Feedback on Quiz # 3 (Summative)

The **Quiz # 3** was composed of **10** questions, which were randomly selected from a Question Bank of 30 questions. Answers and feedback comments for all of the questions are given below:

Q1.1	Which of the following statement(s) is/are true?
(a)	Threads within a process share the same data section.
(b)	Race conditions can occur between threads within a process.
(c)	Every thread within a process must be mapped to a separate kernel thread.
(d)	The behaviour of a multi-threaded process is always deterministic.
(e)	One disadvantage of multithreading is that it always leads to memory consistency errors.
Feedback: Threads within a process have their own registers and stack but they share the code and data sections with other threads within the process. Race conditions occur when multiple threads within a process read and write the same data item ‘simultaneously’. Race conditions can also occur between processes. Threads within a process are not always mapped to a separate kernel thread. Depending on the multithreading model, it could be many-to-one, one-to-one, or many-to-many. The behaviour of a multi-threaded process is not always deterministic. In fact, the behaviour of most multi-threaded processes is non-deterministic because their threads are usually concurrent. Multithreading does not always lead to memory consistency errors. They only occur when mutual exclusion is not enforced within critical sections.	

Q1.2	Which of the following statement(s) is/are true?
(a)	The behaviour of a multi-threaded process can be non-deterministic.
(b)	Threads within a process can be mapped to the same kernel thread.
(c)	Each thread within a process maintains its own copy of shared data.
(d)	A process can have 0 threads.
(e)	Threads within a process are guaranteed not to interfere with each other.
Feedback: The behaviour of a multi-threaded process is not always deterministic. In fact, the behaviour of most multi-threaded processes is non-deterministic because their threads are usually concurrent.	

Depending on the multithreading model, threads within a process can be mapped to the same kernel thread (many-to-one), many kernel threads (many-to-many), or different kernel threads separately (one-to-one). Threads within a process have their own registers and stack but they share the code and data sections of the process. They do not maintain their own copy of shared data. A process must have at least one thread. Threads within a process may interfere with each other, for example, when they reference the same shared data item.

Q1.3	Which of the following statement(s) is/are true?
(a)	Memory consistency errors occur when multiple threads “race” to modify the same shared data item.
(b)	Threads within a process can interact with each other through shared data.
(c)	All threads are managed directly by the operating system.
(d)	Race condition occurs when multiple processes compete to finish their tasks in the shortest amount of time.
(e)	Preventing all but one thread from writing to the same shared data item concurrently is adequate to avoid memory consistency errors.
Feedback: Memory consistency errors occur when different threads have inconsistent views of the same shared data item. This is usually due to different threads accessing the same shared data item concurrently, while at least one thread modifies the shared data item. Therefore, preventing multiple processes from writing to the same shared data item concurrently is not adequate to avoid memory consistency errors. We need to ensure that there is no other thread accessing the same shared data item when a thread is modifying it. Threads within a process interact with each other through shared data. For example, one thread updating a shared data item which is then accessed by another thread. Only kernel threads are managed directly by the operating system. User threads are managed by the operating system indirectly through kernel threads. Race condition occurs when multiple processes or threads read and write the same shared data item at the same time.	

Q2.1	Which of the following protocols are used in the Application Layer?
(a)	HTTP
(b)	SMTP
(c)	TCP
(d)	UDP
(e)	IP

Feedback:
HTTP (Hypertext Transfer Protocol) is an application layer protocol that is used by the World Wide Web. SMTP (Simple Mail Transfer Protocol) is an application layer protocol that is used for electronic mail.

TCP (Transmission Control Protocol) is a transport layer protocol that provides reliable, in-order delivery. UDP (User Datagram Protocol) is a transport layer protocol that provides a "best effort" service.

IP (Internet Protocol) is a network layer protocol used for transporting datagrams across network boundaries.

Q2.2	Which of the following protocols are used in the Transport Layer?
(a)	TCP
(b)	UDP
(c)	HTTP
(d)	SMTP
(e)	IP

Feedback:
HTTP (Hypertext Transfer Protocol) is an application layer protocol that is used by the World Wide Web. SMTP (Simple Mail Transfer Protocol) is an application layer protocol that is used for electronic mail.

TCP (Transmission Control Protocol) is a transport layer protocol that provides reliable, in-order delivery. UDP (User Datagram Protocol) is a transport layer protocol that provides a “best effort” service.

