AL HUSSEIN BIN TALAL UNIVERSITY
Faculty of Information Technology
Department of Computer Science

Course Name: Operating Systems

Lecturer: Prof. Bassam Alqaralleh

Student Name: ....

Student Number: ☐☐☐☐☐☐☐☐☐☐☐☐

Date: 25-Jan-2023

Duration: 90 Minutes

Final Exam

First Sem. 2022/2023

Question	Max mark	Student's score
Q1	20	
Q2	10	
Q3	5	
Q4	10	
Q5	5	
Total	50	

1/1

Note: your answer for each question should fit in the designated area.

Q1: Multiple Choices: Mark the most correct answer (read carefully): [20 Marks, 0.5 mark each]

اختار(ي) الاجابة الأكثر دقة (اقرأ(ي) بعناية وبدون تسرع

1. The scheduler which controls the *degree of multiprogramming* (the number of processes in memory) is:

   A. Short term scheduler
   B. Medium term scheduler
   C. Long term scheduler
   D. Multi-level queue scheduler

2. The key idea behind a medium-term scheduler is that it can:

   A. Increase the degree of multiprogramming
   B. Reduce the degree of multiprogramming
   C. Add processes to active contention for the CPU
   D. None of the above

3. Which of the following properties is not one of the properties of *direct communications*?

   A. A link is established between a pair of processes only if they have a shared mailbox
   B. A link is associated with exactly two processes
   C. Between each pair of processes, there exists exactly one link
   D. None of the above

4. Which of the following is <u>not</u> one of the four major categories of the benefits of multithreaded programming?

   A. Resource sharing
   B. Economy
   C. Utilization of single-processor architectures
   D. None of the above

5. RMIs allow:

   A.   Objects to be passed to remote procedures
   B.   Ordinary data structures to be passed to remote procedures
   C.   Objects to be passed to local procedures
   D.   Ordinary data structures to be passed to local procedures

Scanned with CamScanner

6. Regarding RMI implementation, the ----------------- is responsible for unmarshalling the parameters and invoking the desired method on the server.

   A.   stub
   B.   skeleton
   C.   port
   D.   socket

7. Which of the following is not shared between threads that are belonging to the same process?

   A.   Its code section
   B.   Its data section
   C.   Open files
   D.   A register set

8. In a single-processor architecture, **multithreading** generally occurs by:

   A.   Priority multiplexing
   B.   Time-division multiplexing
   C.   Memory-space multiplexing
   D.   None of the above

9. A kernel-level threading model can be best described as:
   A.   One-to-many
   B.   Many-to-one
   C.   One-to-one
   D.   Many-to-many

10. The model of multi-threading in which the kernel can schedule another thread in the application for execution if a thread performs a blocking system call is:

    A.   Kernel-level multithreading model
    B.   User-level multithreading model
    C.   Many-to-one multithreading model
    D.   None of the above

11. The threads which are generally fast to create and manage are:

    A.   Kernel-level threads
    B.   User-level threads
    C.   Hybrid threads
    D.   None of the above

12. Which of the following is not true regarding one-to-one multithreading model?

A. The overhead of creating kernel threads can burden the performance of an application
B. Each user-level thread maps to kernel thread
C. Multiple threads are unable to run in parallel on multiprocessors
D. None of the above

13. Which of the following is true regarding many-to-many (N:M) multithreading model?

A. Developers can create as many user threads as necessary
B. The corresponding kernel threads can run in parallel on a multiprocessor.
C. The kernel can schedule another thread for execution, when a thread performs a blocking system call
D. All the mentioned above

14. The dispatcher is the module that gives control of the CPU to the next process to be executed. This function involves the following tasks except:

A. Switching context
B. Switching to system mode
C. Jumping to the proper location in the user program to restart that program
D. None of the above

15. Context-switch time is ………………….

A. Overhead
B. Useful
C. desired
D. None of the mentioned above

16. CPU scheduling decisions may take place under the following circumstances except:

A. When a process switches from the running state to the waiting state
B. When a process switches from the ready state to the running state
C. When a process switches from the waiting state to the ready state
D. When a process terminates

17. The collection of processes in the memory that are ready and waiting to be executed:

A. Ready queue
B. Job pool
C. Loading queue
D. None of the above

18. Whenever the CPU becomes idle, the OS must select one of the processes in the ready queue to be executed. The selection process is carried out by

