國立陽明交通大學
113學年度碩士班考試入學招生試題

科目：計算機系統(含作業系統及計算機組織)(8103)

系所班別：資訊聯招

考試日期：113年2月2日 第 3 節

【不可使用計算機】

＊作答前請先核對試題、答案卷(試卷)與准考證之所組別與考科是否相符！！

一、複選題(80%)，共二十題，每題全對得 4 分，答對一個選項得 1 分(答對兩個選項得 2 分，以此類推)，答錯一個選項倒扣 1 分(答錯兩個選項倒扣 2 分，最多扣至本科目計算機系統 0 分為止)，整題未作答不給分。(舉例每題有 a、b、c、d 四個選項，某題答案為 a、b，而作答時選 a、b、c，則三個選項正確一個錯誤，該題得 3−1=2 分)請用答案卡作答

1. Which of the following x86 instructions cannot be executed in Ring 3 privilege level?
   (a) MOV CR0, EAX.
   (b) HLT.
   (c) LGDT.
   (d) LIDT.

2. Which of the following x86 instructions are position-independent (i.e., the code can be executed at any memory address)?
   (a) MOV EAX, 1.
   (b) XOR EBX, EBX.
   (c) INT 0x80.
   (d) ADD EAX, 1.

3. Consider the following C program on x86_64 Linux.

```
01    #include <stdio.h>
02    #include <stdlib.h>
03    #include <sys/mman.h>
04    #include <unistd.h>
05
06    int main() {
07
08      size_t pagesize = sysconf(_SC_PAGESIZE);
09
10      void *buffer = aligned_alloc(pagesize, pagesize);
11      if (!buffer) {
12          perror("Failed to allocate memory");
13          return EXIT_FAILURE;
14      }
15
16
17      if (mprotect(buffer, pagesize, PROT_READ | PROT_WRITE | PROT_EXEC) == -1) {
18          perror("mprotect - error 1");
19          free(buffer);
20          return EXIT_FAILURE;
21      }
22
23      sprintf((char *)buffer, "This is a test.\n");
24
25      if (mprotect(buffer, pagesize, 0 ) == -1) {
```

```
26          perror("mprotect - error2");
27          free(buffer);
28          return EXIT_FAILURE;
29      }
30
31      printf("%s", (char *)buffer);
32      free(buffer);
33
34      return EXIT_SUCCESS;
35  }
```

Which of the following statement(s) are true?

(a) The program will output "This is a test." on the screen.
(b) The memory block pointed to by `buffer` will be allocated on the stack.
(c) The variable `pagesize` will be allocated on the stack.
(d) After a successful memory allocation, the value of the `buffer` pointer will be a multiple of `pagesize`.

4. Which statement(s) best describes the Earliest Deadline First (EDF) scheduling algorithm?

(a) EDF prioritizes tasks based on their computational complexity, scheduling the most complex tasks first.
(b) EDF is a preemptive scheduling algorithm that prioritizes tasks based on their proximity to a fixed deadline.
(c) EDF assigns priority to tasks based on the order of their arrival, with the earliest arrivals being processed first.
(d) EDF is a non-preemptive scheduling algorithm that schedules tasks in a round-robin fashion, regardless of their deadlines.

5. On x86_64 systems, the Linux system calls are made using the `syscall` instruction with the following calling conventions

- `RAX`: Contains the system call number.
- `RDI`, `RSI`, `RDX`, `R10`, `R8`, and `R9`: Are used to pass up to six arguments to the system call.
- `RCX` and `R11`: Are clobbered after the syscall instruction.
- Return Value: The return value of the system call is placed in the `RAX` register.

The implementation of the `sys_write` (`write`) system call in the Linux kernel is as follows.

