國立彰化師範大學 99 學年度碩士班招生考試試題

系所： 資訊工程學系　　　　　　　　　　科目： 作業系統及計算機組織

一、　Please fill the blank in Figure 1 with the following items. **(7%)**

　　a. time slice expired　　**b.** interrupt occur　　**c.** I/O request　　**d.** I/O

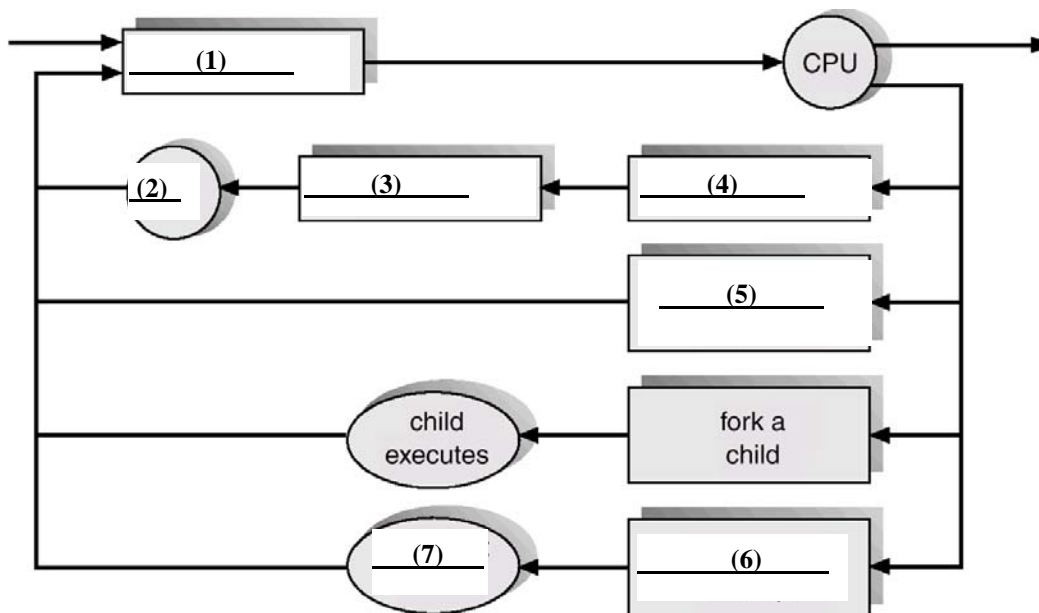　　**e.** I/O queue　　**f.** ready queue　　**g.** wait for an interrupt

Figure 1. Queueing-diagram representation of process scheduling

二、　Suppose that a disk drive has 5000 cylinders, numbered 0 to 4999. The drive is currently serving a request at cylinder 143, and the previous request was at cylinder 125. The queue of pending requests, in first in first out (FIFO) order, is

$$86, 1470, 913, 1774, 948, 1509, 1022, 1750, 130.$$

Starting from the current head position, what is the total distance (in cylinders) that the disk arm moves to satisfy all the pending requests for each of the following disk-scheduling algorithms?

a. first-come first-served **(4%)**　　　　**b.** shortest-seek-time-first **(4%)**

c. SCAN **(4%)**　　　　　　　　　　　　**d.** LOOK **(3%)**

三、　Which of the following programming techniques and structures are "good" for a demand-paged environment? Which are "bad"? Explain your answers. **(15%)**

　　a. Stack　　　　　　　　**b.** Hashed symbol table　　　　**c.** Sequential search

　　d. Binary search　　　　　**e.** Pure code

四、　Please complete the following figure with the following items. **(8%)**

a. sent request to device driver, block process if appropriate

b. can already satisfy request?

c. receive interrupt, store data in device-driver buffer if input, signal to unblock device

d. I/O completed, generate interrupt

e. determine which I/O completed, indicate state change to I/O subsystem

f. monitor device interrupt when I/O completed

g. process request, issue commands to controller, configure controller to block until interrupted

