Course Outline

• Concurrent Programming:
  – Processes and threads,
  – process/thread interaction mechanisms;
  – classical problems in concurrent programming;
  – concurrent programming with threads;
Course Outline

• Network programming:
  – TCP and UDP transport,
  – Introduction to software design models:
    • Client/server, peer-to-peer etc;
    • Server designs;
    • Networking APIs;
Course Outline

• Sockets:
  – introduction to sockets,
  – socket options,
  – socket types,
  – Name and address conversions;
  – Non blocking I/O;
  – Daemons;
  – Broadcasting and multicasting;
  – Network programming with TCP/IP;
Course Outline

• Distributed Programming:
  – Middleware for distributed applications;
  – Remote Procedure Calls (RPC),
  – Distributed programming using RPC;
Course Outline

• Distributed Object based systems:
  • Distributed objects,
  • Architecture;
  • Communication;

• Distributed object programming systems:
  – Java RMI, CORBA, DCOM;
  – Distributed object based programming using Java RMI and CORBA.
Course Outline

• Pre-requisites
  – CSC221 Object-oriented Analysis, Design and Programming
  – CSC223 Operating Systems
  – CSC225 Computer Networks
  – CSC315 Distributed Systems
Course Outline

• Delivery
  – Lectures
  – Tutorials and
  – Lab sessions

• Core Texts
  – UNIX Network Programming Volume 1: The Sockets Networking API by W.R. Stevens
  – UNIX Network Programming Volume 2: Interprocess communication by W.R. Stevens
Course Outline

• Assessment
  – Examination – 50%
  – Coursework – 50%

• Lectures and Lab sessions
  – Attendance
  – Electronic handouts provided
  – Linux Computing Environment
  – Projects
Administrative

• Collaboration and Cheating
  – Discussions
  – Hand in only your own work
  – Writeups must be completed independently