A. long-term scheduler.
B. Short-term scheduler
C. Medium-term scheduler
D. None of the mentioned above

19. The time it takes for the dispatcher to stop one process and start another running is known as the:

A. switching latency
B. Switching delay
C. Dispatch latency
D. Context delay

20. The ready queue is generally stored as a linked list. A ready-queue header contains only one pointer to the:

A. First PCB in the list.
B. Last PCB in the list
C. Both of the above
D. None of the above

21. A CPU-bound program would typically have .... CPU bursts.

A. many short
B. Few long
C. Many long
D. None of the above

22. The CPU scheduling algorithm, which partitions the ready queue into several separate queues where the processes are permanently assigned to one queue, generally based on some property of the processes, is called:

A. Multilevel feedback queue scheduling
B. Multilevel queue scheduling
C. Multiple-processor scheduling
D. Priority scheduling

23. The formula which can be used to predict the value of the length of the next CPU burst is called:

    A. Predicted average formula
    B. Exponential average formula
    C. Accurate average formula
    D. Historical-based average formula

24. In a multilevel queue-scheduling algorithm, there must be a scheduling among the queues, which is commonly implemented as:

    A.   Absolute priority preemptive scheduling
    B.   Absolute priority non-preemptive scheduling
    C.   Fixed-priority preemptive scheduling
    D.   Fixed-priority non-preemptive scheduling

25. The effect, which occurs when some short processes wait for one big process to get off the CPU, and results in lower CPU and device utilization than might be possible if the shorter processes were allowed to go first, is called:

    A.   Delay effect
    B.   Waiting effect
    C.   Convoy effect
    D.   Latency effect

26. In the context of the makeup of dispatch latency, the dispatch latency has two components: the dispatch phase and the

    A.   Conflict phase
    B.   Interrupt phase
    C.   Response phase
    D.   Switch phase

27. The sequence which can be created by monitoring the real system, recording the sequence of actual events, and is then used to drive the simulation, is called:

    A.   Event list
    B.   Event record
    C.   Trace tape
    D.   Trace record

28. CPU scheduling decisions may take place under the following circumstances except:

A. When a process switches from the running state to the waiting state
B. When a process switches from the ready state to the running state
C. When a process switches from the waiting state to the ready state
D. When a process terminates

29. In order to evaluate the performance of different CPU scheduling algorithms, the following methods can be used except:

A. Evaluation modeling
B. Queuing models
C. Simulation
D. Implementation

30. The semaphore whose integer values can only be 0 or 1 is called:

A. Counting semaphore
B. Integer semaphore
C. Limited semaphore
D. Binary semaphore

31. If preemption is required to deal with deadlocks, then the following issues need to be addressed except:

A. Selecting a victim
B. Rollback
C. A resource-ordering
D. Starvation

32. While a process is in its critical section, any other process that tries to enter its critical section must loop continuously in the entry code. This situation can be best described as:

A. Dispatch waiting
B. Spinlock waiting
C. Busy waiting
D. Loop waiting

33. Which of the following is not one of the three main methods which can be used for dealing with deadlocks:

A. Use some protocol to ensure that the system will never enter a deadlock state
B. Allow the system to enter deadlock state, detect it, and then ignore it.
C. Ignore the problem all together, and pretend that deadlock never occur in the system
D. None of the above

34. A deadlock situation may occur if and only if four necessary conditions hold simultaneously in the system:

A. Mutual execution, hold and wait, no pre-emption, no circular wait

B. Mutual execution, hold and wait, pre-emption, circular wait

C. Mutual execution, hold and wait, no pre-emption, circular wait

D. Mutual execution, hold and wait, pre-emption, no circular wait

35. If a deadlock is detected, the system must recover by:

A. Terminating some of the deadlocked processes

A. Preempting resources from some of the deadlocked processes

B. Either A or B

C. None of the above

36. The method for dealing with deadlock which is based on having a priori information on how each process will be utilizing the resources is called:

A. Deadlock prevention

B. Deadlock avoidance

C. Deadlock detection

D. Deadlock prediction

37. A transaction (program unit) where all operations associated to it are executed to completion; or none are performed, can be described as:

A. Atomic

B. real time

C. critical

D. None of the above

38. A solution to the critical section problems must satisfy the following criteria except:

A. Mutual exclusion

B. Stable

C. Bounded waiting

D. Progress

39. A situation where processes wait indefinitely within the semaphore is called:

    A. Aging
    B. Starvation
    C. Definite blocking
    D. Deadlock

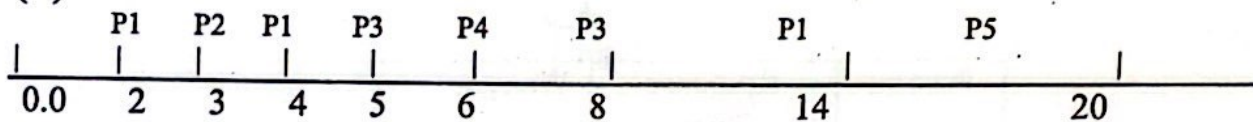40. The major difficulty of the "implementation" evaluation approach of scheduling algorithms is:

    A. Limited accuracy
    B. Cost
    C. Unrealistic
    D. inflexible

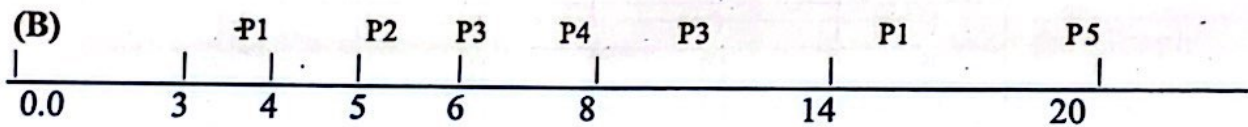**Q2)** Consider the following set of processes, with the length of the CPU-burst time given in milliseconds:

Process	Arrival time	Burst time
P1	0	9
P2	2	1
P3	4	3
P4	5	1
P5	6	6

2.1. Which Gantt chart does illustrate the execution of these processes using a **Preemptive** SJF scheduling algorithm? [5 Marks]

(A)

P1	P2	P1	P3	P4	P3		P1		P5	

0.0 2 3 4 5 6 8 14 20

(B)

	P1	P2	P3	P4	P3		P1		P5	

0.0 3 4 5 6 8 14 20

(C)

P1	P3	P2	P4	P3		P1		P5	

0.0 3 4 5 6 8 14 20

(D)

P1		P3	P4	P3	P2		P5	

0.0 9 10 11 13 14 20
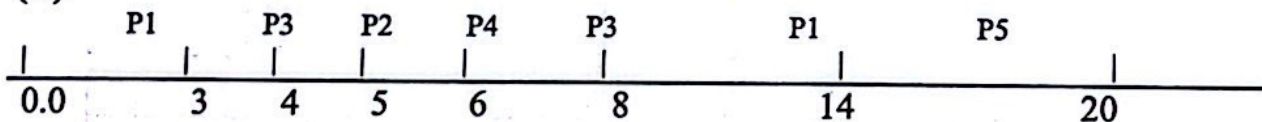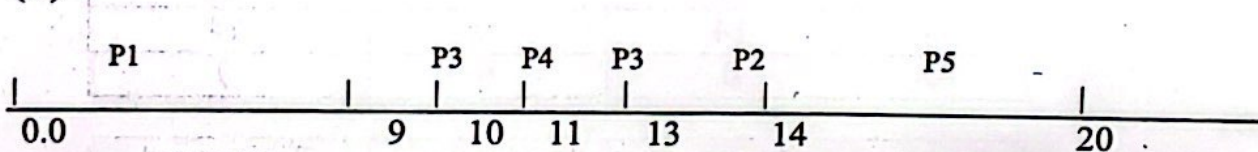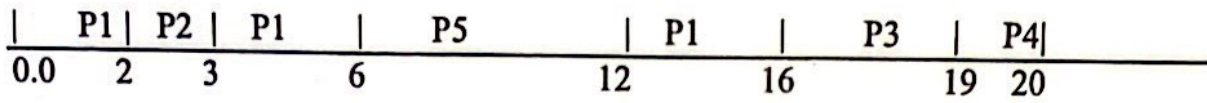
2.2. Consider the following Gantt chart, which table may contain the right response time, waiting time and turnaround time for each process? [5 Marks]

	P1	P2	P1		P5		P1		P3		P4
0.0	2	3		6		12		16		19	20

(A)

Process	Response Time	Waiting Time	Turnaround Time
P1	7	7	16
P2	0	0	1
P3	12	12	15
P4	14	14	15
P5	0	0	6

(B)

Process	Response Time	Waiting Time	Turnaround Time
P1	0	7	16
P2	0	0	1
P3	12	12	15
P4	14	14	15
P5	0	0	6

(C)

Process	Response Time	Waiting Time	Turnaround Time
P1	0	16	7
P2	0	1	0
P3	12	15	12
P4	14	15	14
P5	0	6	0

(D)

Process	Response Time	Waiting Time	Turnaround Time
P1	7	16	0
P2	0	1	0
P3	12	15	12
P4	14	15	14
P5	0	6	0