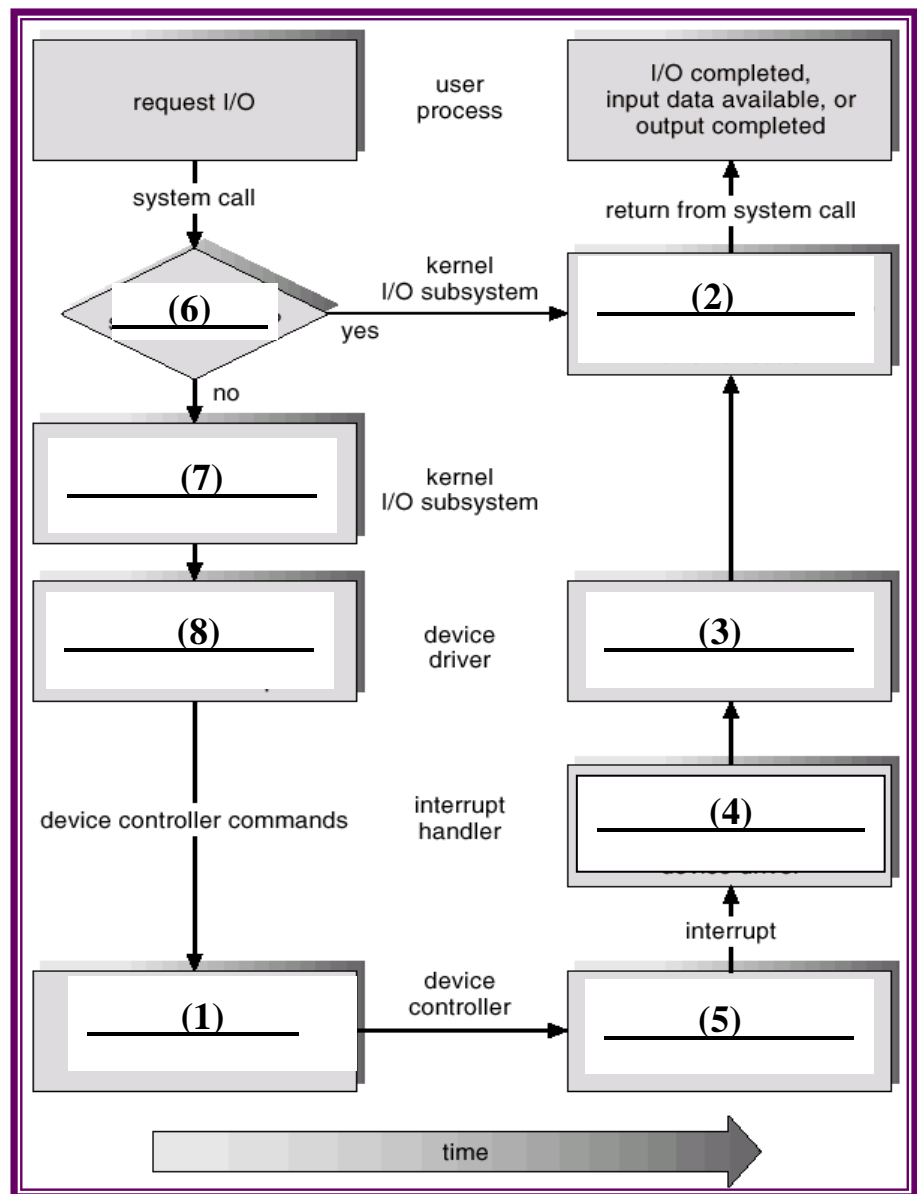**h.** transfer data (if appropriate) to process, return completion or error code

Figure 2. The life cycle of an I/O request.

五、 Consider a logical address space of eight pages of 1024 words each, mapped onto a physical memory of 32 frames.

  a. How many bits are there in the logical address? **(3%)**

  b. How many bits are there in the physical address? **(2%)**

六、 Determine if each of the following statements is true (T) or false (F). **(10%: (2*5)%)**

  a. In the IEEE 754 floating-point representation, the precision of represented numbers is determined by the size of fraction.

  b. In computer arithmetic, overflow occurs when adding two positive numbers and the sum is negative.

  c. In the memory hierarchy design, address mapping is a process by which a physical address is mapped to a virtual address used for accessing memory.

  d. When choosing a block to replace in a direct-mapped cache, all blocks in the cache can be candidates for replacement.

  e. If computer X runs a program in 10 seconds and computer Y runs the same program in 15 seconds, we can say that X is 1.5 times faster than Y.

