107C101

國立臺北科技大學107學年度碩士班招生考試

系所組別：2300資訊工程系碩士班

第一節　計算機概論　試題

第一頁　共四頁

注意事項：
1. 本試題共13題，每題配分如題目所列，共100分。
2. 請標明大題、子題編號作答，不必抄題。
3. 全部答案均須在答案卷之答案欄內作答，否則不予計分。

1. (12 pts) Please answer the following questions shortly and concisely.

   (1) (3 pts) Let R and S be two relations. Please list the statements which are **TRUE** among the following statements. (Note: You must correctly list all the true statements to get the credits. There is no partial credit on this problem.)

   (A) If R and S are transitive, the $R \cup S$ is transitive.

   (B) If R is transitive, the R^{-1} is transitive, where R^{-1} is the inverse of R.

   (C) If R and S are symmetric, then $R \cap S$ is symmetric.

   (D) If R and S are anti-symmetric, then $R \cup S$ is anti-symmetric.

   (E) If R is anti-symmetric, then R^{-1} is anti-symmetric.

   (2) (3 pts) Please convert the following expression in an infix form into a postfix form.

   $$6 + 5 * 3 - 7 - 8 + 9 * 2$$

   (3) (3 pts) Suppose that we use bottom-up heap construction to build up a *minimum heap* with an array of keys initially shown as below.

16	24	8	15	4	30	11	33	7	40	27	18	42	22	3

   After the construction, what is the key value at the last position of the array?

   (4) (3 pts) Please give a red-black tree that is not an AVL tree and explain why.

2. (6 pts) It is assumed that $T(1) = d$ for some constant d. State, using the "big oh" notation, the solution to each of the following two recurrences. Just state the answer - you do NOT need to justify them.

   (1) (3 pts) $T(n) = 4T(\frac{n}{2}) + n^2\sqrt{n}$

   (2) (3 pts) $T(n) = 10T(\frac{n}{3}) + 5n^2$

3. (6 pts) Mark by T(=true) or F(=false) each of the following:

   (1) (3 pts) If someone was able to give an exponential time lower bound for a problem that is NP-complete, then this would imply that $\mathcal{P}$ is not equal to $\mathcal{NP}$, where $\mathcal{P}$ is the set of all decision problems solvable by deterministic algorithms in polynomial time and $\mathcal{NP}$ is the set of all decision problems solvable by nondeterministic algorithm in polynomial time.

   (2) (3 pts) Suppose problem P_1 can be reduced to problem P_2 in linear time. Then, if P_1 is NP-hard then P_2 is NP-hard.

4. (6 pts) Consider the *directed acyclic graph* (DAG) G in Figure 1. Suppose we use the DFS-based approach to find a *topological order* starting with node B. Since we use DFS on the DAG, we can have a DFS-tree T rooted at B. Mark by **True** or **False** each of the following statements:

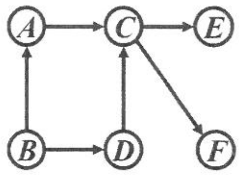

Figure 1. Directed acyclic graph (DAG) G in problem 4

   (1) (2 pts) There are five possible resulting topological sorts.

   (2) (2 pts) There is no forward edge in T by observation. So, a DAG cannot have forward edges.

   (3) (2 pts) There is only one cross edge in T.

5. (10 pts) Please answer the following questions shortly and concisely.

   (1) (2 pts) Given an IPv6 Link Local Address (LLA) FE80::0288:ccff:fe12:3456 of a NIC, what is the NIC's 48-bit MAC address corresponding to this LLA?

   (2) (3 pts) Why is a three-way handshake required for TCP connection establishment?

   (3) (5 pts) What is the standard protocol(s) applied by the following Internet applications respectively?

   (a)Email service　(b)Web service　(c)File transfer　(d)Network management

6. (10 pts) Please answer the following questions.

   (1) (2 pts) What is the standard basis for ZigBee and Bluetooth transmission technology respectively?

   (2) (4 pts) Explain the terms: DDoS (Distributed Denial-of-Service) and DDNS (Dynamic DNS).

   (3) (4 pts) What is the medium access control (MAC) method adopted by IEEE 8.2.11 wireless local area networks (WLAN) technology? How does the Network Allocation Vector (NAV) mechanism in Wi-Fi work? Please explain it.

注意：背面尚有試題

7. (6 pts) For the following questions regarding *process management*, please indicate whether each statement is true or false. If a statement is incorrect, please explain the reasons. (not just correcting the errors)

    (1) (2 pts) Shortest job first (SJF) scheduling algorithm will not result in starvation.

    (2) (2 pts) Multithreaded programs can always provide better performance than a single-threaded solution.

