CITY UNIVERSITY OF HONG KONG

Course code & title	:	CS3103 Operating Systems
Session	:	Semester B 2022
Time allowed	:	50 minutes

---

1. This paper consists of 2 sections.
2. Answer ALL questions in both sections.
3. Section A 50 marks.
4. Section B 50 marks.
5. Specify the Section and Question number clearly for EACH answer in the answer script.
6. Submit ONE pdf file of the answer script to Canvas.
7. Use your Student ID to name the pdf file.

---

*This is an **open-book** examination.*

Section A [50 marks]
Please write the question number with your answer in your submission.

1. [2 mark]
Which of the following is the most efficient for a multiple-word I/O transfer.

A) Interrupt-driven I/O
B) Programmed I/O
C) Direct memory access
D) All the above are equally efficient

2. [2 mark]
The collection of program, data, stack, and attributes is referred to as the _____ .

3. [2 mark]
A semaphore that does not specify the order in which processes are removed from the queue is a _____ semaphore.

4. [2 mark]
A _____ occurs when multiple processes or threads read and write data items so that the final result depends on the order of execution of instructions in the multiple processes.

5. [3 mark]
Please state the difference between blocking receive and non-blocking receive.

6. [2 mark]
A situation in which two or more processes are unable to proceed because each is waiting for one of the others to do something is a _____ .

7. [2 mark]
A _____ is a semaphore that takes on only the values of 0 and 1.

8. [2 mark]
_____ is a code segment within a process that requires access to shared resources and that must not be executed while another process is in a corresponding code segment.

9. [2 mark]
To achieve mutual exclusion, the difference between solutions via mutex and semaphore is that _____

10. [2 mark]
After the I/O device issues an interrupt signal to the processor, the following events will happen. Write down the correct sequence of those events.

(a) Processor loads new PC value based on interrupt
(b) Processor pushes PSW and PC onto control stack
(c) Process the interrupt
(d) Processor finishes execution of current instruction
(e) Processor restores PSW and PC from the control stack

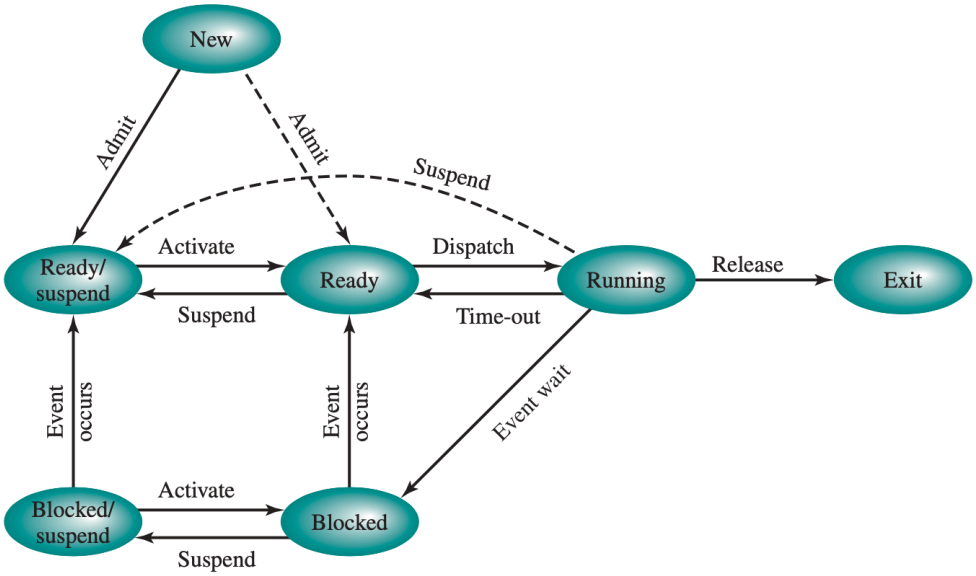
11. [2 mark]
Which of the following statements _____ are correct description of kernel-level threads (KLTs)? Please select all choices that are correct.

(a) KLTs are created by invoking an application-level function.
(b) The kernel is aware of the existence of KLTs
(c) The transfer of control from one thread to another within the same process requires a mode switch to the kernel is one of the advantages of the KLT approach.
(d) Using KLTs cannot provide a better performance than a single-threaded solution on a single-processor system.

