复旦大学计算机科学技术学院

2017 ～ 2018 学年第一学期期末考试试卷

☑A 卷　　☐B 卷　　☐C 卷

课程名称： 操作系统　　**课程代码：** COMP130110.02-04

开课院系： 计算机科学与技术学院　　**考试形式：** 闭卷

姓　名：＿＿＿＿＿＿　　**学　号：**＿＿＿＿＿＿　　**专　业：**＿＿＿＿＿＿

声明：我已知悉学校对于考试纪律的严肃规定，将秉持诚实守信宗旨，严守考试纪律，不作弊，不剽窃；若有违反学校考试纪律的行为，自愿接受学校严肃处理。

学生（签名）：

年　　月　　日

题 号	1	2	3	4	5	6	7	8	总 分
得 分									

1. Choice (36%) fill in the blank with one most appropriate answer

(1) Assume a process P has two pure user-level threads, T1 and T2. Which one of the following statements is correct?
(A) T1 and T2 may be in running state at certain moment;
(B) If P is in Ready state, it is also possible that T1 (or T2) is still in Running state;
(C) If P is in Blocked state, it is impossible that T1 (or T2) is in Running state;
(D) If T1 (or T2) is in Blocked state, P must be in Blocked state.
Answer: B

(2) Some of operations below must be conducted only in kernel mode.

(i) Disable all interrupts. (ii) Make a system call.

(iii) Make a procedure call. (iv)Change the memory map.

Among the following 4 pairs, which one is correct?

(A) i, ii (B) i, iii (C) i, iv (D) iii, iv

Answer: C

(3) Which one of the following memory management schemes suffer from external fragmentation? D

A. Equal-Sized fixed partitioning
B. Unequal-sized fixed partitioning
C. Pure paging
D. Pure Segmentation

(4) What is the processing sequence when one process attempts to perform an I/O operation using user-level command? A

A.user-level command →system call → device driver → device controller →device

B. user-level command →system call → device controller → device driver →device

C. user-level command → device controller → system call → device driver →device

D. user-level command → device controller → device driver → system call →device

(5) A computer with pages in 8KB, a 256MB main memory, and 4GB virtual address space uses an inverted page table to implement its virtual memory. How many entries will the inverted page table contain? C

A. 16 B. 64

C. 32K D. 512K

(6) When designing a disk scheduling policy, which factor has the most significant impact on the disk I/O performance? A

A. Latency of moving the disk head
B. Time required to read one sector from the disk
C. The average rotational delay
D. The maximal rotational delay

(7) A computer has 6 tape drives, with n processes competing for them. Each process may need two drives. For which maximal values of n is the system deadlock free? C

A. 3
B. 4
C. 5
D. 6

(8) Which one of the following actions is effective when thrashing is happening in a computer system? A

A. Terminate some processes
B. Increase the size of disk swapping space
C. Increase the priorities of user processes
D. Upgrade the system with a high-speed CPU

(9) Which one of the following scheduling algorithms is non-preemptive and does NOT suffer from starvation? B

A. Static Priority-based
B. Highest Response Ratio Next
C. Round Robin
D. Shortest Process Next(SJF)

(10) Which one of the following events may cause the running process to be put back into the ready queue? A

A. An interrupt from I/O devices has occurred.
B. The process execute a wait() on a semaphore
C. The process tries to get a message from another process by invoking receive().
D. The process is accessing a page which is not resident in main memory.

(11) Suppose that a file occupies 10 disk blocks. Now we have to read the file, block by block, into a buffer area in main memory. If the buffer area has the same size as a disk block, reading a disk block into the buffer takes 100μs, transferring the data in buffer to a user area takes 50μs, analyzing a data block by CPU takes 50μs, then what is the total time for reading and analyzing the whole file (with single buffering)? _____C_____

A 1100μs
B 1500μs
C 1550μs
D 2000μs

(12) Which one of the following statements is CORRECT? D .

A. Symbolic linking is faster than hard linking
B. Symbolic linking requires less space than hard linking
C. Hard linking cannot link to a file on a remote machine
D. Removing a symbolic linking does not affect the file at all

2. Short Answer Questions（31 points）

(1) Please connect each concept or task on the right to the artifact on the left that use the concept or support it. (4 points)

(a) Process address space	(i) physical address
(b)Memory Management Unit	(ii) virtual address
(c) Main Memory	(iii) speed up address translation
(d)TLB	(iv) mapping the virtual addresses onto the physical addresses

Answer:

(a)-(ii); (b)-(iv); (c)-(i); (d)-(iii)

(2) Consider a memory system with a cache access time of 10ns and a memory access time of 200ns. If the effective access time is 10% greater than the cache access time, what is the hit ratio H? (Fractional answers are okay). (3 points)

Effect Access Time: Te = H x Tc + (1 - H)(Tm + Tc), where Tc = 10ns, Te = 1/1 x Tc, and Tm = 200ns.

