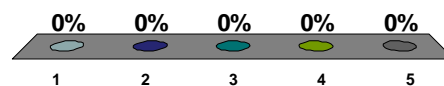


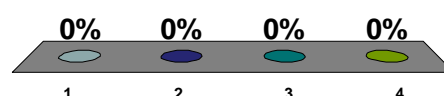
In the context of this course, what does IP stand for?

1. Internet Protocol
2. Intellectual Property
3. International Paper
4. Internet Page
5. Interesting Prof



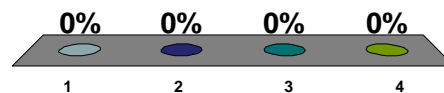
What does TCP stand for?

1. Toronto Centre for Phenogenomics
2. Transmission Control Protocol
3. Transport Control Protocol
4. Totally Crazy Prof



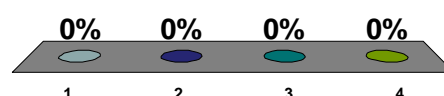
What does UDP stand for?

1. Universal Datagram Protocol
2. User Datagram Protocol
3. Universal Data Port
4. Usually Daft Prof



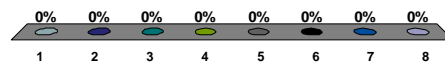
What does TFTP stand for?

1. The File Transfer Protocol
2. Terrifying File Transfer Problem
3. Trivial File Transfer Protocol
4. Terrific Fun for Terrific People



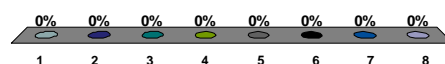
What is the top (most abstract) layer of the 5 layer protocol stack?

1. Network interface
2. Internet
3. UDP
4. TCP
5. Application
6. Transport
7. Network hardware
8. TFTP



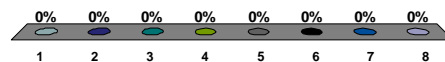
What is the second from the top layer of the 5 layer protocol stack?

1. Network interface
2. Internet
3. UDP
4. TCP
5. Application
6. Transport
7. Network hardware
8. TFTP



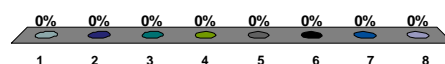
What is the middle layer of the 5 layer protocol stack?

1. Network interface
2. Internet
3. UDP
4. TCP
5. Application
6. Transport
7. Network hardware
8. TFTP



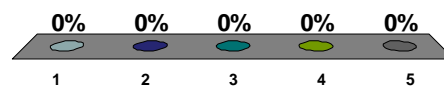
What is the second from the bottom layer of the 5 layer protocol stack?

1. Network interface
2. Internet
3. UDP
4. TCP
5. Application
6. Transport
7. Network hardware
8. TFTP



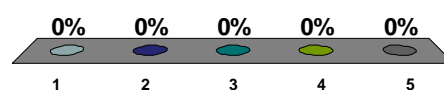
What layer does TCP implement?

1. Network interface
2. Internet
3. Application
4. Transport
5. Network hardware



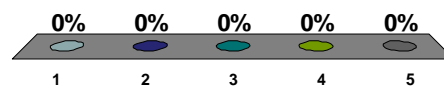
What layer does UDP implement?

1. Network interface
2. Internet
3. Application
4. Transport
5. Network hardware



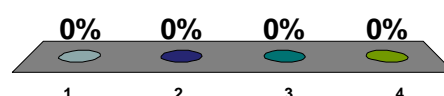
What layer does TFTP implement?

1. Network interface
2. Internet
3. Application
4. Transport
5. Network hardware



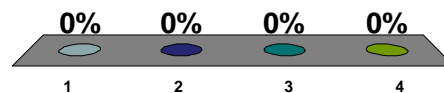
What are the characteristics of IP?

1. Virtual connection & unreliable
2. Virtual connection & reliable
3. Connectionless & unreliable
4. Connectionless & reliable



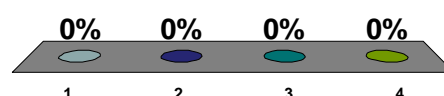
What are the characteristics of TCP?

1. Virtual connection & unreliable
2. Virtual connection & reliable
3. Connectionless & unreliable
4. Connectionless & reliable



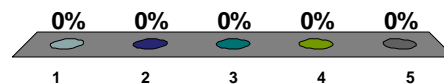
What are the characteristics of UDP?

1. Virtual connection & unreliable
2. Virtual connection & reliable
3. Connectionless & unreliable
4. Connectionless & reliable



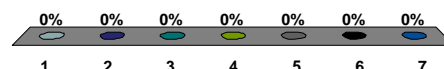
What is the main difference between IP and UDP?

1. Nothing -- they are the same
2. IP identifies a specific application and UDP a computer
3. IP identifies a computer and UDP an application
4. UDP identifies a port and IP a computer
5. IP identifies a port and UDP a computer



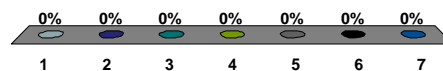
What is the name of a message passed between the application layer and the transport layer?

1. Frame
2. Transport Protocol Packet
3. Message
4. Internet Protocol Datagram
5. Stream
6. 2 and 4
7. 3 and 5



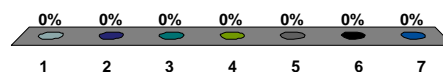
What is the name of a message passed between the internet layer and the transport layer?