/ fs / read_write.c

```
642
643             return ret;
644     }
645
646     SYSCALL_DEFINE3(write, unsigned int, fd, const char __user *, buf,
647                     size_t, count)
648     {
649             return ksys_write(fd, buf, count);
650     }
```

And, the man pages description for `sys_write` is as follows.

```
write(2)                    System Calls Manual                    write(2)

NAME      top

       write - write to a file descriptor

LIBRARY      top

       Standard C library (libc, -lc)

SYNOPSIS      top

       #include <unistd.h>

       ssize_t write(int fd, const void buf[.count], size_t count);

DESCRIPTION      top

       write() writes up to count bytes from the buffer starting at buf
       to the file referred to by the file descriptor fd.

       The number of bytes written may be less than count if, for
       example, there is insufficient space on the underlying physical
       medium, or the RLIMIT_FSIZE resource limit is encountered (see
       setrlimit(2)), or the call was interrupted by a signal handler
       after having written less than count bytes.  (See also pipe(7).)
```

Which of the following statement(s) is true for invoking the `sys_write` system call?

(a) Need to assign `fd` to RAX.
(b) Need to assign the address value of `buf` to `RSI`.
(c) The actual number of bytes written may be less than the number specified by `count`.
(d) User-level programs can use `syscall` to call any function in the kernel.

6. Which of the following statements on process are true?

(a) In UNIX, a new process is created by the fork() system call. The new process consists of a copy of the address space of the original process.
(b) A thread shares with other threads belonging to the same process its code section, data section, stack, and other operating-system resources, such as open files and signals.
(c) With using standard C library, user program can easily read or write a file from storage devices without the help from operating systems.
(d) When a context switch occurs, the kernel saves the context of the old process in its PCB and loads the saved context of the new process scheduled to run. The context includes the value of the CPU registers, the process state, and memory-management information.

7. Which of the following statements on I/O system are true?
   (a) NAND-flash based SSDs suffer from limited lifetime. The SSD controller runs a software layer to alleviate this issue by leveling writes without the help from the operating system.
   (b) When the CPU detects that a controller has asserted a signal on the interrupt-request line, the CPU performs a state save and jumps to the interrupt-handler routine at a fixed address in memory.
   (c) When DRAM generates an interrupt, CPU saves the state of the user code, switches to kernel mode, and dispatches to the kernel routine or thread that implements the requested service.
   (d) In Linux, the swap space is now used only for anonymous memory. That is, Linux will not swap out storage-backed files.

8. Which of the following statements on memory management are true?
   (a) When a CPU needs to access data in DRAM, the DRAM controller verifies the existence of the data. If the data is not found in the main memory, the DRAM controller generates a page fault.
   (b) In the system adopting paging, when a process is to be executed, its pages are loaded into any available memory frames from their source (a file system or the backing store).
   (c) In Linux and Solaris, the slab-allocation algorithm uses caches to store kernel objects. When a new object for a kernel data structure is needed, the allocator can assign any free object from the cache to satisfy the request.
   (d) On Linux which adopts paging, multiple processes can share one page table to save memory consumption.

9. Which of the following statements on virtual memory management are true?
   (a) After enabling virtual memory, CPUs operating in user mode can decide to access either physical or virtual addresses based on the compiler configurations.
   (b) If a required data is not in physical memory, the user process raises a page fault and runs the page fault handler to transfer the data from storage to memory, possibly relying on DMA for efficiency.
   (c) When a process forks a child by using Copy-on-Write (Cow), the process and its child share all memory pages until one of them writes to the page.
   (d) When a logical address is generated by an x86 CPU, the MMU first checks if its page number is present in the TLB. If not, CPU walks the page table to get the physical address without the help from the operating system.

10. Which of the following statements on file system are true?
   (a) A file control block (FCB) (an inode in UNIX file systems) contains information about the file, including ownership, permissions, and location of the file contents.
   (b) File-name extensions (e.g., .txt, .jpeg) on UNIX systems are necessary and critical, as they guide the operating system to run the right application (e.g., text editor or gallery) for opening files.
   (c) Multiple processes may be allowed to map the same file concurrently and to allow sharing of data. Writes by any of the processes modify the data in virtual memory and can be seen by all others that map the same section of the file.
   (d) A disk can be sliced into multiple partitions and each partition is not functionable without containing a file system. That is, each partition must contain one file system (e.g., NTFS, Ext4).