Thus, (1.1) x (10) = H x 10 + (1 – H) (200 + 10) 11 = 10H + 210 – 210H

H = 199/200

(3) Explain how a mixture of I/O-bound processes with CPU-bound processes maximizes system utilization. Why is this more important in batch systems than it is on laptop computers we use? (4 points)

Want to keep all parts of the system busy, so if you can have some processes using the cpu while others wait on I/O, then the system will be more utilized. In general, our laptop systems are underutilized (there are a lot of wasted cycles both on I/O devices as well as cpu), and user interaction is more important. Batch systems were expensive systems that were busy all the time.

(4) In a demand-paging system with fixed allocation strategy, consider a page reference string 1, 2, 3, 4, 2, 5, 7, 2, 3, 2, 1, 7, 8 (6 points)

a) How many page faults would there be using FIFO replacement and 4 page frames?

answer: 10 faults

b) How many page faults with LRU and 4 page frames?

Answer: 9 faults

c) How many page faults using the Optimal algorithm and 4 page frames?

Answer: 7 faults

(5) Consider the following set of processes with arrival times and CPU execution times given in milliseconds.

Process	Arrival time	Execution time
P1	0	3
P2	2	10
P3	3	8
P4	4	4
P5	9	6

Consider the following three scheduling algorithms:

(a) Shortest Job First

(b) Shortest Remaining Time First

(c) Highest Response Ratio Next

Calculate the average waiting times for above scheduling algorithms. (6 points)

P1→P3→P4→P5→P2 [6.4ms]

P1→P3(1ms)→P4→P3(7ms)→P5→P2 [5.8ms]

P1→P2→P4→P3→P5 [8.0ms]

(6) Describe the concept of working set briefly, and explain why it can be used to

control the degree of multiprogramming? (4 points)

Answer:

At any instant of time t, the set of all the pages used by the k most recent memory references is call the working set w(k, t). It works as: the working set of each process is monitored; periodically remove from the resident set of a process those pages that are not in its working set. Because a process may execute only if its working set is in main memory, the control of the degree of multiprogramming is implied by the work set method.

(7) Given the following disk requests and arrival times, show the completion times for each request under the following disk scheduling policies: FCFS and SCAN. The head is initially at track 0 and it takes 0 time to read a track, but 1 time unit to move the head one track. (4 points)

Requested Track	Arrival Time	Completion Time (FCFS)	Completion Time (SCAN)
50	0		
0	10		
100	40		
90	100		

Answer:

FCFS: 50, 100, 200, 210

SCAN:50,200,100,110

3. Analysis (33%).

(1) Consider the following snapshot of a system (P=Process, R=Resource) : (11 points)

Current Available			
RA	RB	RC	RD
1	2	2	2

Maximum Demand				
	RA	RB	RC	RD
P0	3	2	1	4
P1	0	2	5	2
P2	5	1	0	5
P3	1	5	3	0
P4	3	0	3	3

Current Allocation				
	RA	RB	RC	RD
P0	1	0	1	1
P1	0	1	2	1
P2	4	0	0	3
P3	1	2	1	0
P4	1	0	3	0

Answer the following questions using banker's algorithm:

a) Calculate the *Needs* matrix: (2 points)

Needs				
	RA	RB	RC	RD
P0				
P1				
P2				
P3				
P4				

b) Is the system in a safe state? If so, show a safe order in which the processes can run. (3 points)

c) In the above given state can a request of one instance of RA by Process P0 be granted safely according to Banker's algorithm? If yes, show the safe sequence. A blank Need matrix is repeated here for convenience. (3 points)

d) In the above given state (as in b), can a request one instance of RD by P0 be granted safely according to Banker's algorithm? If yes, show the safe sequence. (3 points)

Answer

Needs				
	RA	RB	RC	RD
P0	2	2	0	3
P1	0	1	3	1
P2	1	1	0	2
P3	0	3	2	0
P4	2	0	0	3

Answer: Yes, P2→P0P4→P1→P3

可以是任何下面的一种

P2→P0→P4→P1→P3

P2→P0→P1→P4→P3

P2→P0→P1→P3→P4

P2→P4→P0→P1→P3

P2→P4→P1→P0→P3

P2→P4→P1→P3→P0

Answer:

如果请求获得授权，则更新后的状态为

Current Available			
RA	RB	RC	RD
0	2	2	2

Current Allocation				
	RA	RB	RC	RD
P0	2	0	1	1
P1	0	1	2	1
P2	4	0	0	3
P3	1	2	1	0
P4	1	0	3	0

Needs

	RA	RB	RC	RD
P0	1	2	0	3
P1	0	1	3	1
P2	1	1	0	2
P3	0	3	2	0
P4	2	0	0	3

更新后的状态不是安全状态，因此银行家算法不会对该请求进行授权。

Answer:

如果请求获得授权，则更新后的状态为

Current Available			
RA	RB	RC	RD
1	2	2	1

Current Allocation				
	RA	RB	RC	RD
P0	1	0	1	2
P1	0	1	2	1
P2	4	0	0	3
P3	1	2	1	0
P4	1	0	3	0

Needs				
	RA	RB	RC	RD
P0	2	2	0	2
P1	0	1	3	1
P2	1	1	0	2
P3	0	3	2	0
P4	2	0	0	3

更新后的状态不是安全状态，因此银行家算法不会对该请求进行授权。

(2) The students like accessing e-learning system. Two engineers maintain the e-learning system, one for software maintenance and the other for hardware repairing. The following policies are enforced:

- Students can access the e-learning system when neither of the engineers are maintaining the system;
- An engineer can not start maintenance when some students are using e-learning system;
- Hardware and software maintenance cannot be in action at the same time.
- Students are not allowed to access e-learning system during maintenance.

Try to simulate the scenario using pseudo-code in terms of semaphore. (12 points)

Ref. Answer:

This scenario can be modeled as the reader/writer problem that the readers have priority.

Pseudo-code is:

```
semaphore rSem = 1;/* 1 point */
semaphore wSem = 1;      /* 1 point */
int readCount = 0;

void students()
{
      while(TRUE)    /* 0.5 point */
      {
            wait(rSem);  /* Mutex, 2 points */
                  readCount++;
                  if (readCount == 1)
                        wait(wSem);       /* 1 point */
            signal(rSem);
            Surf();
            wait(rSem);  /* Mutex, 2 points */
                  readCount--;
                  if (readCount == 0)
                        signal(wSem);    /* 1 point */
            signal(rSem);
      }
}

void engineers()
{
      while(TRUE) /* 0.5point */
      {
            wait(wSem);      /* Mutex, 2 points */
                  Maintain();
            signal(wSem);
      }
}

void main()
{
      parbegin(students, engineers);      /* 1 point*/
}
```

(3) Consider a Unix-like file system where the file organization is based on the i-node structure. Assume there are 6 direct block pointers, a singly indirect pointer, a

doubly indirect pointer, and a triply indirect pointer in each i-node. Further, the system block size and the disk sector size are both 2KB, and each disk block pointer occupies 4 bytes. (10 points)

(a) Why not simply use only triple indirection to locate all file blocks? (2 points)

(b) What is the maximum file size supported by this system? (2 points)

(c) Suppose you want to create a new file of the maximum size. How many disk blocks would be used by the file system in order to create and store that file? NOTE: you can give the expression or equation, there's no need to write the final results. (3 points)

(d) Assuming that only the i-node is in main memory, how many disk accesses are required to access the byte at position 10,000? And how about the byte at position 100,000,000? (3 points)

Answer:

(a) Triple indirection is much slower, as it may result in multiple seeks to get to the desired block. Seeks take a long time. Triple indirection also consumes more index blocks for tiny files. Tiny files that only use a few KB will perform much better if they do not have to do seeks to find each level of indirection

(b) 6*2K+512*2K+512*512*2K+512*512*512*2K=12K+1M+512M+0.5G

(c) Besides the data blocks, we also need indrect index blocks.

data blocks: 6+2^9+2^18+2^27

index blocks: 1+(1+2^9)+(1+2^9+2^18)

total: 9+2^10+2^19+2^27

(d) 1 disk access is required to access the byte in position 10,000
100,000,000>6*2K
100,000,000>6*2K+1M
100,000,000<6*2K+1M+512M
thus, the byte in position 100,000,000 can be accessed through the doubly indirect pointer, which leads to 3 disk accesses.