IP (Internet Protocol) is a network layer protocol used for transporting datagrams across network boundaries.

Q2.3	Which of the following protocols are used in the Network Layer?
(a)	IP
(b)	HTTP
(c)	TCP
(d)	UDP
(e)	SMTP

Feedback:
HTTP (Hypertext Transfer Protocol) is an application layer protocol that is used on the World Wide Web. SMTP (Simple Mail Transfer Protocol) is an application layer protocol that is used for electronic mail.

TCP (Transmission Control Protocol) is a transport layer protocol that provides reliable, in-order delivery. UDP (User Datagram Protocol) is a transport layer protocol that provides a “best effort” service.

IP (Internet Protocol) is a network layer protocol used for transporting datagrams across network boundaries.

Q3.1	The term ‘Latency’ refers to …
(a)	Delay that happens in data communication over a network
(b)	Transmission capacity of a network
(c)	Rate at which something is able to move
(d)	Rate at which bits are transferred end-to-end between sender and receiver
(e)	Strength of the connection between two hosts in the network
Feedback: Latency = Delay that happens in data communication over a network Bandwidth = Transmission capacity of a network Speed = Rate at which something is able to move Throughput = Rate that bits are transferred end-to-end between sender and receiver ‘Strength of the connection between two hosts in the network’ is meaningless	

Q3.2	The term ‘Bandwidth’ refers to …
(a)	Transmission capacity of a network
(b)	Delay that happens in data communication over a network
(c)	Rate at which something is able to move
(d)	Rate at which bits are transferred end-to-end between sender and receiver
(e)	Strength of the connection between two hosts in the network
Feedback: Latency = Delay that happens in data communication over a network Bandwidth = Transmission capacity of a network Speed = Rate at which something is able to move Throughput = Rate that bits are transferred end-to-end between sender and receiver ‘Strength of the connection between two hosts in the network’ is meaningless	

Q3.3	The term 'Throughput' refers to …
(a)	Rate at which bits are transferred end-to-end between sender and receiver
(b)	Transmission capacity of a network
(c)	Length of time a packet needs to queue in the buffer
(d)	Delay that happens in data communication over a network
(e)	Strength of the connection between two hosts in the network
Feedback: Latency = Delay that happens in data communication over a network Bandwidth = Transmission capacity of a network Throughput = Rate that bits are transferred end-to-end between sender and receiver 'Strength of the connection between two hosts in the network' is meaningless 'Length of time a packet needs to queue in the buffer' does not relate to throughput	

Q4.1	Which of the following statements about ‘Hosts’ are true?
(a)	Hosts can be clients on the network
(b)	Hosts can be servers on the network
(c)	Hosts are the same thing as End Systems
(d)	Hosts can be routers in the network core
(e)	Hosts must be connected to the network via a wired connection
Feedback: The term ‘host’ refers to any end system (i.e. a system that a network user will directly interact with) on the network. These are all clients and servers on the network, and may include routers in the network edge (but not the network core). They can be connected via a wire or wirelessly.	

Q4.2	Which of the following statements about the ‘Network Edge’ are true?
(a)	Network Edge includes all clients in the network
(b)	Network Edge includes all servers in the network
(c)	Network Edge includes all routers in the network
(d)	Network Edge is where packet switching takes place
(e)	Network Edge means the physical wires that send bits on the network
Feedback: The term ‘host’ refers to any end system (i.e. a system that a network user will directly interact with) on the network. These are all clients and servers on the network, and may include routers in the network edge (but not the network core). They can be connected via a wire or wirelessly. The network edge is the abstract part of the network that consists of all end systems on the network. Therefore, it includes all clients and servers, but not routers.	

Q4.3	Which of the following statements about the ‘Network Core’ are true?
(a)	Network Core is a mesh of interconnected routers
(b)	Network Core is where packet routing takes place
(c)	Network Core is where packet forwarding takes place
(d)	Network Core includes all hosts in the network
(e)	Network Core means the physical wires that send bits on the network
Feedback: The network core is the abstract part of the network that consists of the mesh of interconnected routers that are responsible for packet switching. Packet switching means both routing and forwarding packets on the network.	