11. For the 32-bit MIPS ISA, an instruction is coded in 32-bit. Different addressing modes are defined in the ISA for instruction and data accessing. Which of the following statements about the MIPS instruction design are true?

    (a) The immediate addressing mode and the base addressing mode both use the I-type instruction format, while the PC-relative addressing mode does not.
    (b) There are 5 addressing modes in the MIPS ISA.
    (c) The immediate values in the pseudodirect addressing mode use word addresses.
    (d) A 32-bit constant that has non-zero upper and lower 16-bit values cannot be loaded into a register using only one instruction (excluding pseudoinstructions).

12. The following in-order pipelined five-stage MIPS processor design shown in Fig. 1 has a bug:

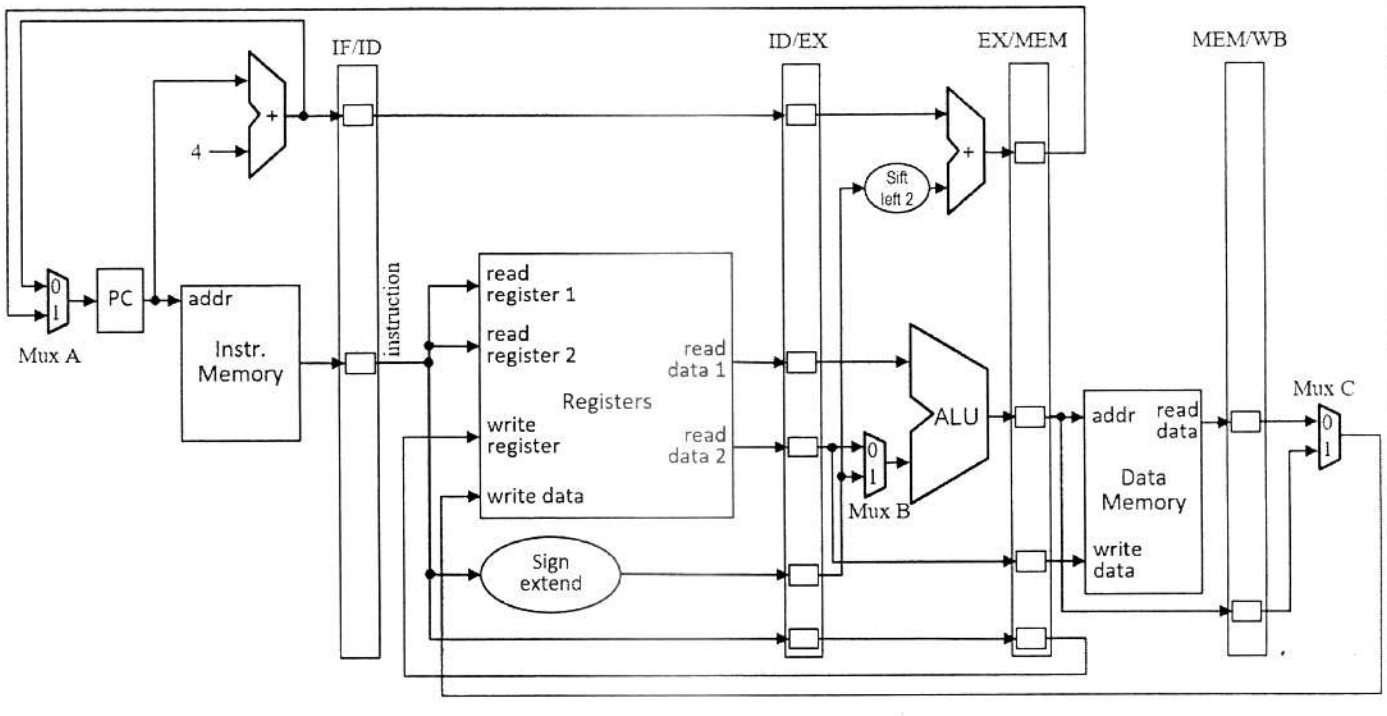


Fig. 1

Note that the processor has no forwarding paths, and if it detects data hazards at the decode stage (the hazard control unit is not shown), the pipeline stages will stall properly. If we use the buggy processor to execute the following code segment:

```
add   $4, $1, $2
or    $0, $1, $1
addi  $2, $2, 3
add   $3, $4, $3
add   $1, $4, $0
```

