

**Course Title: Network Programming (3 Cr.)**

**Course Code: CACS355**

**Year/Semester: III/VI**

**Class Load: 5 Hrs. / Week (Theory: 3Hrs. Practical: 2 Hrs.)**

**Course Description**

This course is designed to extend students' knowledge and practice in analysis and design of computer networks by focusing on computer network programming. It includes introduction, Internet Address, URLs and URIs, HTTP, URLConnections, Socket Programming, IP Multicast and RMI. The JAVA programming language will be used throughout the course. It does not entirely focus on theoretical concept but also strongly focuses on practical skill based knowledge.

**Course objectives**

The general objectives of this course are to provide theoretical as well as practical knowledge of network programming to make students capable of developing, implementing, managing and troubleshooting the issues of network programming in their personal as well professional life.

**Course Contents**

- |   |               |
|---|---------------|
| <b>Unit 1: Introduction</b>   | <b>3 Hrs.</b> |
| 1.1 Network Programing Features and Scope   |               |
| 1.2 Network Programming Language, Tools & Platforms   |               |
| 1.3 Client and Server Applications  |               |
| 1.4 Client server model and software design   |               |
| <b>Unit 2: Internet Addresses</b>   | <b>4 Hrs.</b> |
| 2.1 The InetAddress Class: Creating New InetAddress Objects, Getter   |               |
| 2.2 Methods, Address Types, Testing Reachability and Object Methods   |               |
| 2.3 Inet4Address and Inet6Address   |               |
| 2.4 The Network Interface Class: Factory Method & Getter Method   |               |
| 2.5 Some Useful Programs: SpamCheck, Processing Web Server Logfiles   |               |
| <b>Unit 3: URLs and URIs</b>  | <b>5 Hrs.</b> |
| 3.1 URIs: URLs and Relative URLs  |               |
| 3.2 The URL Class: Creating New URLs, Retrieving Data From a URL, Splitting a URL into Pieces, Equality & Comparison and Conversion   |               |
| 3.3 The URI Class: Constructing a URI, The Parts of the URI, Resolving Relative URIs, Equality & Comparison and String Representation |               |
| 3.4 x-www-form-urlencoded: URL Encoder and URL Decoder  |               |
| 3.5 Proxies: System Properties, The ProxyClass and The ProxySelector Class  |               |
| 3.6 Communicating with Server-Side Programs Through GET   |               |
| 3.7 Accessing Password-Protected Sites: The Authenticator Class, The PasswordAuthentication Class and The JPasswordField Class        |               |
| <b>Unit 4: HTTP</b>   | <b>2 Hrs.</b> |
| 4.1 The protocol: Keep-Alive  |               |
| 4.2 HTTP Methods  |               |
| 4.3 The Request Body  |               |



4.4 Cookies: CookieManager and CookiesStore	
<b>Unit 5: URLConnections</b>	<b>5 Hrs.</b>
5.1 Opening URLConnections	
5.2 Reading Data from Server	
5.3 Reading Header: Retrieving specific Header Fields and Retrieving Arbitrary Header Fields	
5.4 Cache: Web Cache for Java	
5.5 Configuring the Connection: protected URL url, protected boolean connected, protected boolean allowUserInteraction, protected boolean doInput, protected boolean doOutput, protected boolean ifModificationSince, protected boolean useCaches and Timeouts	
5.6 Configuring the Client Request HTTP Header	
5.7 Security Considerations for URLConnections	
5.8 Guessing MIME Media Types	
5.9 HttpURLConnection: The Request Methods, Disconnecting from the Server, Handling Server Responses, Proxies and Streaming Mode	
<b>Unit 6: Socket for Clients</b>	<b>5 Hrs.</b>
6.1 Introduction to Socket	
6.2 Using Sockets: Investigating Protocols with telnet, Reading from Servers with Sockets, Writing to Servers with Sockets	
6.3 Constructing and connecting Sockets: Basic Constructors, Picking a Local Interface to Connect From, Constructing Without Connecting, Socket Addresses and Proxy Servers	
6.4 Getting Information about a Socket: Closed or Connected?, toString()	
6.5 Setting Socket Options: TCP_NODELAY, SO_LINGER, SO_TIMEOUT, SO_RCVBUF and SO_SNDBUF, SO_KEEPALIVE, OOBINLINE, SO_REUSEADDR and IP_TOS Class of Services	
6.6 Socket in GUI Applications: Whois and A Network Client Library	
<b>Unit 7: Socket for Servers</b>	<b>5 Hrs.</b>
7.1 Using ServerSockets: Serving Binary Data, Multithreaded Servers, Writing to Servers with Sockets and Closing Server Sockets	
7.2 Logging: What to Log and How to Log	
7.3 Constructing Server Sockets: Constructing Without Binding	
7.4 Getting Information about Server Socket	
7.5 Socket Options: SO_TIMEOUT, SO_RCVBUF, SO_REUSEADDR and Class of Service	
7.6 HTTP Servers: A Single File Server, A Redirector and A Full-Fledged HTTP Server	
<b>Unit 8: Secure Socket</b>	<b>4 Hrs.</b>
8.1 Secure Communication	
8.2 Creating Secure Client Sockets	
8.3 Event Handlers	
8.4 Session Management	
8.5 Client Mode	
8.6 Creating Secure Server Socket	



