Subject:	Computer Science	**Course/Grade Level:**	Network Programming / 11th-12th
Focus Statement:	Students will demonstrate how to create network-connected programs.		

Outcome 1:

CTE.NET.1		**Students will describe the basic concepts of networks.**	
Pacing:		**Local Code:**	**Components:**
Instruct	**Assess**		**Students will:**
NA	NA	CTE.NET.1.1	Describe the protocols available in different layers of a network.
NA	NA	CTE.NET.1.2	Describe the TCP/IP four-layer model.
NA	NA	CTE.NET.1.3	Identify the quantities in a datagram.
NA	NA	CTE.NET.1.4	Identify the layer to which specific elements belong.
NA	NA	CTE.NET.1.5	Explain the Internet protocol.
NA	NA	CTE.NET.1.6	Explain how IP addresses and domain names work.
NA	NA	CTE.NET.1.7	Describe the purpose of unassigned IP addressed.
NA	NA	CTE.NET.1.8	Identify well-known default port assignments.
NA	NA	CTE.NET.1.9	Explain internet address blocks and network address translation.
NA	NA	CTE.NET.1.10	Explain the purpose of firewalls and proxy servers.
NA	NA	CTE.NET.1.11	Describe the client/server model.
NA	NA	CTE.NET.1.12	Compare and contrast the IETF and W3C.

Outcome 2:

CTE.NET.2		**Students will utilize various input and output streams in Java.**	
Pacing:		**Local Code:**	**Components:**
Instruct	**Assess**		**Students will:**
NA	NA	CTE.NET.2.1	Utilize Java's OutputStream class.
NA	NA	CTE.NET.2.2	Use the dispose pattern to clean up objects before garbage collection.
NA	NA	CTE.NET.2.3	Utilize Java's InputStream class.
NA	NA	CTE.NET.2.4	Ensure all bytes have been read when using the read method.
NA	NA	CTE.NET.2.5	Understand the purpose of a buffered stream.
NA	NA	CTE.NET.2.6	Chain filters together.
NA	NA	CTE.NET.2.7	Utilize Java's DataInputStream and DataOutputStream classes.
NA	NA	CTE.NET.2.8	Utilize Java's character readers and writers.

Outcome 3:

CTE.NET.3		**Students will demonstrate how to run and schedule threads.**	
Pacing:		**Local Code:**	**Components:**
Instruct	**Assess**		**Students will:**
NA	NA	CTE.NET.3.1	Explain the purpose of threading in a network application.
NA	NA	CTE.NET.3.2	Run a thread in Java.
NA	NA	CTE.NET.3.3	Understand how to create a subclass of the Thread class or implement the Runnable interface in Java.
NA	NA	CTE.NET.3.4	Return information from a thread in Java using a callback.
NA	NA	CTE.NET.3.5	Return information from a thread using the ExecutorService in Java.

NA	NA	CTE.NET.3.6	Utilize synchronized blocks and methods in Java.
NA	NA	CTE.NET.3.7	Describe alternatives to synchronization.
NA	NA	CTE.NET.3.8	Explain the problem of deadlock.
NA	NA	CTE.NET.3.9	Schedule threads in Java.
NA	NA	CTE.NET.3.10	Describe the eight ways a thread can pause in Java.
NA	NA	CTE.NET.3.11	Pool threads in Java.

Outcome 4:

CTE.NET.4		**Students will describe how internet addresses work.**	
Pacing:		**Local Code:**	**Components:**
Instruct	**Assess**		**Students will:**
NA	NA	CTE.NET.4.1	Understand how IPv4 and IPv6 are written.
NA	NA	CTE.NET.4.2	Describe how the Domain Name System (DNS) works.
NA	NA	CTE.NET.4.3	Utilize InetAddress objects in Java.
NA	NA	CTE.NET.4.4	Describe potential security issues with InetAddress objects.
NA	NA	CTE.NET.4.5	Describe the 10 different IP address types.
NA	NA	CTE.NET.4.6	Utilize NetworkInterface objects in Java.

Outcome 5:

CTE.NET.5		**Students will describe URLs and URIs.**	
Pacing:		**Local Code:**	**Components:**
Instruct	**Assess**		**Students will:**
NA	NA	CTE.NET.5.1	Describe the syntax of URIs and URLs.
NA	NA	CTE.NET.5.2	Explain the benefits of relative URLs.

NA	NA	CTE.NET.5.3	Describe the five pieces that compose a URL.
NA	NA	CTE.NET.5.4	Construct a URL using Java's URL class.
NA	NA	CTE.NET.5.5	Retrieve data from a URL using Java's URL class.

Outcome 6:

CTE.NET.6		**Students will describe the components of the HTTP protocol.**	
Pacing:		**Local Code:**	**Components:**
Instruct	**Assess**		**Students will:**
NA	NA	CTE.NET.6.1	Describe the four steps in an HTTP request from client to server.
NA	NA	CTE.NET.6.2	Describe common HTTP response codes.
NA	NA	CTE.NET.6.3	Explain the purpose of a "keep-alive" HTTP connection.
NA	NA	CTE.NET.6.4	Explain the purpose of the four main HTTP methods: GET, POST, PUT and DELETE.
NA	NA	CTE.NET.6.5	Describe the four required items in a POST or PUT request body.
NA	NA	CTE.NET.6.6	Explain how cookies function.
NA	NA	CTE.NET.6.7	Implement cookies in Java using the CookieManager and CookieStore classes.

Outcome 7:

CTE.NET.7		**Students will demonstrate how to connect to URLs and send and receive information.**	
Pacing:		**Local Code:**	**Components:**
Instruct	**Assess**		**Students will:**
NA	NA	CTE.NET.7.1	Communicate with a server-side program using the GET and POST methods.
NA	NA	CTE.NET.7.2	Authenticate on password-protected sites through Java's Authenticator class.
NA	NA	CTE.NET.7.3	Open URL connections using Java's URLConnection class.

NA	NA	CTE.NET.7.4	Read data from a server using the URLConnection class in Java.
NA	NA	CTE.NET.7.5	Implement caches in Java.
NA	NA	CTE.NET.7.6	Upload a file to a server using PUT in Java.

Outcome 8:

CTE.NET.8		**Students will demonstrate how to use sockets to communicate between two hosts.**	
Pacing:		**Local Code:**	**Components:**
Instruct	**Assess**		**Students will:**
NA	NA	CTE.NET.8.1	State the seven basic operation of a socket connection between two hosts.
NA	NA	CTE.NET.8.2	Write to a server using sockets.
NA	NA	CTE.NET.8.3	Describe the basic life cycle of a server program in Java.
NA	NA	CTE.NET.8.4	Utilize the ServerSockets class in Java.
NA	NA	CTE.NET.8.5	Create a server log using Java's java.util.logging package.
NA	NA	CTE.NET.8.6	Create secure socket connections using SSL.