Q5.1	You and your friend John are working on a Parallel Programming project, which is an application that does not use IO functions. You have been told by your project manager that your software should have at-least 6 times speedup as compared to its sequential implementation. You have also been told that your application will run on a system with 16 CPUs. Estimate the minimum percentage of the code that needs to be parallelized in your application, in order to meet the above performance requirement?
(a)	**88.89%**
(b)	93.92%
(c)	98.07%
(d)	92.90%
(e)	86.75%

Feedback:
Let's denote Speedup by S.
Amdhal's law states that: S = 1 / ((1-f) + f/N)

We have been given S = 6, and N = 16 and we are required to estimate f (the percentage of parallel code). We can rewrite Amdhal's law and separate f as shown below:

f = (N (S – 1)) / (S (N – 1))

f = (16 (6 – 1)) / (6 (16 – 1)) => 0.8889 => 88.89%

Sanity check: If we plug 88.89% into the equation we get 6x speedup.

Q5.2	You and your friend Alice are working on a Parallel Programming project, which is an application that does not use IO functions. You have been told by your project manager that your software should have at-least 10 times speedup as compared to its sequential implementation. You have also been told that your application will run on a system with 24 CPUs. Estimate the minimum percentage of the code that needs to be parallelized in your application, in order to meet the above performance requirement?
(a)	**93.92%**
(b)	88.89%
(c)	98.07%
(d)	92.90%
(e)	86.75%

Feedback:
Let's denote Speedup by S.
Amdhal's law states that: S = 1 / ((1-f) + f/N)

We have been given S = 10, and N = 24 and we are required to estimate f (the percentage of parallel code). We can rewrite Amdhal's law and separate f as shown below:

f = (N (S – 1)) / (S (N – 1))

f = (24 (10 – 1)) / (10 (24 – 1)) => 0.9392 => 93.92% (we roundup to meet the 10x requirement)

Sanity check: If we plug 93.92% into the equation we get 10x speedup.

Q5.3	You and your friend Ahmad are working on a Parallel Programming project, which is an application that does not use IO functions. You have been told by your project manager that your software should have at-least 20 times speedup as compared to its sequential implementation. You have also been told that your application will run on a system with 32 CPUs. Estimate the minimum percentage of the code that needs to be parallelized in your application, in order to meet the above performance requirement?
(a)	**98.07%**
(b)	88.89%
(c)	93.92%
(d)	92.90%
(e)	86.75%

Feedback:
Let's denote Speedup by S.
Amdhal's law states that: S = 1 / ((1-f) + f/N)

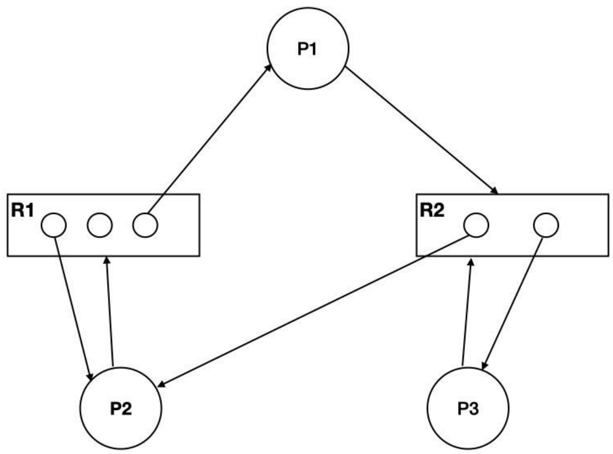
We have been given S = 20, and N = 32 and we are required to estimate f (the percentage of parallel code). We can rewrite Amdhal's law and separate f as shown below:

f = (N (S – 1)) / (S (N – 1))

f = (32 (20 – 1)) / (20 (32 – 1)) => 0.9807 => 98.07% (we roundup to meet the 10x requirement)

Sanity check: If we plug 98.07% into the equation we get 20x speedup.

Q6.1 Consider the following resource allocation graph (RAG) of three processes and two types of resources:

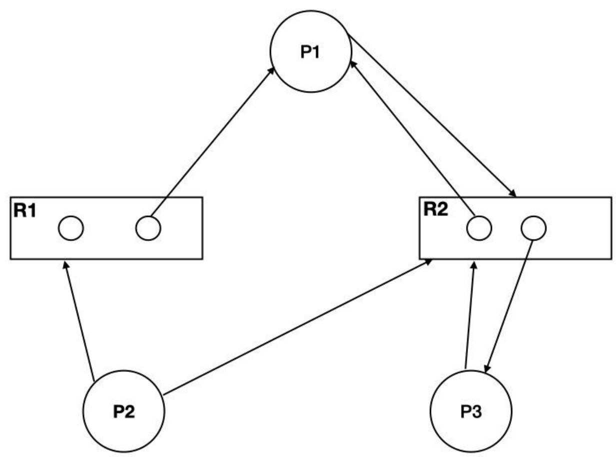


From the following statements, select all that are valid for the above set of processes and resources.