and assume that initially the registers \$0 ~ \$4 contains the values 0 ~ 4, respectively. That is, register \$0 is 0, register \$1 is 1, …, etc. Which of the following statements are true right after the writeback stage of the fourth instruction (`add $3, $4, $3`) is complete?

(a) Register \$0 has a value of 0.
(b) Register \$1 has a value of 7.
(c) Register \$1 has a value of 1.
(d) Register \$4 has a value of 4.

13. Assume that we have a classical five-stage pipelined 32-bit MIPS processor with a branch predictor and all required forwarding mechanisms. Which of the following statements are true?

(a) If a load instruction is immediately followed by a conditional branch that is on the load result, and assume that the branch condition is evaluated on the decode stage. One stall cycle is needed to execute the two instructions properly.
(b) If the 16-bit immediate value in a branch instruction is *imm*, the branch target address, is computed by PC = PC + 4 + sign-ext(*imm*). Note that sign-ext(*imm*) means sign extension of the parameter *imm* to 32-bit.
(c) For better performance, the branch target buffer that stores the destination PC for a branch is usually implemented as a tagged memory.
(d) A branch predictor using the information of both a local branch instruction and the global behavior of recently executed branches is called a correlating predictor.

14. Assume that we have two processors, P1 and P2, that are different microarchitecture implementations of the same ISA. The instructions can be divided into four classes according to their CPI (class A, B, C, and D). P1 has a clock cycle of 1ns and CPIs of 1, 1, 4, and 3, respectively for each instruction class. P2 has a clock cycle of 0.8ns and CPIs of 2, 2, 2, and 2, respectively for each instruction class. Given a program with its instructions divided into the four classes as follows: 30% class A, 25% class B, 20% class C, and 25% class D. Which of the following statements about the execution of this program using processor P1 and P2 are true?

(a) The average CPI of P1 is larger than that of P2.
(b) The execution time of P1 is larger than that of P2.
(c) If both processors run at the same clock rate, their CPIs do not change.
(d) For the processor that has longer execution time, it is possible to improve only one of its instruction classes so that it becomes faster.

15. Which of the following statements about the MIPS ISA design are true?

(a) Not all instructions using the I-type format perform sign-extension on the 16-bit immediate values.
(b) The jump instruction, `j`, can always jump from address A to address B as long as the offset between A and B is less than $\pm 2^{15}$ words.
(c) MIPS branch instructions are encoded using the I-type format that contains only two registers. This design is based on the principle of making the common case fast.
(d) If we add more R-type instructions to the ISA, and allow the compiler to make use of these extra instructions to compiled a program, then the program size in terms of the number of instructions will always be smaller than or equal to the size compiled using the original ISA.

16. Which of the following statements are true?

(a) AI accelerator chips require access to extensive data during their operation. Therefore, enhancing memory access and mitigating the impact of large memory requirements pose significant challenges in the design of AI accelerators.
(b) Conducting Large Language Model (LLM) inference at the edge of computing provides a direct enhancement to information security.
(c) High-performance GPUs are crucial for developing LLMs. Consequently, power optimization is no longer a concern in GPU design.
(d) The increase of a CPU's clock rate is constrained by CPU's power consumption, while a GPU, with its distinct architecture, avoids the limitation of clock rate imposed by power consumption.

17. Consider the CPU performance and the instruction count at the machine level, which of the following statements are true?