Q3) [5 marks] Given the **resource-allocation graph below:**

A. Draw the corresponding wait-for graph? [3 marks]

| Resource-allocation graph | Wait-for-Graph |
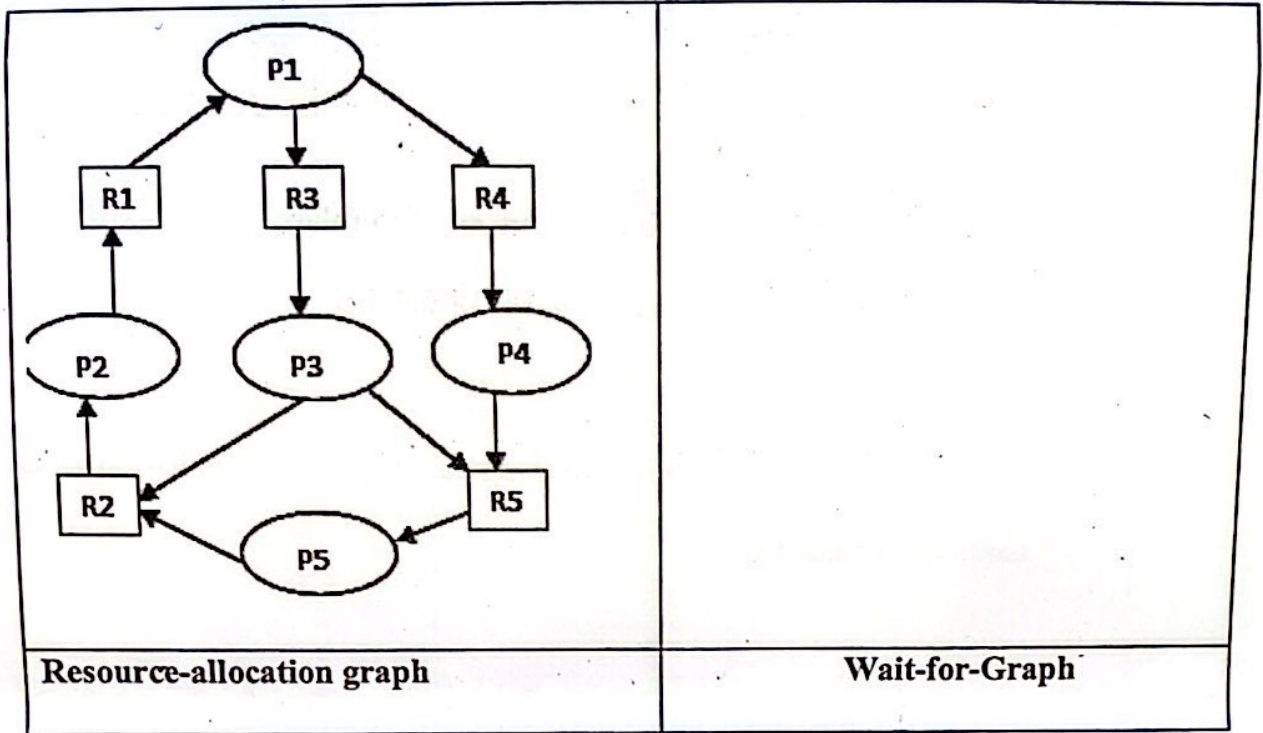
B. Show if there are any cycles exist? [2 marks]

1- .
2- .
3- .
4- .

Q4) [10 marks] Consider the following snapshot of a system:

	Allocation				Max				Available				Need			
	A	B	C	D	A	B	C	D	A	B	C	D	A	B	C	D
P0	0	0	1	2	0	0	1	2	1	5	2	0				
P1	1	0	0	0	1	7	5	0								
P2	1	3	5	4	2	3	5	6								
P3	0	6	3	2	0	6	5	3								
P4	0	0	1	4	0	6	5	6								

Answer the following questions using the banker's algorithm:

A. What is the content of the matrix need? **(2 marks)**

B. Is the system in a safe state? If your answer is "yes", give at least one safe sequence? **(4 marks)**

C. If a request from process P1 arrives for (0,4,2,0), can the request be granted immediately? **(4 mark)**

Q5) [5 Marks] Write down the solution for first readers writers problem – Reader process using the semaphores **wrt** and **mutex**.

Hint: The code of the writer process is written below; you need to write the code of the reader process only.

Writer Process	Reader Process

wait(wrt);
 	
   writing is performed
 	
signal(wrt);	

Best wishes