USTC Operating Systems, Fall 2020

Midterm Exam

所在院系：33信息安全

Question	1	2	3	4	5	6	7	8	9		Total
Pts Assigned	20	5	5	20	15	10	11	14	5		100 + 5
Pts Obtained											

76

【考试要求】闭卷考试，每人可自带不超过 A4 大小的一张 cheating sheet。除此之外，在考试过程中不能参考任何文字资料，不得使用包括手机在内的任何电子设备。请用英文答题，必须写在试卷上指定的答题处，位置不够时写在试卷反面。

--- Good luck! ----

17

1. Write T or F next to each of the following statements, to indicate whether it is True (T) or False (F). (20 pts)

(1) An operating system is defined as hardware that converts software into a useful form for applications. **Answer:** F

(2) A context switch from one process to another can be accomplished without executing OS code in kernel mode. **Answer:** F

(3) An advantage of implementing threads in user space is that they don't incur the overhead of having the OS schedule their execution. **Answer:** T

(4) The OS (Process) provides the illusion to each thread that it has its own address space. **Answer:** F

(5) Threads that are part of the same process share the same stack. **Answer:** T

(6) Deadlock can never occur if no process is allowed to hold a resource while requesting another resource. **Answer:** T

(7) In Round Robin scheduling, it is advantageous to give each I/O bound process a longer quantum than each CPU-bound process (since this has the effect of giving the I/O bound process a higher priority). **Answer:** T

(8) The CPU's kernel mode provides operations that are not available in user mode. **Answer:** T

(9) A trap is an interrupt caused by an external event such as a mouse click. **Answer:** F (HW)

(10) A context switch is initiated by an interrupt, such as clock interrupt (HW) or a trap (SW). **Answer:** T

(11) In a batch system, every process runs to completion before the next process runs. (job) **Answer:** T

(12) A modern OS virtualizes a single CPU with time-sharing. **Answer:** T

(13) PCB stands for process control base. (block) **Answer:** F

(14) A user-level process cannot modify its own page table entries. (PTE, OS) **Answer:** T

(15) The address space of a process is part of its process state. **Answer:** T

(16) Entering a system call involves changing from user mode to kernel mode. **Answer:** T

(17) If all jobs have identical run lengths, a FIFO and a SJF (Shortest Job First) scheduler will behave the same. **Answer:** T

(18) If all jobs have identical run lengths, a Round Robin scheduler provides better average turnaround time than FIFO. **Answer:** F

(19) A SJF scheduler requires an oracle to predict how long each job will perform I/O in the future. **Answer:** T

(20) Shortest Remaining Time First is the best preemptive scheduling algorithm that can be really implemented in an Operating System. **Answer:** F

3

2. Categorize the following as one of the following: (I) interrupt, (E) exception, or (N) neither. (5 pts)

(1) Timer **Answer:** E

(2) Keyboard input **Answer:** I

(3) Divide by zero **Answer:** E

(4) Procedure call **Answer:** N

(5) System call **Answer:** I

5

3. For each of the following instructions, indicate whether the instruction should be Privileged (P) or Not (N). (5 pts)

(1) Change memory management registers **Answer:** P

(2) Write the program counter **Answer:** N

(3) Read the time-of-day clock — **Answer:** N

(4) Set the time-of-day clock — **Answer:** P

(5) Change processor priority — **Answer:** P

15

4. Answer yes or no, or with a single term or short answer, as appropriate. (20 pts)

(1) Does the test-and-set instruction need to be a privileged instruction?

Answer: Yes, because it works as an atom sentence, which cannot be interrupted

(2) What approach to dealing with deadlock does the Banker's algorithm implement?

Answer: It prevents processes from holding on the resources that may cause deadlock. In other words, it prevents "holding and waiting", the 2nd condition.

(3) Which of the following scheduling algorithms can lead to starvation? FIFO, Shortest Job First, Priority, and Round Robin

Answer: SJF may cause the longest process to starve.
If the priority level cannot be changed, the priority also can cause starve.