    (3) (2 pts) A multithreaded program using multiple user-level threads achieves better performance on a multiprocessor system than on a single-processor system.

8. (4 pts) Among the following statements about *memory management*, please indicate whether each statement is true or false. If a statement is incorrect, please explain the reasons. (not just correcting the errors)

    (1) (2 pts) Pure paging requires less memory overhead than pure segmentation to maintain the address translation structures.

    (2) (2 pts) Pure segmentation has the problem of internal fragmentation, since unused memory is internal to a partition.

9. (8 pts) Answer the following questions regarding *demand-paging systems*:

    (1) (4 pts) What is *page fault*? What are the steps in handling a page fault?

    (2) (4 pts) Consider the following page reference string:

    7, 2, 3, 1, 2, 5, 3, 4, 6, 7, 7, 1, 0, 5, 4, 6, 2, 3, 0, 1

    Assuming *demand paging* with three memory frames, how many page faults would occur for the LRU (Least Recently Used) page replacement algorithms? Please show the process in details.

    [Note that the memory frames are initially empty, so the first access to any page will cause a page fault.]

10. (4 pts) Regarding the following descriptions about *file management*, please indicate whether each statement is true or false. If a statement is incorrect, please explain the reasons. (not just correcting the errors)

    (1) (2 pts) The contiguous allocation algorithm suffers from the problems of internal fragmentation and size declaration.

    (2) (2 pts) The benefit of indexed allocation such as inode in UNIX operating system is that: direct access to blocks in a file, without suffering from external fragmentation.

11. (3 pts) Answer the following questions regarding *process coordination*.

    What is the meaning of the term *busy waiting*? Can busy waiting be avoided altogether? Why spinlocks are not appropriate for single-processor systems yet are often used in multiprocessor systems. Please explain your answers.

12. (15 pts) Given a processor implemented by a single-cycle control and datapath shown in Figure 2. Assume that the functional blocks adopted to implement the datapath have the following latencies in Table 1, and we need to design a processor that can support the MIPS instructions listed in Table 2:

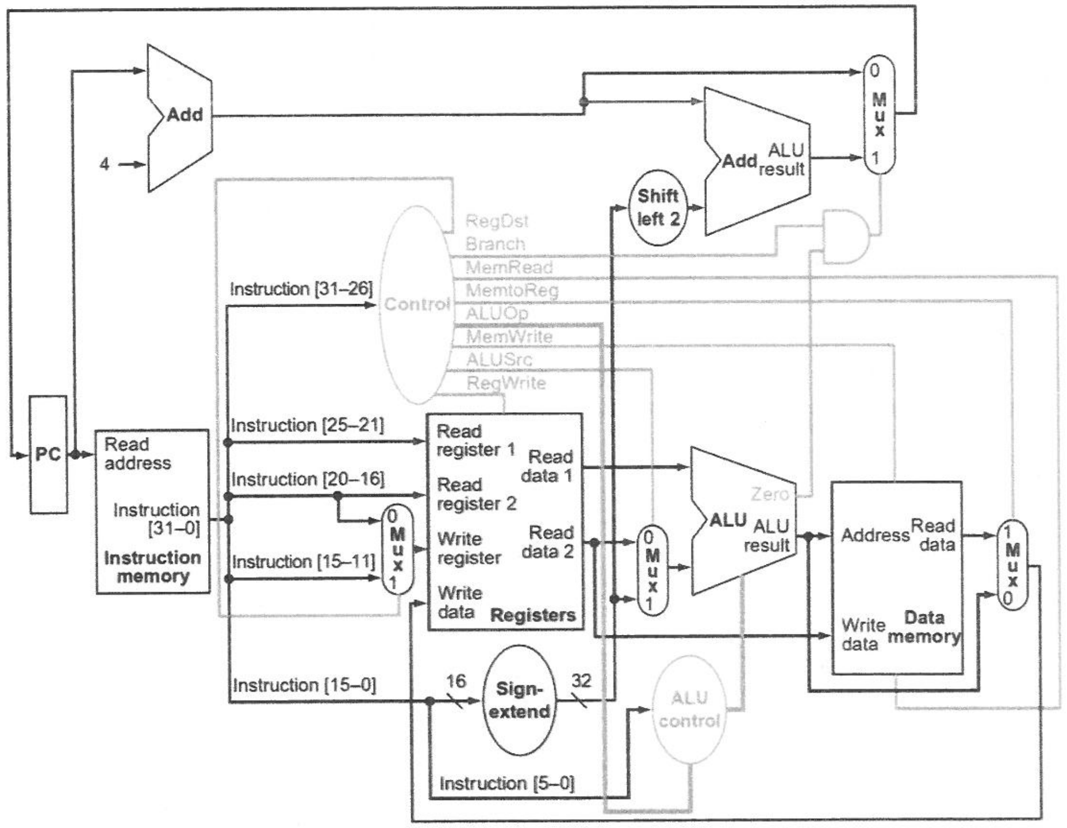

Figure 2. The single-cycle datapath with the control unit

Table 1. Latencies of the functional units on the datapath

PC's Clk.-to-Q	Inst.-Mem.	Add	Mux	ALU	Ctrl. Unit	ALU-Ctrl.	Regs. Access	Data-Mem.	Sign-extend	AND-Gate	Shift-left-2
0ps	600ps	200ps	50ps	250ps	150ps	50ps	300ps	500ps	100ps	50ps	30ps

Table 2. MIPS Instruction Opcode & Funct fields

Instr.	Opcode	Funct
add	0_{10}	32_{10}
sub	0_{10}	34_{10}
beq	4_{10}	
lw	35_{10}	
sw	43_{10}	

(1) (4 pts) If this processor only need to support **add** and **beq** instructions, what is the **clock cycle time**? (Hint: You should analyze the critical path of the required instructions)

(2) (4 pts) What is the **clock cycle time** of this processor that support the instructions listed in Table 2?

Next, the following two problems refer to a clock cycle in which the above processor fetches the following **binary instruction word**:

00010001000100010000000000011001

Assume that **data memory is all zeros** and **that the processor's registers have the following values** at the beginning of the cycle in which the above instruction word is fetched (Please refer to the MIPS Instruction Opcode & Funct fields listed in Table 2.):

r3	*r4*	*r8*	*r9*	*r10*	*r11*	*r12*	*r16*	*r17*	*r18*	*r19*	*r20*	*r31*
4	2	6	5	5	6	6	5	6	4	5	8	7

(3) (4 pts) What are the **control signal values** of the **ALUOp** (2 bits) and **ALU-Control** (4-bits) for this instruction?

(4) (3 pts) What is the **new PC address** after this instruction is executed. Assume that the value of the current PC address is 10000 (in decimal).

13. (10 pts) A typical MIPS processor with a five-stage pipeline datapath as illustrated in Figure 3, including:

(1) **IF**-Instruction fetch

(2) **ID**-Instruction decode and register fetch

(3) **EX**-Execution or calculate effective address

(4) **MEM**-Access data memory

(5) **WB**-Write back to registers

In the following problems, we assume that the following MIPS codes is executed on the above-mentioned 5-stage pipelined processor, a **predict-taken branch predictor with a branch target buffer**, and there are **no delay slots**:

Instruction sequence	
i1	***add r1, r2, r3***
i2	***lw r2, 0(r1)***
i3	***beq r2,r1, Label*** ***# not-taken***
i4	***add r3,r2,r1***
i5	***beq r2,r1, Label*** ***# taken***
i6	***lw r2,0(r3)***
i7	***Label: lw r1,0(r2)***