1. Frame
2. Transport Protocol Packet
3. Message
4. Internet Protocol Datagram
5. Stream
6. 2 and 4
7. 3 and 5



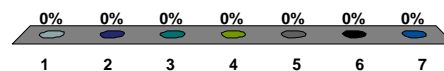
What is the name of a message passed between the internet layer and the network interface layer?

1. Frame
2. Transport Protocol Packet
3. Message
4. Internet Protocol Datagram
5. Stream
6. 2 and 4
7. 3 and 5



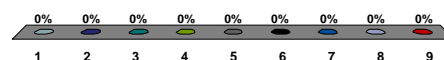
What is the name of a message passed between the network hardware layer and the network interface layer?

1. Frame
2. Transport Protocol Packet
3. Message
4. Internet Protocol Datagram
5. Stream
6. 2 and 4
7. 3 and 5



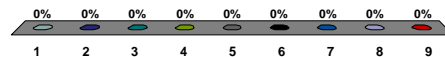
What is the name of a message passed between the TCP and IP layers?

1. TCP Packet
2. IP Packet
3. IP Datagram
4. IP Segment
5. UDP Packet
6. UDP Segment
7. UDP Datagram
8. TCP Datagram
9. TCP Segment



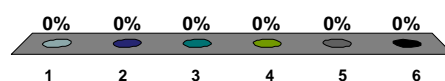
What is the name of a message passed between the IP and UDP layers?

1. TCP Packet
2. IP Packet
3. IP Datagram
4. IP Segment
5. UDP Packet
6. UDP Segment
7. UDP Datagram
8. TCP Datagram
9. TCP Segment



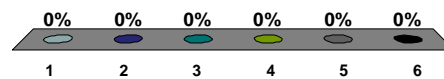
What is the name of the Java class that implements an IP address?

1. InternetAddress
2. InetAddr
3. InetAddress
4. InternetAddr
5. IPAddr
6. IPAddress



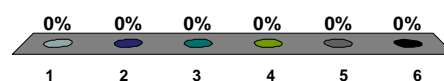
What are the names of the two Java classes that implement UDP?

1. Socket and ServerSocket
2. Socket and DatagramPacket
3. DatagramPacket and ServerSocket
4. DatagramSocket and Socket
5. DatagramSocket and DatagramPacket
6. DatagramSocket and ServerSocket



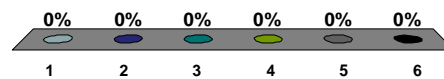
What type of threads are created directly by the programmer in Java?

1. Demon
2. Non-demon
3. Daemon
4. Non-Daemon
5. 1 and 2
6. 3 and 4



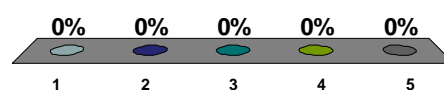
There are two ways to create a thread, they are:

1. Extend Object or implement Thread
2. Extend Thread or implement Object
3. Extend Runnable or implement Object
4. Extend Thread or implement Runnable
5. Extend Runnable or implement Thread
6. Extend Object or implement Runnable



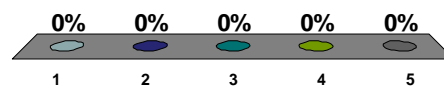
What does it mean to “Extend Thread”

1. Our new class is a subclass of the Thread class
2. Our new class implements the Thread class
3. Our new class is a superclass of Thread
4. Our new class is named Thread
5. Our new class implements the Thread interface



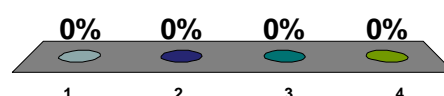
What does it mean to “Implement Runnable”

1. Our new class is a subclass of the Runnable class
2. Our new class implements the Runnable class
3. Our new class is a superclass of Runnable
4. Our new class is named Runnable
5. Our new class implements the Runnable interface



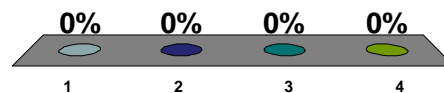
Why can't we always Extend Thread to create our new class?

1. We can -- the second way is just to make life more confusing (and quizzes more difficult)!
2. Some versions of Java don't have a Thread class
3. We may want to extend another class and Java only has single inheritance
4. Not all versions of Java allow us to subclass Thread



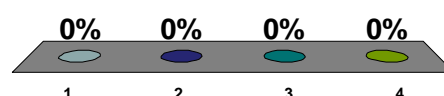
If we have created several Java threads in “main”, when will our program end?

1. When all the non-daemon threads are finished
2. When all the non-daemon threads and any daemon threads that are eligible to run have terminated
3. Never -- it will run forever
4. When the main function returns



If we have created several Java threads in “main”, when will our main program end?

1. When all the non-daemon threads are finished
2. When all the non-daemon threads and any daemon threads that are eligible to run have terminated
3. Never -- it will run forever
4. When the main function returns



If we have created several Java threads in “main”, which thread or threads are actually running at any given time?

1. All of them
2. All of them that are eligible to run (i.e. not blocked/waiting)
3. Whichever thread or threads have been chosen by the scheduler based on the processor(s) available
4. One of them, at random

