※ 注意：請用 2B 鉛筆作答於答案卡，並先詳閱答案卡上之「畫記說明」。

答題注意事項 / Instructions

1) 考題分為「單選題」與「複選題」，各題配分不完全相同，分別標示。
There are two types of questions: **Multiple Choices** and **Multiple Choices with at Least One Correction Choice** --- each question has its individual score.

2) **單選題**：只有一個選項為正確答案。不倒扣。
Multiple Choices: Only one among the four or five choices is the correct answer. No penalty for incorrect answer.

3) **複選題**：至少有一個選項為正確答案。每一個選項分別計分。整題空白，則該題零分。
Multiple Choices with at Least One Correction Choice: At least one among the four or five choices is the correct answer. Each choice is graded individually. No penalty for incorrect selection. Zero point for not selecting any choice.

見背面

單選題：只有一個選項為正確答案。不倒扣。

Multiple Choices: Only one among the four or five choices is the correct answer. No penalty for incorrect answer.

Q1 (2pts) As shown below, the function "compute_distance" runs through a list of 2D points and computes the distance between each pair of them. Note that each point is represented by single precision float-point numbers. Please answer the following questions.

```
void compute_distance(float x[], float y[], int N, float distance[]) {
    for (int i = 0; i < N; i++) {
        for (int j = 0; j < N; j++) {
            float x_dist = x[i]- x[j];
            float y_dist = y[i] - y[j];
            distance[j + i * N] = sqrt(x_dist*x_dist + y_dist*y_dist);
        }
    }
}
```

Let us observe the data access pattern first. For each iteration in the inner loop, the function loads 4 floating-point numbers and store 1 floating-point number. Assume N=1024, and the processor cache is too small to hold all the data points. The cache is configured with a write-back/write-allocate policy, with 32-bytes blocks. What would be the average cache miss rate?

(A) Less than 1%.
(B) Between 1% and 5%.
(C) Between 5% and 10%.
(D) Between 10% and 20%.
(E) More than 20%.

接次頁

Q2 (2pts) Following the previous question, let us calculate the *arithmetic intensity*, which is defined as the ratio between the number of executed operations and the number of bytes transferred between the processor and the memory. For the compute_distance function, we focus on floating-point operations only. What is the arithmetic intensity of this function?

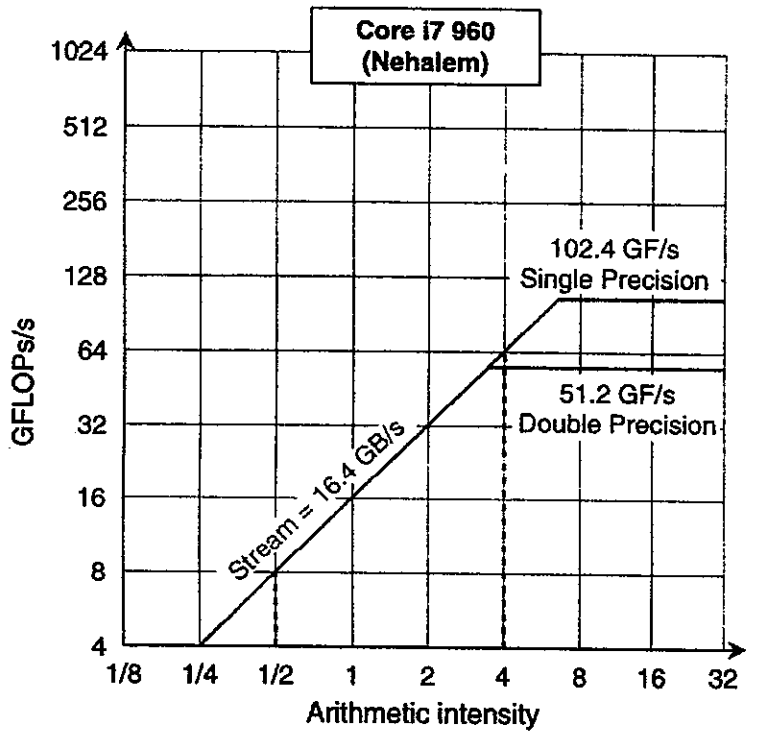
(A) Less than 0.35.
(B) Between 0.35 and 0.7.
(C) Between 0.7 and 1.
(D) Between 1 and 2.
(E) More than 2.

Q3 (2pts) Following the previous question, suppose we make the processor large enough to store all the 2D points, so that all the x[] and y[] are already in the cache before the function is called. What is the arithmetic intensity of this function in this case?