(a)	Process P2 can complete its execution, followed by either P1 or P3.
(b)	If we remove the currently available instance of R1, processes P1, P2, and P3 will be deadlocked.
(c)	Only process P3 is currently blocked.
(d)	If we add one more instance to R2, the deadlock will be resolved.
(e)	Processes P1, P2, and P3 are currently deadlocked.

Feedback:
In the given RAG:
P1 is holding one instance of R1. It is also requesting one instance of R2, which is currently unavailable, hence P1 cannot complete.
P2 is holding one instance of R1 and one instance of R2. It is also requesting one instance of R1, which is currently available. Therefore, P2 can complete its execution.
P3 is holding one instance of R2. It is also requesting one more instance of R2, which is currently unavailable, hence P3 cannot complete.

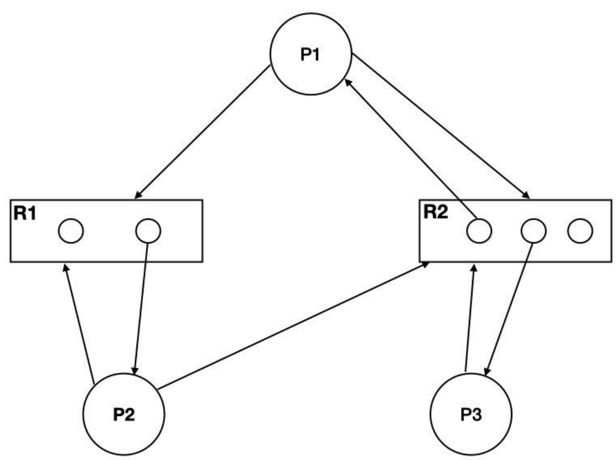
From the above, we can see that P1 and P3 are blocked. P2 can complete its execution, followed by either P1 or P3.
As the overall system is not deadlocked, adding more instances to R2 is irrelevant.
If we remove one instance of R1, then P2 will be blocked. The overall system is then deadlocked.

Q6.2	Consider the following resource allocation graph (RAG) of three processes and two types of resources:  From the following statements, select all that are valid for the above set of processes and resources.
(a)	Processes P1, P2, and P3 are currently deadlocked.
(b)	If we add one more instance to R2, the deadlock will be resolved.
(c)	If we add one more instance to R1, the deadlock will be resolved.
(d)	Process P2 can complete its execution, followed by either P1 or P3.
(e)	Only processes P1 and P3 are currently blocked.

Feedback:
In the given RAG:
P1 is holding one instance of R1 and one instance of R2. It is also requesting one more instance of R2, which is currently unavailable, hence P1 cannot complete.
P2 is holding no resource. It is requesting one instance of R1 and one instance of R2, but there is no instance of R2 available. Therefore, P2 cannot complete its execution.

P3 is holding one instance of R2. It is also requesting one more instance of R2, which is currently unavailable, hence P3 cannot complete.
From the above, we can see that all processes P1, P2 and P3 are deadlocked. The overall system is currently deadlocked.
If we add one more instance to R2, then process P1 or P2 or P3 may be allocated that instance, therefore, it will be able complete its execution, consequently breaking the circular wait and resolving the deadlock.
Adding one more instance to R1 does not help to break the circular wait and resolve the deadlock because the deadlock is caused by lacking instances of R2.

Q6.3 Consider the following resource allocation graph (RAG) of three processes and two types of resources:



From the following statements, select all that are valid for the above set of processes and resources.

(a)	Process P2 can complete its execution, followed by either P1 or P3.
(b)	Process P1 can complete its execution, followed by either P2 or P3.
(c)	Processes P1, P2, and P3 are currently deadlocked.
(d)	If we add one more instance to R1, the deadlock will be resolved.
(e)	Only process P3 is currently blocked.

