DADDUI is
(a) 1
(b) 2
(c) 3
(d) none of the above

13.Massively parallel computers are usually
(a) multiprocessors
(b) multicomputers

14. Message passing achieves
(a) communication
(b) synchronization
(c) both communication and synchronization
(d) none of the above

15. There is no need to physically move data between communicating processes in
(a) multiprocessors
(b) multicomputers

16. In a distributed shared memory system
(a) direct access to the local memory of a remote processor is prohibited
(b) any processor can access the local memory of any other processor

17. In a COMA machine, remote cache access is achieved through
(a) distributed cache directories
(b) snoopy protocols
(c) CSMA/CD protocol
(d) irrelevant since it is prohibited

18. In the Stanford Dash architecture, the directory keeps track of
(a) intra-cluster communication
(b) remote nodes that have a copy of each memory block
(c) the configuration of the interconnection network
(d) none of the above

19. A multistage network is a
(a) shared path network
(b) switching network

20. In a single shared-bus multiprocessor, the bus exchange lines
(a) contain bus request lines
(b) contain data lines
(c) contain address lines
(d) contain interrupt lines

21. In a centralized arbitration scheme for a single bus, a requesting master
(a) keeps its request line active until the bus transaction is over
(b) deactivates its request line when the arbiter activates its grant line
(c) generates a pulse train on the request line while the bus transaction is going on
(d) none of the above

22. In the daisy-chained grant scheme for bus arbitration, when a master receives an active grant line but does not require the bus, it
(a) generates a bus trap
(b) activates the common bus request line
(c) activates its output grant line
(d) none of the above.

23. In the MSI protocol for cache coherence, if a cache is in the shared (S) state, a PrRd request causes the state to change to the state
(a) I
(b) M
(c) S (unchanged)
(d) none of the above

24. In the MSI protocol, if a cache is in the shared (S) state and observes a Bus Read Exclusive (BusRdX) transaction, it changes its state to
(a) I
(b) M
(c) S (unchanged)
(d) none of the above

25. In the MSI protocol, if a bus is in the modified (M) state and observes a BusRd transaction, it
(a) remains in the M state
(b) changes to the I state
(c) flushes its data onto the bus and changes to the S state
(d) none of the above

26. In the MESI protocol, suppose a cache is in the I state and receives a PrRd request. The state of the cache changes to
(a) E, if the ‘shared’ signal is inactive
(b) S, if the ‘shared’ signal is active
(c) both (a) and (b)
(d) M

27. On receiving a PrWr request in the MESI protocol, the cache changes its state to M if the current state is
(a) E
(b) S
(c) I
(d) all of the above

28. In directory-based cache coherence, if a node incurs a cache miss, it
(a) sends a read request to the owner node
(b) communicates with the directory entry for the block
(c) executes (b) followed by (a)
(d) executes (a) followed by (b)

For Q29-Q30:

Consider the following code fragment for the MIPS floating-point unit
Using Dynamic Scheduling (operands i, j, k where i is the result).

```
----------------------------------
1. L.D          F6, 34(R2)
2. L.D          F2, 45(R3)
3. MUL.D        F0, F2, F4
4. SUB.D        F8, F2, F6
5. DIV.D        F10, F0, F6
6. ADD.D        F6, F8, F2
-----------------------------------
```

Instruction-1 has completed and its result is available on the CDB. Instructions 2-6 have been issued not completed.

29. For the ADD.D instruction, Qj is
(a) L.D F2, 45(R3)
(b) MUL.D F0, F2, F4
(c) SUB.D F8,F2, F6
(d) none of the above

30. For the MUL.D instruction, Vj is
(a) L.D F2, 45(R3)
(b) Regs[F2]
(c) Mem[45 + Regs[R3]]
(d) none of the above

31. WAR dependencies
(a) are false dependencies
(b) can be eliminated by register renaming
(c) both (a) and (b)
(d) require stalls to prevent hazards

For Q32-Q33:

Consider the following program for the CRAY X-MP:

```
----------------------
A1      53
VL      A1
V4      V2 v+ V3
-----------------------
```

32. The first result emerges after
(a) 3 cycles
(b) 4 cycles
(c) 5 cycles
(d) 6 cycles

33. The destination register V4 becomes available after
(a) 53 cycles
(b) 56 cycles
(c) 61 cycles
(d) none of the above

34. On the CRAY X-MP, an instruction is decoded in the
(a) NIP
(b) CIP
(c) LIP
(d) none of the above

35. On the CRAY X-MP, the length of a Vector Register is
(a) 64 words
(b) 22 words
(c) 24 words
(d) 32 words

GROUP-B

36. A pipelined processor has five stages: fetch, decode, register-read, execute, and write results into registers.

Consider the following program fragment:

```
------------------------------------------------------------------------------------
ADD     r1, r4, r7        ; r1 := r4 + r7
BEQ     r2, #0, r1        ; jump to the address specified in r2 if r1 = 0
SUB     r8, r10, r11
MUL     r12, r13, r14
-------------------------------------------------------------------------------------
```

Assume that the branch is **NOT TAKEN**.

What is the execution time (in cycles) ? **14**

-- END --