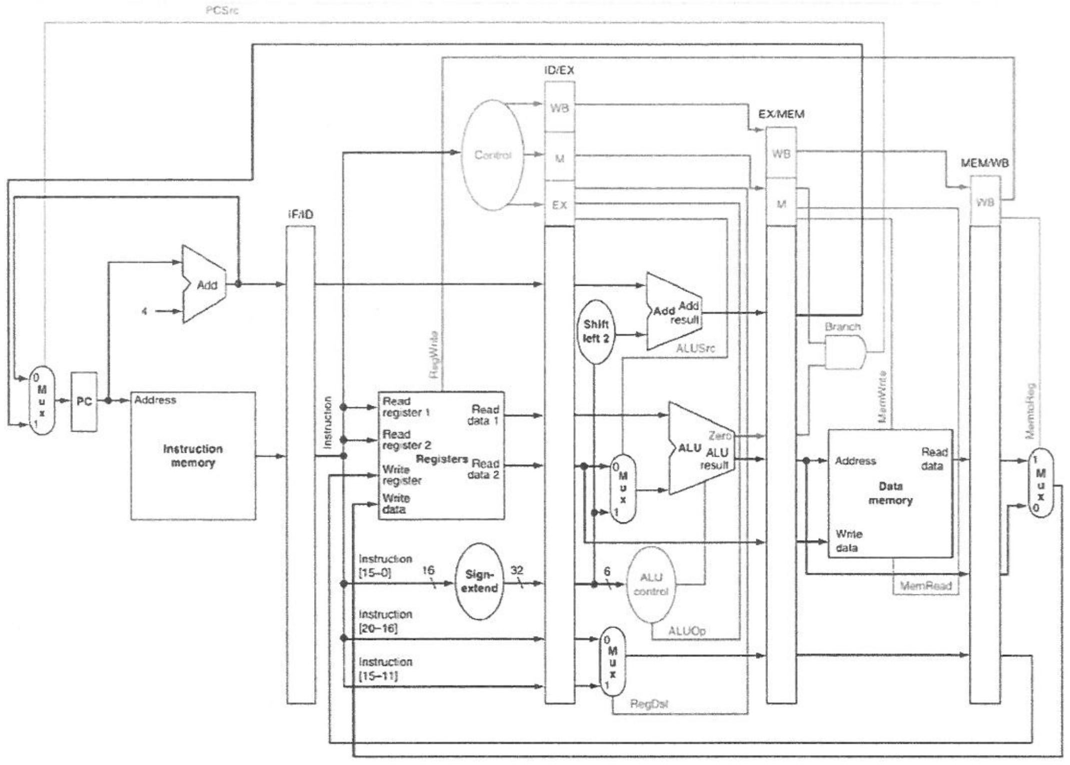

Figure 3. Typical MIPS pipelined datapath and control signals

(1) (5 pts) Assuming that we have **data hazard detection and forwarding units** to perform full forwarding (i.e. forward all results that can be forwarded), and **assuming that branches are decided in MEM stage (as** Figure 3 **depicts).** Due to data hazards and control hazards, **how many cycles will it take to complete the above MIPS codes?**

(2) (5 pts) Now assume that we optimize the datapath for branch by **moving branch execution into the ID stage** (as Figure 4 depicts), and we also make additions to the **forwarding and hazard detection units** to **forward** to or **stall** the branch at the **ID stage** in case the branch decision depends on an earlier result. Due to data hazards and control hazards, **how many cycles will it take to complete the MIPS codes**?

注意：背面尚有試題

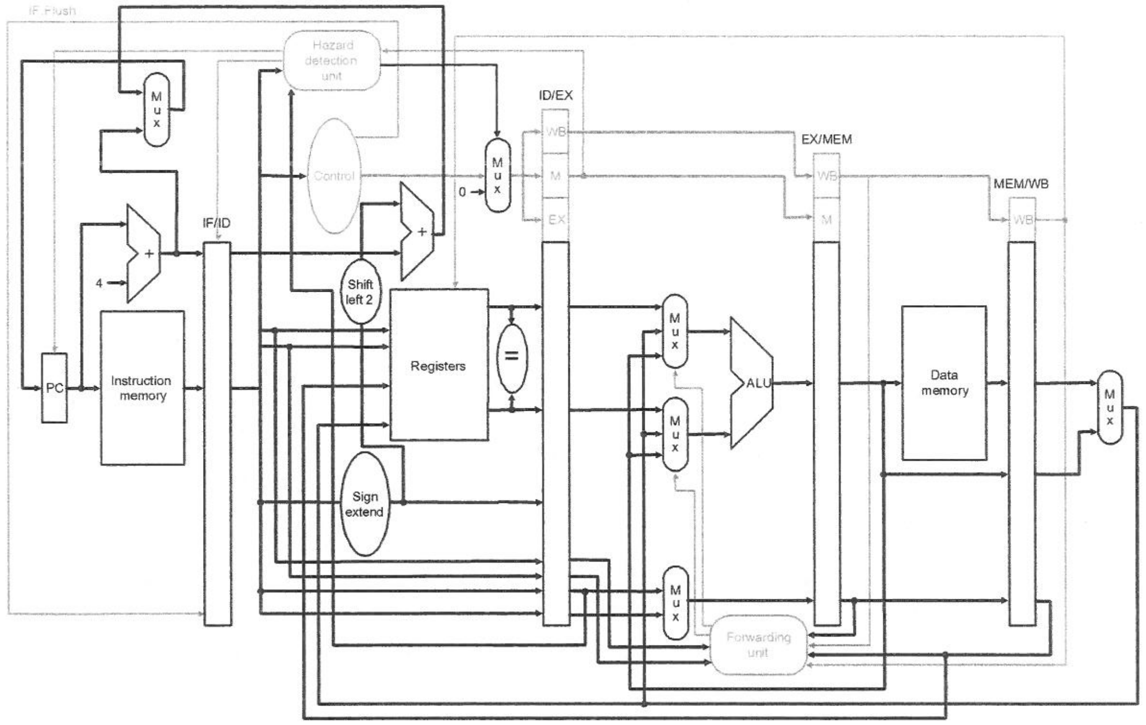

Figure 4. The optimized datapath for Branch