(A) Less than 0.35.
(B) Between 0.35 and 0.7.
(C) Between 0.7 and 1.
(D) Between 1 and 2.
(E) More than 2.

見背面

Q4 (2pts) Suppose the compute_distance function above is executed on an Intel Core i7 960 processor. The roofline model for the processor is shown as the figure below. The processor has 4 cores, and the capacity of Level 1 data cache on each core is 32KB. Assume N=1024. Based on the roofline model, how long does it take to finish the function when the processor cache is too small to store all the data points?



(A) Less than 1,000 seconds.
(B) Between 1,000 and 10,000 seconds.
(C) Between 10,000 and 100,000 seconds.
(D) Between 100,000 and 1,000,000 seconds.
(E) More than 1,000,000 seconds.

Q5 (2pts) Following the previous question, if we wish to improve the performance, which of the following method is most effective?

(A) Increase the associativity of the cache.
(B) Increase the memory bandwidth.
(C) Enable data prefetch.
(D) Make the processor pipeline deeper
(E) Enable multithreading.

接次頁

Q6 (6pts) Consider two hardware threads, Proc0 and Proc1, that interact with a shared memory that maps addresses to values. Both memory locations, A and B, have an initial value of 0. x0 and x1 are CPU registers. Each of the hardware threads has a store buffer. The hardware threads support out-of-order execution. Assume the following code in two cases runs on Proc0 and Proc1 concurrently. Which of the following is true about the final state of x0 and x1 after the code finishes executing?

Case 0		Case 1		Case 2	
Proc0	Proc1	Proc0	Proc1	Proc0	Proc1
store #1, [A];	store #1, [B];	load x0, [A];	load x1, [B];	store #1, [A];	load x0, [B];
load x0, [B];	load x1, [A];	store #1, [B];	store #1, [A];	store #1, [B];	load x1, [A];

(A) In case 0, x0 on Proc0 and x1 on Proc1 can be either both 0 or 1.
(B) In case 0, suppose the threads support total store order (TSO); x0 on Proc0 and x1 on Proc1 can never be both 0.
(C) In case 1, suppose the threads support TSO; x0 on Proc0 and x1 on Proc1 can be both 1.
(D) In case 2, suppose the threads support TSO; x0 and x1on Proc1 can be 1 and 0, respectively.

Q7 (5pts) Consider the following instruction sequence. Assume that the instructions are executed on a pipeline datapath with five stages.

```
or x10, x3, x4
ld x5, 8(x10)
ld x4, 0(x2)
add x5, x10, x5
sd x5, 4(x10)
```

Assume that the processor has neither implemented hazard detection nor forwarding. How many NOP instructions should be added to ensure the execution is correct?

(A) 0
(B) 3
(C) 5
(D) 6
(E) 8

見背面

Q8 (5pts) Which of the following statement is true?

(A) Introducing support of a new complex instruction will improve performance for all programs.
(B) Reducing the number of registers in the ISA will decrease the instructions per program for all programs.
(C) Introducing support for instructions with shorter lengths (e.g., 16-bit instructions in RISC-V) in the ISA will improve all programs' performance.
(D) Increasing cache associativity will result in an increase in cache hit time for all programs.
(E) Reducing cache block size will result in a reduction of capacity misses.

複選題：至少有一個選項為正確答案。每一個選項分別計分，不倒扣。整題空白，則該題零分。
Multiple Choices with at Least One Correction Choice: At least one among the four or five choices is the correct answer. Each choice is graded individually. No penalty for incorrect selection. Zero point for not selecting any choice.

Q9 (8pts) Modern CPU architectures provide nested page tables (NPTs) to support virtualization. The MMU hardware performs two-stage memory translation. In the first stage, it translates virtual addresses (VAs) using the virtual machine (VM)'s page tables (PTs) to intermediate physical addresses (IPAs). The VM's PTs are specified in IPAs. In the second stage, the hardware translates the IPAs using the NPTs to physical addresses (PAs). All page tables are 4K bytes, and a page table entry is 8 bytes. All VAs and IPAs are 39 bits, and the PTs and NPTs include 2M or 4K byte mappings. Assume the TLB or cache is not implemented on the hardware. Considering both PTs and NPTs are enabled, which of the following could be the number of memory accesses taken for the hardware to perform the two-stage memory translation (e.g., VA of a VM to a PA)?