Feedback:

In the given RAG:

P1 is holding one instance of R2. It is also requesting one instance of R1 and one one instance of R2, which are both currently available. Therefore, P1 can be completed.

P2 is holding no resource. It is requesting one instance of R1 and one instance of R2, which are both currently available. Therefore, P1 can be completed.

P3 is holding one instance of R2. It is also requesting one more instance of R2, which is currently available. Therefore, P3 can complete.

From the above, we can see that all processes P1, P2, and P3 can complete their execution, and the system can pick any of them to run first. Therefore, the overall system is not deadlocked.

As the overall system is not deadlocked, adding more instances to R1 is irrelevant.

Q7.1 A resource allocation system that uses Banker's algorithm for 3 resource types (A, B, C) and 4 processes (P0, P1, P2, P3) is currently in the following state.
(Max. Claim Matrix: max need of each process. Allocation Matrix: resources held by each process. Request (Need) Matrix: resources needed to complete. Available Resource Vector: free resources.)

Max. Claim Matrix

	A	B	C
P0	4	3	2
P1	3	3	1
P2	4	0	2
P3	7	2	1

Allocation Matrix

	A	B	C
P0	1	2	1
P1	2	1	1
P2	3	0	0
P3	2	1	0

Request (Need) Matrix

	A	B	C
P0	3	1	1
P1	1	2	0
P2	1	0	2
P3	5	1	1

Available Resource Vector

A	B	C
2	1	2

Presuming that the system is currently in Safe State, which one(s) of the following safe sequence(s) is/are possible?

(a)	P2,P0,P1,P3
(b)	P2,P0,P3,P1
(c)	P2,P1,P0,P3
(d)	P2,P1,P3,P0
(e)	The presumption is incorrect, the system is not in a Safe State.

Feedback:
Only P2 can run to completion. So, we will have the following available vector:

A	B	C
6	1	4

Either P0 or P3 can run to completion but since the options continue with P0 we will take it. So, we will have the following available vector:

A	B	C
10	4	6

Either P1 or P3 can run to completion, let's take P1. So, we will have the following available vector:

A	B	C
13	7	7

Let's complete P3 now:

A	B	C
20	9	8

So, the correct options are:
P2, P0, P1, P3
P2, P0, P3, P1

Q7.2 A resource allocation system that uses Banker's algorithm for 3 resource types (A, B, C) and 4 processes (P0, P1, P2, P3) is currently in the following state.
(Max. Claim Matrix: max need of each process. Allocation Matrix: resources held by each process. Request (Need) Matrix: resources needed to complete. Available Resource Vector: free resources.)

Max. Claim Matrix

	A	B	C
P0	2	3	3
P1	1	1	0
P2	2	5	1
P3	3	4	3

Allocation Matrix

	A	B	C
P0	1	2	3
P1	1	0	0
P2	1	3	0
P3	0	3	1

Request (Need) Matrix

	A	B	C
P0	1	1	0
P1	0	1	0
P2	1	2	1
P3	3	1	2

Available Resource Vector

A	B	C
0	2	1

	Presuming that the system is currently in Safe State, which one(s) of the following safe sequence(s) is/are possible?
(a)	P1,P2,P0,P3
(b)	P1,P2,P3,P0
(c)	P1,P3,P0,P2
(d)	P1,P3,P2,P0
(e)	The presumption is incorrect, the system is not in a Safe State.

Feedback:

Only P1 can run to completion. So, we will have the following available vector:

A	B	C
1	3	1

Either P0 or P2 can run to completion but since the options continue with P2 we will take it. So, we will have the following available vector:

A	B	C
3	8	2

Either P0 or P3 can run to completion, let's take P0. So, we will have the following available vector:

A	B	C
5	11	5

Let's complete P3 now:

A	B	C
8	15	8

So, the correct options are:

P1,P2,P0,P3

P1,P2,P3,P0

Q7.3 A resource allocation system that uses Banker's algorithm for 3 resource types (A, B, C) and 4 processes (P0, P1, P2, P3) is currently in the following state.
(Max. Claim Matrix: max need of each process. Allocation Matrix: resources held by each process. Request (Need) Matrix: resources needed to complete. Available Resource Vector: free resources.)

