King Saud University

**College of Computer and Information Sciences** 

**Computer Science Department** 

CSC 227: Operating Systems

Total Marks: 20	Time: 7:45pm – 9:15pm (90 minutes)
Fall 2016-17	Name:
Midterm Exam II	ID#:
Date: 8-December-2016	Section#: or Teacher Name:

**Instructions:** 

- This exam has 8 pages.
- Do not use pencil or erasable pens.
- Write clearly and neatly.

Exercise No.	Full Mark	Student's Mark
Exercise1	5	
Exercise2	5	
Exercise3	5	
Exercise4	5	
Total	20	

Question 1. (5 marks) Select ONLY ONE ANSWER (the best answer).

### <u>Copy your answer for question 1-1 to 1-10 in the table on page3. ONLY THAT TABLE</u> <u>WILL BE GRADED. (your answer should be one of the letters A, B, C or D)</u>

1.	Degree of multiprogramming is controlled by
A Short-term scheduler	
В	Long-term scheduler
С	Process Control Block (PCB)
D	Fork()

3	When a parent process terminates, a possible
5.	solution
۸	The parent process life-span is extended until all
A	children voluntarily terminates
В	Terminates all parents up to the root process
C	This case is impossible to occur - a child
C	process always terminates first.
D	All children terminated - Cascading termination

5	If one thread opens a file with read privileges
5.	then
٨	Other threads in the same process can also read
A	from that file
D	Any other thread connet read from that file
D	Any other thread cannot read from that the
С	Process state
D	All of the mentioned above

7	The time required to create a new thread in an
7.	existing process is:
٨	Greater than the time required to create a new
А	process
B	Equal to the time required to create a new
D	process
C	Less than the time required to create a new
C	process
р	Greater or equal than the time required to create
ע	a new process

2.	I/O bound process
А	Spends more time doing I/O than computations
В	Spends more time doing computations.
С	Is managed by short-term scheduler
D	Is not a process but an I/O device

ł.	Which one of the following is not shared by threads?
A	Program counter
В	Stack
С	Open files

D	Both A and B
6.	Thread synchronization is required because
۸	All threads of a process share the same address
A	space
R	All threads of a process share the same global
D	variables
С	All threads of a process can share the same files
D	All of the mentioned

8.	Which is not the requirement of a solution to the critical-section problem?
А	Mutual Exclusion
В	Progress
C	Busy Waiting
D	Bounded Waiting

9.	Which one of these operations is not	10.	Which one is a hardware solution to the		
	necessarily a critical section?		critical-section problem?		
A	Changing a common variable	А	Test and Set Lock		
В	Opening a file on the disk	В	Shared memory		
C	Writing in a database	С	Semaphore		
D	Modifying a shared memory.	D	Monitor		

# Your Answer should be written in the following table:

1.	2.	3.	4.	5.	6.	7.	8.	9.	10.
Α	А	D	D	А	С	С	С	В	Α

#### Question 2. (5 Marks)

**2.1** (a) Explain the purpose of long-term scheduler and short-term scheduler? (1 Mark).

#### Answer:

**Long-term scheduler:** (or job scheduler) – selects which processes should be brought into the ready queue

**Short-term scheduler:** (or CPU scheduler) – selects which process should be executed next and allocates CPU

**2.1** (b) Fill in the blanks the applicable process state: (1 Mark).

Possible Process States: New, Waiting, Ready, Terminated And Running





**2.1** Explain context switching? Is it an overhead? (1.5 marks)

### Answer:

When CPU switches to another process, the system must save the state of the old process and load the saved state for the new process via a context switch Context of a process represented in the PCB Context-switch time is overhead; the system does no useful work while switching

- 2.2 Explain why processes cooperate (list at least two reasons) ? List the two interprocess communication models? (1.5 marks)

# **Reasons for cooperating processes:**

- Information sharing
- Computation speedup
- Modularity
- Convenience

Two models of IPC

- 1. Shared memory
- 2. Message passing

#### Question 3 (5 Mark).

3.1 Define each of the following terms: (2 Marks).

fork(); a system call to create a new process, which by default is a complete copy of its parent,

exec(); usually works as normal – replace the running process including all threads

**3.2** Explain and give the number of "السلام عليكم" printed by the following program: (3 Marks).

#include <stdio.h></stdio.h>
#include <unistd.h></unistd.h>
main()
{
int i;
for (i=0; i<3;i++) fork();
printf( "السلام عليكم");
}



#### Question 4 (5 Mark).

**4.1** A good solution of the critical section problem requires three conditions. One of them is "Bounded Waiting". What is the meaning of "Bounded waiting" in a critical-section solution? [1 Mark].

A bound must exist on the number of times that other processes are allowed to enter their critical sections after a process has made a request to enter its critical section and before that request is granted. This should assume that each process executes at a nonzero speed and no assumption concerning relative speed of the n processes.

**4.2** Considering the producer-consumer problem and considering the code below running on a multi-processor system, what is the problem with this code? Explain [1 mark]

The following are shared between the two processes:		
Buffer[], counter and BUFFER_SIZE.		
while (true) {/*produce 1 item in	while (true) {	
next_produced */	while (counter $== 0$ )	
while (counter == BUFFER_SIZE)	; /* do nothing */	
; /* do nothing */	next_consumed = buffer[out];	
buffer[in] = next_produced;	out = (out + 1) % BUFFER_SIZE; counter	
$in = (in + 1)$ % BUFFER_SIZE;	;	
counter++;	/* consume the item in next_consumed */	
}	}	

The problem is that counter++ and counter-- can be implemented as 3 different operations and thus their execution may be interleaved and cause a wrong result.

- 4.3 Answer the following questions: [2.5 marks]
  - a. The kernel code can be non-preemptive. What does it mean exactly? [0.5 mark]

Non-preemptive means that it cannot be interrupted.

b. If the kernel code is non-preemptive, list 3 conditions in which it can stop running? [1 mark]

It will run until it exits kernel mode (1), or until it blocks (2), or until it voluntarily yields CPU (3).

c. The kernel code can be non-preemptive. What does it mean exactly? [0.5 mark]

Non-preemptive means cannot be interrupted.

**d.** What is priority inversion? [0.5 mark]

Scheduling problem when lower-priority process holds a lock needed by higher priority process.

e. What technique is used to solve the priority inversion problem? [0.5 mark]
Priority-inheritance protocol.

# End of the exam.