(A) 10
(B) 11
(C) 12
(D) 13

接次頁

Q10 (8pts) Which of the following are possible values stored in the memory location A after the following sequence of instructions has been executed, each on a different processor? Assume each processor has its own cache, memory location A initially contains value 0.

CPU0	CPU1	CPU2

store #4, (A);
load reg, (A);
add reg, reg, #4;
store reg, (A);
	load reg, (A);
	add reg, reg, #1;
	store reg, (A);
	load reg, (A);
	add reg, reg, reg;
	store reg, (A);
		store #6, (A);
		load reg, (A);
		add reg, reg, reg;
		store reg, (A);

(A) 8
(B) 9
(C) 10
(D) 16

見背面

Q11 (8pts) Assume the system has the following properties:

- The memory is byte-addressable, and memory accesses are to 1-byte words.
- Addresses are 13 bits wide.
- The cache is 4-way set associative, with 4-byte block size, and 8 sets.

Index	Tag	V	Bytes:0-3	Tag	V	Bytes:0-3	Tag	V	Bytes:0-3	Tag	V	Bytes:0-3
0	F0	1	ED 32 0A A2	8A	1	74 80 1D FC	14	1	EF 09 86 2A	BC	0	25 44 FA 1A
1	BC	0	03 3E CD 38	A0	0	16 7B ED 5A	BC	1	8E 74 DF 18	E4	1	FB B7 12 02
2	BC	1	54 9E 1E FA	B6	1	DC FA B2 14	00	0	B6 1F 7B 44	74	0	10 F5 B8 2E
3	BE	0	2F 7E 3D A8	C0	1	27 95 A4 74	C4	0	07 11 6B D8	BC	0	C7 B7 AF C2
4	7E	1	74 21 1C 2C	8A	1	22 C2 DC 34	BC	1	BA DD 37 D8	DC	0	E7 A2 39 BA
5	98	0	A9 76 DB EE	54	0	BC 91 D5 92	98	1	80 BA 9B F6	BC	1	48 16 81 0A
6	38	0	5D 4D F7 DA	BC	1	69 C2 DB 74	8A	1	A8 CE 7F DA	38	1	FA 93 EB 48
7	8A	1	04 2A 32 6A	9E	0	B1 86 56 0E	CC	1	96 30 47 F2	BC	1	F8 1D 42 DB

Suppose a program running on the machine references a 1-byte word at address addr. Select the true statements.

(A) Read from the addr:0x71A results in a cache hit. The return byte value is 0xDB.
(B) Read from the addr:0x16E8 results in a cache hit. The return byte value is 0x74.
(C) Read from the addr: 0x178B results in a cache hit. The return byte value is 0xFA.
(D) Read from the address 0xA73 results in a cache miss.

Q12 (5pts) Regarding allocating secondary storage space, select correct description(s).

(A) Contiguous allocation suffers from the problem of external fragmentation more than linked allocation and indexed allocation.
(B) Linked allocation is good if files are large and usually accessed randomly.
(C) Indexed allocation is good if files are large and usually accessed sequentially.
(D) File-allocation table (FAT) is one variation of linked allocation.
(E) If a linked list is used for free-space management, although traversing the whole list is not efficient, traversing the whole list is not a frequent action of an operating system.

接次頁

Q13 (5pts) Regarding security, select correct description(s).

(A) Recursively and continuously calling **fork()** is a denial of service (DoS) attack.
(B) A message authentication code (MAC) is computed by F(M, K), where F is a MAC-computation algorithm, M is a message, and K is a shared secret key. The message authentication code is able to verify the authenticity of a message sender.
(C) The message authentication code above is able to verify that a message has not been modified.
(D) The message authentication code above is able to protect against replay attacks.
(E) A buffer overflow may cause a code-injection attack.

Q14 (10pts) Consider two-level paging without virtual memory. Assume

- A system uses a W-bit logical address space.
- The page size is 2^X bytes.
- The size of an entry in the outer page table is 2^Y bytes.
- The size of an entry in the inner page table is 2^Z bytes.
- We also assume that the size of each entry in each page table is long enough to store all information needed.

Given the following (W, X, Y, Z), which can fit the outer page table into one page?

(A) (W, X , Y, Z) = (16, 8, 6, 6).
(B) (W, X , Y, Z) = (16, 12, 16, 8).
(C) (W, X , Y, Z) = (32, 16, 8, 4).
(D) (W, X , Y, Z) = (32, 24, 6, 10).
(E) (W, X , Y, Z) = (64, 32, 20, 20).