Max. Claim Matrix

	A	B	C
P0	3	2	2
P1	4	1	2
P2	0	3	2
P3	1	2	1

Allocation Matrix

	A	B	C
P0	3	1	1
P1	1	1	1
P2	0	1	0
P3	0	0	1

Request (Need) Matrix

	A	B	C
P0	0	1	1
P1	3	0	1
P2	0	2	2
P3	1	2	0

Available Resource Vector

A	B	C
2	2	0

Presuming that the system is currently in Safe State, which one(s) of the following safe sequence(s) is/are possible?

(a)	P3,P0,P1,P2
(b)	P3,P0,P2,P1
(c)	P3,P2,P1,P0
(d)	P3,P2,P0,P1
(e)	The presumption is incorrect, the system is not in a Safe State.

Feedback:
Only P3 can run to completion. So, we will have the following available vector:

A	B	C
3	4	1

Either P0 or P1 can run to completion but since the options continued with P0 we will take it. So, we will have the following available vector:

A	B	C

6	6	3

Either P1 or P2 can run to completion, let's take P1. So, we will have the following available vector:

A	**B**	**C**
10	7	5

Let's complete P2 now:

A	**B**	**C**
10	10	7

So, the correct options are:

P3,P0,P1,P2

P3,P0,P2,P1

Q8.1	Let’s assume we have two hosts A & B connected through 2 routers with zero propagation delay. We would like to transmit a 20 Mbit file from A to B, where the link bandwidths are given as R_1 = 10Mbps, R_2 = 4Mbps and R_3 = 5Mbps. Assume that the file is completely stored by a router before it starts sending it to the next link. What would be the end-to-end delay between the hosts A & B?
(a)	**11 seconds**
(b)	13 seconds
(c)	17 seconds
(d)	5 seconds
(e)	2 seconds
Feedback: The transmission delay from first host to router #1: 20 Mbits / 10Mbps = 2 seconds The transmission delay from router #1 to router #2: 20 Mbits / 4Mbps = 5 seconds The transmission delay from router #2 to second host: 20 Mbits / 5Mbps = 4 second Total End-to-End delay will be = 11 seconds	

Q8.2	Let’s assume we have two hosts A & B connected through 2 routers with zero propagation delay. We would like to transmit a 24 Mbit file from A to B, where the link bandwidths are given as R_1 = 8Mbps, R_2 = 4Mbps and R_3 = 6Mbps. Assume that the file is completely stored by a router before it starts sending it to the next link. What would be the end-to-end delay between the hosts A & B?
(a)	**13 seconds**
(b)	11 seconds
(c)	17 seconds
(d)	6 seconds
(e)	3 seconds
Feedback: The transmission delay from first host to router #1: 24 Mbits / 8Mbps = 3 seconds The transmission delay from router #1 to router #2: 24 Mbits / 4Mbps = 6 seconds	

The transmission delay from router #2 to second host: 24 Mbits / 6Mbps = 4 second Total End-to-End delay will be = 13 seconds

Q8.3	Let's assume we have two hosts A & B connected through 2 routers with zero propagation delay. We would like to transmit a 45 Mbit file from A to B, where the link bandwidths are given as R_1 = 5Mbps, R_2 = 15Mbps and R_3 = 9Mbps. Assume that the file is completely stored by a router before it starts sending it to the next link. What would be the end-to-end delay between the hosts A & B?
(a)	**17 Seconds**
(b)	11 seconds
(c)	13 seconds
(d)	9 seconds
(e)	3 seconds
Feedback: The transmission delay from first host to router #1: 45 Mbits / 5Mbps = 9 seconds The transmission delay from router #1 to router #2: 45 Mbits / 15Mbps = 3 seconds The transmission delay from router #2 to second host: 45 Mbits / 9Mbps = 5 second Total End-to-End delay will be = 17 seconds	

Q9.1	A packet propagates over an optical fibre link of distance 3600 km with a transmission rate of 36 Mbps. The length of the packet is 1500 bytes and the propagation speed is 2.5×10^8 m/s. Select all the following statements that are true:
(a)	**The transmission delay d_{trans} = 333.33 µs**
(b)	**The propagation delay d_{prop} = 14.40 ms**
(c)	The transmission delay d_{trans} = 41.67 µs (forgetting to multiply by 8)
(d)	The propagation delay d_{prop} = 14.40 sec
(e)	The propagation delay depends on packet length.