(a) When comparing two algorithms designed to solve the same problem, the one with higher complexity typically demands a greater number of instructions to complete the operation compared to the other.
(b) As we implement the same algorithm in Python and C programming languages, the Python program generally requires a greater number of instructions than the C program to execute the algorithm.
(c) The Millions of Instructions Per Second (MIPS) performance metric may not be equitable, as an instruction set with lower CPI can achieve a higher clock rate but necessitates a larger number of instructions.
(d) Google's warehouse-scale computer centers typically maintain server utilization levels ranging from 70% to 90%, ensuring the optimal utilization of server resources.

18. Which of the following statements are correct?

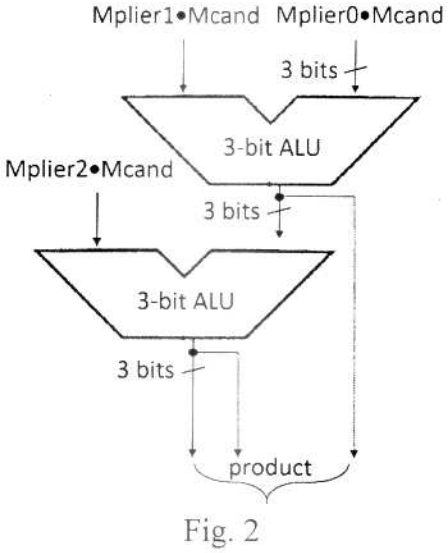


Fig. 2

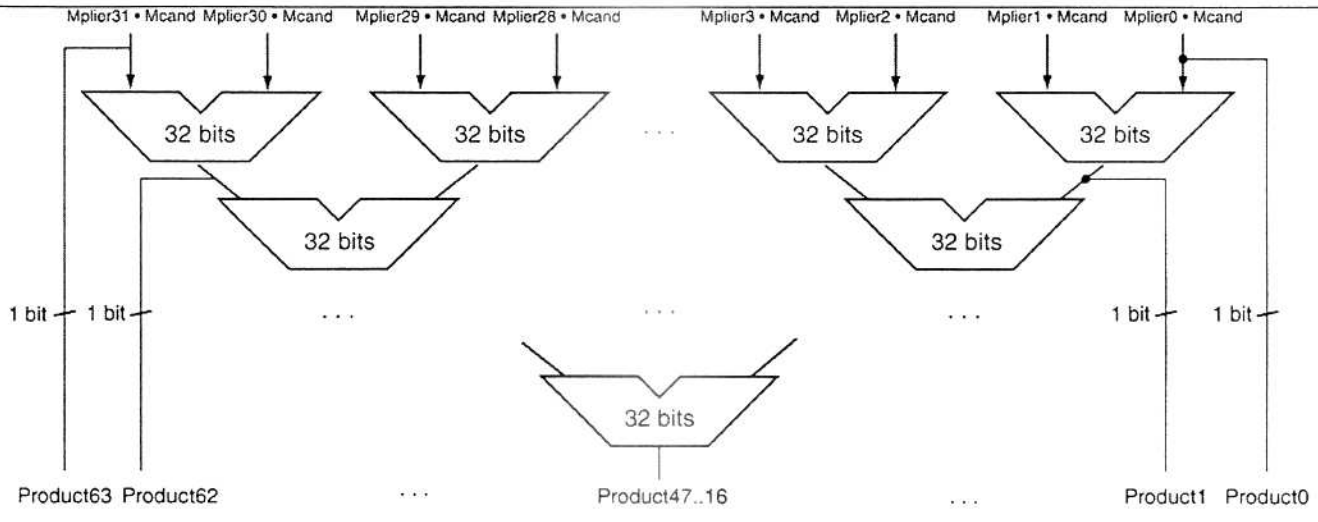


Fig. 3

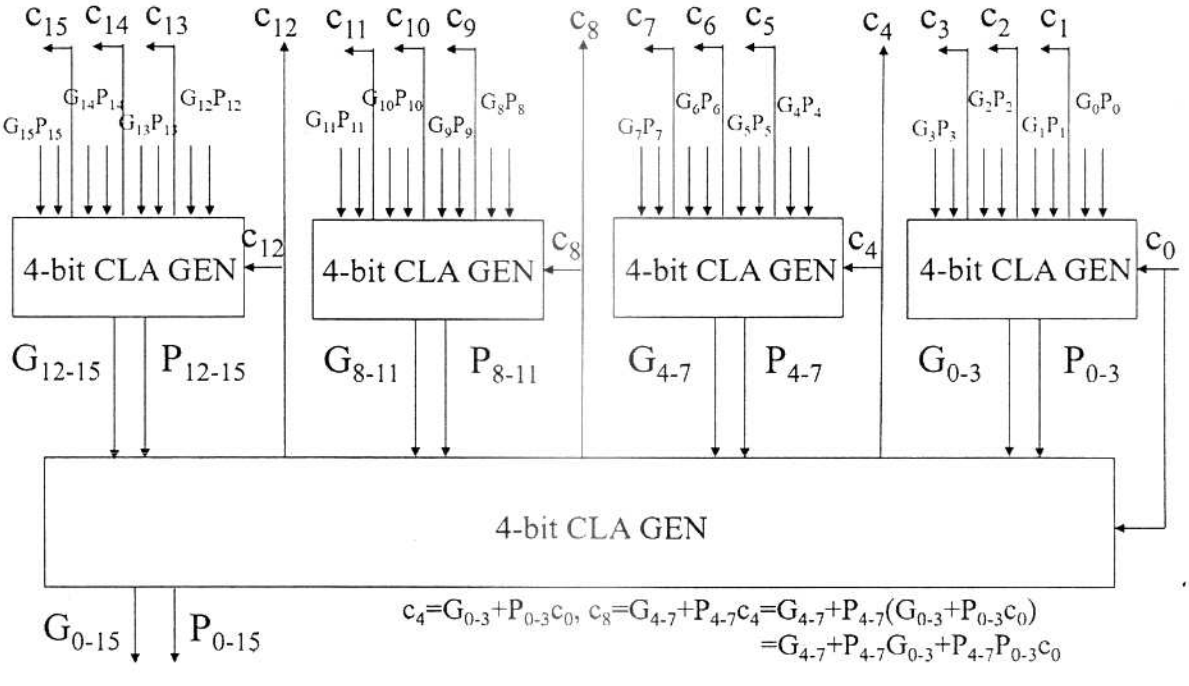


Fig. 4

(a) Fig. 2 shows a correct 3-bit×3-bit multiplier.

(b) Fig. 3 shows a wrong 32-bit×32-bit multiplier.

(c) The most significant bit of a carry look-ahead adder often experiences elevated input loading, leading to increased delays. Employing a multi-level carry look-ahead adder helps mitigate the issue of high input loading.

(d) Consider the 16-bit two-level carry look-ahead adder in Fig. 4 and the timing of transitioning to correct value on each signal line, c_1 transitions to correct signal before c_4 to c_{15}, $G_{(4i-4)-(4i-1)}$ and $P_{(4i-4)-(4i-1)}$ transition to correct values prior to c_{4i}, for $i = 1$ to 3, and c_{4i} transitions to correct value prior to c_{4i+1} to c_{4i+3}, for $i = 1$ to 3.