12. [3 mark]
Please explain if the following states are process only, thread only, or both.

(a) Running
(b) Suspend
(c) Blocked

Consider the 7-state process model and then answer the question 13 and 14.



13. [5 marks]

Complete the following table by inserting the state of a process.

State	Status of the process	Location of the process
	Available for execution	Main memory
	Awaiting an event	Main memory
	Awaiting an event	Secondary memory

14. [5 marks]

Complete the table by filling the state transition. The first one is an example.

Event	Transition
The dispatcher chooses a process to run.	Ready -> Running

A process requests a resource which is not currently available.	
OS swaps out a process periodically.	
A resource for which a process waiting in the main memory becomes available.	

15. [4 mark]
Please state the drawback of disabling interrupts for achieving mutual exclusion.

16. [3 mark]
_____, referred as the tendency of execution involving a number of memory locations that are clustered, can be exploited via _____ cache blocks.

(a) Temporal locality, recently used
(b) Spatial locality, frequently used
(c) Temporal locality, frequently used
(d) Spatial locality, large

17. [2 mark]
Which of the following situations may cause deadlock?

(a) A process releases the resources
(b) A process enters an infinite loop
(c) Multiple processes compete for resources and circular wait happens
(d) Multiple processes compete for a sharable device

18. [2 mark]
If a process is awakened, then:

(a) this process can compete for CPU again
(b) the priority of this process becomes the highest
(c) the PCB is moved to the head of the ready queue
(d) this process turns into the running state

19. [3 mark]
Which of the following does not belong to a PCB?

(a) Process ID
(b) Process state
(c) Stack pointer
(d) Global variable

Section B
Q1 [15 marks]
Please read the following program and answer the questions.

```
int x = 0;

void *function (void *arg) {
     x-=2;
     cout << "thread: " << x << endl;
     pthread_exit(NULL);
}

int main () {

     // variables are declared here
     pid = fork();
     x++;

     if (pid == 0) {
           pthread_create(&tid, NULL, function, NULL);
           pthread_join(tid, NULL);
           x+=1;
           cout << "process0: " << x << endl;
      }
      else {
           wait(NULL);
           x+=10;
           cout << "process1: " << x << endl;
     }

      pthread_exit(NULL);

}
```

(1) List the outputs of the program. Please show the output in order. [7 marks]
(2) Please explain why the program will have the above values. [8 marks]

Q2 [15 marks]

Refer to the following solution to the bounded-buffer producer/consumer problem using semaphore. The buffer size is 18.

```
/* program boundedbuffer */
const int sizeofbuffer = /* buffer size */;
semaphore s = 1, n= 0, e= sizeofbuffer;
void producer()
{
     while (true) {
          produce();
          semWait(e);
          semWait(s);
          append();
          semSignal(s);
          semSignal(n);
     }
}
void consumer()
{
     while (true) {
          semWait(n);
          semWait(s);
          take();
          semSignal(s);
          semSignal(e);
          consume();
     }
}
void main()
{
     parbegin (producer, consumer);
}
```

(1) What is the value of each of the three semaphores when a consumer wants to take an item from the empty buffer? Assuming there is no producer. [7 marks]
(2) Following (1), a producer is producing data. Show the changes of the three semaphores. [8 marks]

Q3 [20 marks]

Consider the following solution using semaphores to solve the one-write many-readers problem.

int readcount;
Semaphore sem_x, sem_y;

```
// Program for the writer

semWait(sem_x);

/* writing */

semSignal(sem_x);
```

```
// Program for the reader

semWait(sem_y);

readcount++;

if (readcount==1) semWait(sem_x);

semSignal(sem_y);

/*reading*/

semWait(sem_y);

readcount--;

if (readcount==0) semSignal(sem_x);

semSignal(sem_y);
```

A) Suppose only a reader is now reading. No other readers or writer. [14 marks]
(1) [5 marks]
Show the values of the following three variables: readcount, sem_x, sem_y

(2) [5 marks]
What will happen if a writer wants to write, while the first reader in (1) is still reading? Explain with detailed semaphore values.

(3) [4 marks]

After the writer described in (2), what will happen if two more readers want to read? (The system already has a reader and writer)

B) [6 marks]
(1) [3 marks]
Could we replace a semaphore with a mutex? Explain the possibility.

(2) [3 marks]
What is the potential issue of this solution?