(4) A program containing a race condition will always/sometimes/never result in data corruption or some other incorrect behavior?

Answer: Sometimes. It depends on the scheduling of OS, the processing of data and the accessing of memory

(5) A system that meets the four deadlock conditions will always/sometimes/never result in deadlock?

Answer: Sometimes. It depends on the allocation of resources.

(6) On a system with *n* CPUs, what is the maximum number of processes that can be in the ready, run, and blocked states?

Answer: The maximum number of ready processes and blocked processes are unlimited. It depends on the memory's size. The maximum number of running processes must be n

(7) On a system with *n* CPUs, what is the minimum number of processes that can be in the ready, run, and blocked states?

Answer: The minimum number of ready process and blocked processes are 0.
The minimum number of running processes can be 1, when only OS is running

(8) What characteristic is common to traps, interrupts, and system calls, but different in subroutine calls?

Answer: The first three are blocked because some other items, such as software, I/O hardware, OS need to use the CPU.
While the last one doesn't, it's only because that process has already been running for such a long time that the time exceeds the maximum allowing time. So OS blocks it and make other process continue running to avoid starve

3/8

7

5. Answer the questions. (15 pts)

For the processes listed in the following table, draw a chart illustrating their execution and calculate the average turnaround time (rounding to the nearest hundredth) using

Process	Arrival Time	Processing Time
A	0.000	3
B	1.001	6
C	4.001	4
D	6.001	2

(1) First-Come First-Served

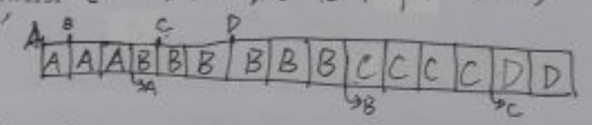
(2) Shortest Job First

(3) Shortest Remaining Time First

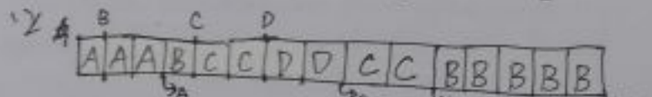
(4) Round Robin (quantum = 2)

(5) Round Robin (quantum = 1)

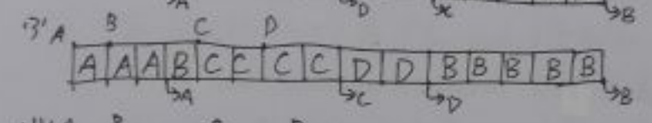
Answers: (the unit time is 1 per block)

1) 

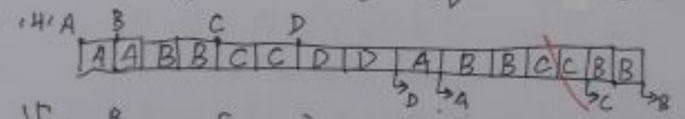
$\frac{3+8+9+9}{4} = 7.25$

2) 

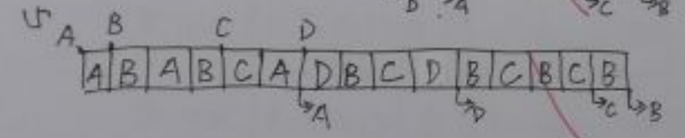
$\frac{3+14+6+2}{4} = 6.25$

3) 

$\frac{3+14+4+4}{4} = 6.25$

4) 

$\frac{9+14+9+2}{4} = 8.5$

5) 

$\frac{6+14+10+2}{4} = 9.75$

4/8

6. Answer the questions. (10 pts)

Suppose two threads execute the following C code concurrently, accessing shared variables a, b, and c:

Initialization

int a = 4; int b = 0; int c = 0;

Thread 1	**Thread 2**
① if (a < 0) {	④ b = 10;
② c = b - a;	⑤ a = -3;
} else {	
③ c = b + a;	
}	

What are the possible values for c after both threads complete? You can assume that reads and writes of the variables are atomic, and that the order of execution of statements within each thread is preserved by the C compiler and the hardware so it matches the code above.