8.7 Configure SSLServerSockets: Choosing the Cipher Suits, Session Management and Client Mode

**Unit 9: Nonblocking I/O**

3 Hrs.

9.1 An Example Client and Server

9.2 Buffers: Creating Buffers, Filling and Draining, Bulk Methods, Data Conversion, View Buffers, Compacting Buffers, Duplicating Buffers, Slicing Buffers, Marking and Resetting, Object Methods

9.3 Channels: SocketChannel, ServerSocketChannel, The Channels Class, Asynchronous Channels, Socket Options

9.4 Readiness Selection: The Selector Class, The SelectionKey Class

**Unit 10: UDP**

5 Hrs.

10.1 UDP Protocol

10.2 UDP Clients

10.3 UDP Servers

10.4 The DatagramPacket Class: The Constructor, The get Methods, The setter Methods

10.5 The DatagramSocket Class: The Constructor, Sending and Receiving Datagrams, Managing Connections

10.6 Socket Options: SO\_TIMEOUT, SO\_RCVBUF, SO\_SNDBUF, SO\_RUMEMADDR, SO\_BROADCAST and IP\_TOS

10.7 UDP Applications: Simple UDP Clients, UDP Server and A UDP Echo Client

10.8 DatagramChannel: Using DatagramChannel

**Unit 11: IP Multicast**

2 Hrs.

11.1 Multicasting: Multicast Address and Groups, Clients and Servers, Routers and Routing

11.2 Working with Multicast Sockets: The Constructor, Communicating with a Group

**Unit 12: Remote Method Invocation (RMI)**

2 Hrs.

12.1 Defining and Implementing RMI Service Interface

12.2 Creating an RMI Server and Client

12.3 Running the RMI System

**Laboratory Work**

Laboratory work should be done covering all the topics listed above and a small project work should be carried out using the concept learnt in this course using Java programming Language.

**Teaching Methods**

The teaching faculties are expected to create environment where students can update and upgrade themselves with the current scenario of computing and information technology with the help of topics listed in the syllabus. The general teaching pedagogy that can be followed by teaching faculties for this course includes class lectures, laboratory activity, group discussions, case studies,



guest lectures, research work, project work, assignments (Theoretical and Practical), and written and verbal examinations.

### Evaluation

Examination Scheme				
Internal Assessment		External Assessment		Total
Theory	Practical	Theory	Practical	
20	20 (3 Hrs.)	60 (3 Hrs.)	-	

### Reference Books:

1. Elliott Rusty Harold, "Java Network Programming", O'Reilly, 2014.
2. Douglas E. Comer, David L. Stevens, "Internetworking with TCP\_IP, Vol. III\_ Client-Server Programming and Applications, Linux\_Posix Sockets Version" Addison-Wesley, 2000.
3. David Reilly, Michael Reilly, "Java Network Programming and Distributed Computing", Addison-Wesley Professional, 2002
4. Kenneth L. Calvert, Michael J. Donahoo, "TCP-IP Sockets in Java. Practical Guide for Programmers", Morgan Kaufmann, 2008.
5. Andrew S. Tanenbaum, David J. Wetherall, "Computer Networks, 5/e", Prentice Hall, 2011.
6. Kurose, Ross, "Computer Networking: A Top-Down Approach", Pearson Education Limited, 2017.