七、 Answer the following questions. **(16%: (3*4+4)%)**

  a. When a program is executed on a computer, how to calculate the *CPU time* from its *instruction count*, *CPI*, and *clock cycle time*?

  b. In the MIPS addressing mode, what does *immediate addressing* mean?

  c. When page fault occurs and if all the pages in the main memory are in use, the operating system usually uses the LRU replacement scheme to choose a page to replace. What does this scheme mean?

  d. When handling a write request to a memory with cache, what is the *write-through* scheme?

  e. In the design of a MIPS processor, how to pass parameters and return result values during a procedure call? **(4%)**

八、Answer the following questions about data hazard. **(12%: (3*4)%)**

a. Data hazard is a type of pipeline hazard. One way to solve it is called *forwarding*, what does that mean?

b. However, the forwarding method cannot be applied in solving the *load-use* data hazard, why?

c. One solution to the load-use data hazard is called *pipeline stall*, what does that mean?

d. Reorder the code segment in Fig. 3 to avoid pipeline stalls when solving the load-use data hazard.

lw	$t1, 0($t0)
lw	$t2, 4($t0)
add	$t3, $t1, $t2
sw	$t3, 12($t0)
lw	$t4, 8($t0)
add	$t5, $t1, $t4
sw	$t5, 16($t0)

Fig. 3

九、Shown in Fig. 4 is a single-cycle implementation of the MIPS processor, please explain in detail how a branch-on-equal instruction, e.g., beq $t1, $t2, offset, is executed in the datapath. **(12%)**
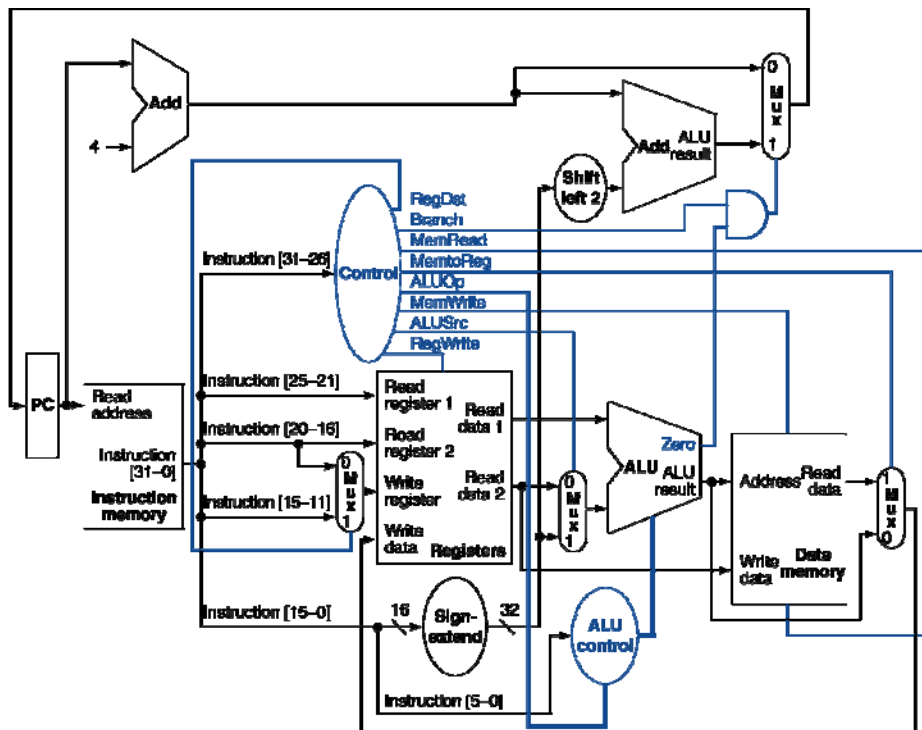

Fig. 4