Answer: Insert the two threads together:

④⑤①②③	c = 13	④①⑤③②	c = 7
①④⑤②③	c = 7	④①③⑤②	c = 7
①④⑤③②	c = 7	④①③②⑤	c = 14
①④③②⑤	c = 4	①④⑤③②	c = 7
		①④③②⑤	c = 14
		①④③⑤②	c = 14

Above all, c = 4, 7, 13, 14 are possible values.

7. Answer the questions. (11 pts)

Recall the various deadlock detection and prevention algorithms we've discussed in this course, and consider the following snapshot of a system with five processes (P1, P2, P3, P4, P5) and four resources (R1, R2, R3, R4). There are no current outstanding queued unsatisfied requests.

Currently Available Resources

R1	R2	R3	R4
2	1	2	0

	Current Allocation				Max Need			
Process	R1	R2	R3	R4	R1	R2	R3	R4
P1	0	0	1	2	0	0	3	2
P2	2	0	0	0	2	7	5	0
P3	0	0	3	4	6	6	5	6
P4	2	3	5	4	4	3	5	6
P5	0	3	3	2	0	6	5	2

(1) Is this system currently deadlocked, or can any process become deadlocked? Why or why not? If not deadlocked, give an execution order. (5 pts)

Answer: ①. give (0,0,2,0) to P1, and P1 exits.
now available resources: (2,1,3,2)
② give (2,0,0,2) to P4, and P4 exits.
now available resources: (4,4,8,6)
③ give (0,3,2,0) to P5, and P5 exits.
now available resources: (4,7,11,8)
④ give (0,7,5,0) to P2, and P2 exits.
now available resources: (6,7,11,8)
⑤ give (6,6,2,2) to P3, and P3 exits
now available resources: (8,7,14,12)

(2) If a request from a process P1 arrives for (0, 4, 2, 0), can the request be immediately granted? Why or why not? If yes, show an execution order. (3 pts)

Answer: NO, because P1 doesn't need R2, and R2 is not enough
Moreover, if it means P1 change its needs to (0,4,2,0)
it cannot be granted as well, because now the deadlock happens.

(3) If a request from a process P2 arrives for (0, 1, 2, 0), can the request be immediately granted? Why or why not? If yes, show an execution order. (3 pts)

Answer: NO. because the available resources now are changed into (2,0,0,0), and no process can continue. In other words, deadlock happens.

8. Answer the questions. (14 pts)

A new theme park, Jurassic Park, consists of a dinosaur museum and a safari park. Park visitors wander around the museum for a while, then take a ride through the safari park in an automated train. However, there are only N single-passenger cars in the train, so if a visitor wants to take a safari ride, and no car is free, then he or she must wait in line. Complete the pseudo-code below to model entry and exit to the train (i.e., a "trainEntry" function and a"trainExit" function) using semaphores, i.e., semWait() and semSignal().

Answer:

semaphore mutex = 1;

semaphore emptyCar = N; /* number of empty cars */

semaphore fullCar = 0; /* number of full cars */


```
trainEntry {
    while (1) {
        semSignal (emptyCar);
        semWait (fullCar);
        semSignal (mutex);
        put passenger on train;
        semWait (mutex);
        semSignal (fullCar);
        semWait (emptyCar);
    }
}

trainExit {
    while (1) {
        semSignal(fullCar);
        semWait (emptyCar);
        semSignal (mutex);
        have passenger exit train;
        semWait(mutex);
        semSignal (emptyCar);
        semWait (fullCar);
    }
}
```

9. You will get full credit for the problem below no matter what your answer is, so just tell me the truth. (Extra Credit: 5 pts)

Was the exam too easy/just right/too hard?

Answer: A little hard.

Indeed, it is the first time that I have an all-English exam. And I find I didn't get the key points of PPTs, which means some of what I reviewed doesn't exist on the paper and some of what I ignored shows on the paper.