Feedback:
Propagation delay d_{prop} depends on the length of the physical link (d) and on the propagation speed in the medium (s).

d = 3600 km

s = 2.5×10^8 m/s

d_{prop} = d/s = 3600 km/ (2.5×10^8 m/s) = 14.40 ms.

Transmission delay d_{trans} depends on the packet length (L) and the link bandwidth/transmission rate (R).

L = 1500 bytes

R = 36 Mbps

d_{trans} = L/R = 1500 bytes/ 36 Mbps = 1500 x 8 bits / 36 x 10^6 bps = 333.33 µs.

Q9.2	A packet propagates over an optical fibre link of distance 2700 km with a transmission rate of 22 Mbps. The length of the packet is 1400 bytes and the propagation speed is 2.5 x 10^8 m/s. Select all the following statements that are true:
(a)	**The transmission delay d_{trans} = 509.09 µs**
(b)	**The propagation delay d_{prop} = 10.80 ms**
(c)	The transmission delay d_{trans} = 63.63 µs (forgetting to multiply by 8)
(d)	The propagation delay d_{prop} = 10.80 sec
(e)	**The propagation delay depends on the length of the link.**

Feedback:

Propagation delay d_{prop} depends on the length of the physical link (d) and on the propagation speed in medium (s).

d = 2700 km

s = 2.5 x 10^8 m/s

d_{prop} = d/s = 2700 km/ (2.5 x 10^8 m/s) = 10.80ms.

Transmission delay d_{trans} depends on the packet length (L) and the link bandwidth/transmission rate (R).

L = 1400 bytes

R = 22 Mbps

d_{trans} = L/R = 1400 bytes/ 22 Mbps = 1400 x 8 bits / 22 x 10^6 bps = 509.09 µs.

Q9.3	A packet propagates over an optical fibre link of distance 4300 km with a transmission rate of 41 Mbps. The length of the packet is 1350 bytes and the propagation speed is 2.5 x 10^8 m/s. Select all the following statements that are true:
(a)	**The transmission delay d_{trans} = 263.41 µs**
(b)	**The propagation delay d_{prop} = 17.20 ms**
(c)	The transmission delay d_{trans} = 32.93 µs (forgetting to multiply by 8)
(d)	The propagation delay d_{prop} = 17.20 sec
(e)	The propagation delay depends on the packet length.

Feedback:

Propagation delay d_{prop} depends on the length of the physical link (d) and on the propagation speed in medium (s).

d = 4300 km

s = 2.5 x 10^8 m/s

d_{prop} = d/s = 4300 km/ (2.5 x 10^8 m/s) = 17.20ms.

Transmission delay d_{trans} depends on the packet length (L) and the link bandwidth/transmission rate (R).

L = 1350 bytes

R = 41 Mbps

d_{trans} = L/R = 1350 bytes/ 41 Mbps = 1350 x 8 bits / 41 x 10^6 bps = 263.41 µs.

Q10.1	Which of the following does the TCP protocol provide?
(a)	Congestion control
(b)	Flow control
(c)	In-order delivery
(d)	Low delay
(e)	Reasonable throughput

Feedback:
TCP is a connection-oriented protocol that gives reliable, in-order delivery. It also provides congestion and flow control.

UDP is a connectionless protocol that provides a "best effort" service. It has no connection establishment (which can add delay) and no congestion/flow control.

However, neither protocol provides any delay, bandwidth or throughput guarantees.

Q10.2	Which of the following does the UDP protocol provide?
(a)	Connectionless communication
(b)	Best-effort communication
(c)	Flow control
(d)	In-order delivery
(e)	Delay guarantees

Feedback:
TCP is a connection-oriented protocol that gives reliable, in-order delivery. It also provides congestion and flow control.

UDP is a connectionless protocol that provides a "best effort" service. It has no connection establishment (which can add delay) and no congestion/flow control.

However, neither protocol provides any delay, bandwidth or throughput guarantees.

Q10.3	Which of the following cannot be provided by TCP?
(a)	Delay guarantees
(b)	Bandwidth guarantees
(c)	Throughput guarantees
(d)	Ordered delivery guarantees
(e)	Congestion control guarantees

Feedback:
TCP is a connection-oriented protocol that gives reliable, in-order delivery. It also provides congestion and flow control.

UDP is a connectionless protocol that provides a "best effort" service. It has no connection establishment (which can add delay) and no congestion/flow control.

However, neither protocol provides any delay, bandwidth or throughput guarantees.