見背面

Q15 (10pts) Given 3 pages (P1, P2, P3) and 2 frames, the page reference has the following features:

- The frames are initially empty.
- There will be 4 page references.
- The first page reference is P1.
- At the time of a page reference, a page replacement algorithm only knows the probability of the page of the next page reference, and it does NOT know the exact page of the next page reference.
- After a page reference of P1, the next page reference is P2 with 0.8 probability or P3 with 0.2 probability.
- After a page reference of P2, the next page reference is P3 with 0.4 probability or P1 with 0.6 probability.
- After a page reference of P3, the next page reference is P1 with 0.9 probability or P2 with 0.1 probability.
- An optimal page replacement algorithm minimizes the expected number of page faults for the total 4 page references.

Regarding the expected number of page faults of the optimal page replacement algorithm, select correct description(s).

(A) E [Number of page faults for the 2nd page reference] = 1.
(B) $0.275 \leq$ E [Number of page faults for the 3rd page reference] ≤ 0.325.
(C) $0.300 \leq$ E [Number of page faults for the 3rd page reference] ≤ 0.350.
(D) $0.275 \leq$ E [Number of page faults for the 4th page reference] ≤ 0.325.
(E) $0.300 \leq$ E [Number of page faults for the 4th page reference] ≤ 0.350.

接次頁

Q16 (10pts) Given 2 periodic processes in the following table:

Process	Period (msec)	Deadline (msec)	Processing (Burst) Time (msec)	Softness
P1	T1	D1	C1	S1
P2	T2	D2	C2	S2

The softness is a positive integer, meaning that 1 deadline needs to be met for any Si consecutive instances of Pi. For example:

- If Si = 1, then each instance needs to meet its deadline.
- If Si = 2, then 1 deadline needs to be met for any 2 consecutive instances. (If an instance misses its deadline, then the next instance must meet its deadline.)

Note that, if Si > 1, then Pi does not need all instances to be processed. The real-time scheduling problem has the following features:

- For each i, $Ci \leq Di \leq Ti$.
- The time of context switching and interrupt handling can be ignored.
- The scheduling is preemptive.
- The scheduling is based on dynamic priority assignment.
- A scheduling algorithm is optimal in that if a problem cannot be scheduled by the scheduling algorithm, then it cannot be scheduled by any other scheduling algorithm.

Select correct description(s).

(A) If S1 = S2 = 1 and (C1 / T1) + (C2 / T2) > 1, then the problem must be non-schedulable.
(B) If S1 = S2 = 1 and (C1 / T1) + (C2 / T2) ≤ 1, then the problem must be schedulable.
(C) If S1 = 1 and S2 = 2 and (C1 / T1) + (C2 / T2) > 1, then the problem must be non-schedulable.
(D) If S1 = 1 and S2 = 2 and (C1 / T1) + (C2 / T2 / 2) ≤ 1, then the problem must be schedulable.
(E) If S1 = 1 and S2 = 2 and (C1 / T1) + (C2 / T2 / 2) ≤ 1, then an optimal scheduling algorithm must interleavingly (…, meet, miss, meet, miss, meet, miss, meet, …) meet and miss the deadlines of S2.

見背面

Q17 (10pts) A state is "safe" if the system can allocate resources to each thread in some order and still avoid a deadlock. A system is in a safe state only if there exists a "safe sequence" of threads. Given 7 threads (T0, T1, T2, T3, T4, T5, T6) and 3 resource types (R0, R1, R2), the follow tables show the current state of the system:

- Available: the number of currently-available resources of each type.
- Max: the maximum demand of each thread.
- Allocation: the number of resources of each type currently allocated to each thread.

Available

R0	R1	R2
3	0	6

	Max			Allocation		
	R0	R1	R2	R0	R1	R2
T0	0	3	0	0	1	0
T1	3	6	5	1	2	2
T2	2	1	5	1	1	0
T3	5	6	1	1	2	0
T4	6	1	0	2	1	0
T5	4	1	3	1	1	1
T6	0	6	8	0	0	0

How many safe sequences of threads are there? Note that there are 7! = 5,040 possible sequences. Assume that the answer is 1000W + 100X + 10Y + Z, where W, X, Y, Z are integers and $0 \le W, X, Y, Z \le 9$. Select correct description(s).

(A) $W + X + Y + Z = 9$.
(B) $W - 2X - 3Y + 4Z = 15$.
(C) $WX + YZ = 32$.
(D) $(W + Y + 1) / (X + Z + 1) = 0.5$.
(E) $W^2 + X^2 + Y^2 + Z^2 = 20$.

試題隨卷繳回