19. Which of the following statements about computer arithmetic are correct?

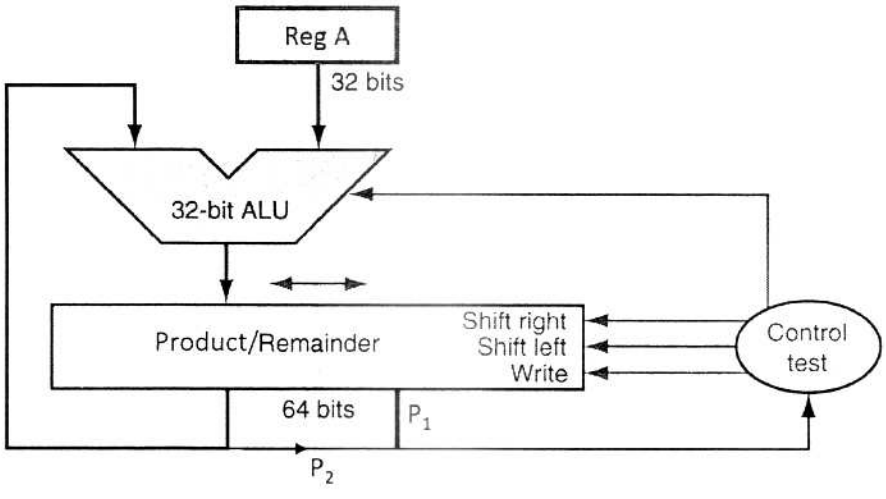


Fig. 5

(a) The hardware depicted in Fig. 5 above serves dual functions, functioning as both a multiplier and a divider. The Product/Remainder register necessitates a shift-right operation for multiplication and a shift-left operation for division. The control test unit receives its input from P2 when operating as a multiplier and from P1 when operating as a divider.

(b) For floating-point addition, aligning decimal points involves shifting the number with the smaller exponent to ensure proper alignment. This is because rounding the significand of the number with smaller exponent has a minor impact on accuracy. Conversely, if decimal points are aligned by shifting the number with the larger exponent, it may result in overflow on the significand of the number with the larger exponent.

(c) Addition is associative for both integers and floating-point numbers.

(d) Assuming the accuracy of a floating number is to the second decimal place and rounding half down with additional guard, round, and sticky bits. The ulp of the addition $5.01 \times 10^2 + 1.56 \times 10^5$ is 0.499? (note: round_half_down(x) = $\lceil x - 0.5 \rceil$).

20. The (7,4) Hamming code has bit indices ranging from 1 to 7 (left to right). Parity bits p_1, p_2, and p_3 are bits 1, 2, and 4, respectively. Data bits are bits 3, 5, 6, and 7. Which of the following statements are true?

(a) The word 1 0 1 0 1 1 1 is not correct, and bit 6 should be 0.

(b) Add an additional parity bit p_4 for the whole word and make it satisfy $p_1+p_2+d_1+p_3+d_2+d_3+d_4+p_4 = 0$ (p_4 even). This will increase the Hamming distance to 4.

(c) The word 0 1 0 0 0 1 0 1 is not correct. p_4 should be 0.

(d) The word 1 0 0 1 0 0 1 1 is not correct and should have double errors since (p_4 even) holds and $(p_1, p_2, p_3)_{\text{computed}} + (p_1, p_2, p_3)_{\text{read}} = (0,1,0)$.

二、題組(20%)，共四個題組，各題組下的小題為單選，每一題組內所有小題完全答對得 5 分，答錯任一小題或未作答得 0 分。題組 A 包含題號 21~22，題組 B 包含題號 23~24，題組 C 包含題號 25~27，題組 D 包含題號 28~30

題組 A

Consider the following C code snippet:

```
01  #include <stdio.h>
02  #include <string.h>
03
04  void safeFunction() {
05      printf("This is a safe function call.\n");
06  }
07
08  void vulnerableFunction(char *str) {
09      char buffer[50];
10      strcpy(buffer, str);
11  }
12
13  int main(int argc, char *argv[]) {
14      if (argc > 1) {
15          vulnerableFunction(argv[1]);
16      }
17      return 0;
18  }
```

Assume that the program runs on x86-64 Linux.

21. For the command-line argument argv[1], which of the following situations would corrupt the stack?
    (a) If argv[1] contains a non-printable character.
    (b) If argv[1] contains a space.
    (c) If argv[1] is a comma-delimited string of the phone numbers of all NYCU CS faculty members.
    (d) None of the above.

22. If the stack is corrupted, which of the following is least likely to be affected?
    (a) The program control flow.
    (b) The local variables.
    (c) The function call arguments.
    (d) The program code.

題組 B

Assume each page is 512 bytes in size and the size of an integer is 2 bytes. Consider two C programs (Program 1 & Program 2) designed to initialize all elements of a 256-by-256 two-dimensional array (called arr as below) to 0. The array is stored in row-major order; that is, the array is stored arr[0][0], arr[0][1], …, arr[0][255], arr[1][0]. If the operating system only allocates 128 page frames on-demand to each program and adopts Least-Recently-Used (LRU) page replacement policy to manage those fames, please answer the following two questions

23. How many page faults would be generated while running Program 1:

(a) 1

(b) 128

(c) 256

(d) 65536

```
int x, y;
int [][] arr;

for (y=0; y<256; y++)
  for (x=0; x<256; x++)
    arr[x][y] = 0;
```

Program 1.

24. How many page faults would be generated while running Program 2:

(a) 1

(b) 128

(c) 256

(d) 65536

```
int x, y;
int [][] arr;

for (x=0; x<256; x++)
  for (y=0; y<256; y++)
    arr[x][y] = 0;
```

Program 2.

題組 C

P is a single-issue, in-order execution, five-stage pipeline implementation of the MIPS ISA without a branch prediction unit and without any data-forwarding path. The registers are read at the second half cycle of the Decode stage. The registers are written at the first half cycle of the Writeback stage. The branch decisions and targets are determined at the Execute stage. The fetch of the next instruction after a branch will be stalled until the branch decision is made. Answer the following questions when P is used to execute Program 3:

```
        addi $s1, $zero, 8
        addi $s2, $s1, $zero
Loop:   lw   $t0, 100($s1)
        addu $t0, $t0, $s2
        sw   $t0, 100($s1)
        addi $s1, $s1, -4
        bne  $s1, $zero, Loop
        addi $s2, $zero, $zero
```

Program 3.

25. How many total stall cycles will be inserted into the pipeline for proper execution throughout the entire program?

    (a) 8

    (b) 12

    (c) 16

    (d) 18

26. How many stall cycles are due to control hazards?

    (a) 2

    (b) 4

    (c) 6

    (d) 8

27. How many stall cycles can be removed by adding only the forwarding path from the EX/MEM register to one of the inputs of the ALU in the execute stage?

    (a) 6

    (b) 8

    (c) 10

    (d) 12

題組 D

Consider a 16-bit computer with a hierarchical memory system consisting of a disk, a 2^{16}-byte main memory using byte-addressing, and a two-level cache with primary cache of 2^9-byte data inside CPU. The primary cache is designed as follows. It is a direct mapped cache. Each block is 16-word long and the replacement policy is LRU. The OS adopts a 16-bit virtual address with a page size of 0.5KB.

28. Which configuration of the hierarchical memory system is true?

(a) The hierarchical memory system is suited to apply physically tagged and virtually indexed scheme since the virtual page offset field is equal to or longer than the index field for cache addressing.

(b) The virtual memory has 2^8 pages.

(c) The virtual page number field has a length of 6 bits.

(d) The tag field of physical address has a length of 8 bits.

29. Without regarding the disadvantage of virtually addressed cache, the virtually addressed cache is used in this computer. Assume the cache is empty initially. Given the following virtual address accesses to access cache: 0x8A78, 0x8A64, 0x8876, 0x8FF8, 0x8DE7. Which statement is true?

(a) The result of access to 0x8A64 is *miss*.

(b) The result of access to 0x8876 is *hit*.

(c) The result of access to 0x8FF8 is *miss* and the block containing 0x8A78 is replaced.

(d) The result of access to 0x8DE7 is *miss* and the block containing 0x8FF8 is replaced.

30. If the CPU base CPI = 1 and the CPU clock is running at 5GHz. The miss rate per lw/sw instruction in primary cache is 2%. There are 25% lw/sw instructions in a program. The access time to the second-level cache is 6ns and the global miss rate to main memory is 0.2% with the miss penalty of 240ns. Which statement is true?

(a) If the CPU has only primary cache without the second-level cache, the effective CPI is 6.

(b) If the CPU has only primary cache without the second-level cache, the effective CPI is 7.5.

(c) The CPU with 2-level cache has the effective CPI of 4.9.

(d) The CPU with 2-level cache has the effective CPI of 4.2.