CGS Agenda Item: 17-23
Effective Spring 2018

Eastern Illinois University
New Course Proposal
CSM 4980, Networking and Distributed Computing

Banner/Catalog Information (Coversheet)

1. **_X_ New Course or _____Revision of Existing Course**

2. **Course prefix and number:** CSM 4980

3. **Short title:** Networking

4. **Long title:** Networking and Distributed Computing

5. **Hours per week:** 3 Class 0 Lab 3 Credit

6. **Terms:** __Fall X Spring ___ Summer ___ On demand

7. **Initial term**: _ Fall X Spring ___ Summer Year: 2018

8. **Catalog course description:** An overview of the concepts and algorithms in networking and distributed computing. Topics include protocol stacks, link, network, transport, and application layers, network management, network architecture, network abstractions, distributed algorithms, distributed environments, the client-server model, the peer-to-peer model, and remote procedure calls. There will be case studies and analysis of existing systems.

9. **Course attributes:**

   General education component: ____None_____

   ___ Cultural diversity ___ Honors ___ Writing centered ___ Writing intensive ___Writing active

10. **Instructional delivery**
    Type of Course:

    X Lecture ___ Lab ___ Lecture/lab combined ___ Independent study/research

    ___ Internship ___ Performance ___ Practicum/clinical ___ Other, specify: _____

    Mode(s) of Delivery:

    X Face to Face ___ Online ___ Study Abroad

    ___ Hybrid, specify approximate amount of on-line and face-to-face instruction_____

11. Course(s) to be deleted from the catalog once this course is approved. None

12. **Equivalent course(s):** None

    a. Are students allowed to take equivalent course(s) for credit? ___ Yes X No

13. **Prerequisite(s):** C or better in CSM 3980

    a. Can prerequisite be taken concurrently? ___ Yes X No

    b. Minimum grade required for the prerequisite course(s)? *C*

c. Use Banner coding to enforce prerequisite course(s)? X Yes ___ No

d. Who may waive prerequisite(s)?

___ No one X Chair X Instructor ___ Advisor ___ Other (specify)

14. Co-requisite(s): None

15. Enrollment restrictions

a. Degrees, colleges, majors, levels, classes which may take the course: All

b. Degrees, colleges, majors, levels, classes which may not take the course: None

16. Repeat status: X May not be repeated ___ May be repeated once with credit

17. Enter the limit, if any, on hours which may be applied to a major or minor: No limit

18. Grading methods: X Standard _CR/NC __Audit ___ ABC/NC

19. Special grading provisions:

___ Grade for course will not count in a student's grade point average.

___ Grade for course will not count in hours toward graduation.

___ Grade for course will be removed from GPA if student already has credit for or is registered in: _____

___ Credit hours for course will be removed from student's hours toward graduation if student already has credit for or is registered in: _____

20. Additional costs to students:
Supplemental Materials or Software None

Course Fee X No ___Yes, Explain if yes_____

21. Community college transfer:

___ A community college course may be judged equivalent.

X A community college may not be judged equivalent.

Note: Upper division credit (3000+) will not be granted for a community college course, even if the content is judged to be equivalent.

Rationale, Justifications, and Assurances (Part I)

1. X Course is required for the major(s) of Computer Science

   ___Course is required for the minor(s) of _____

   ___Course is required for the certificate program(s) of _____

   Course is used as an elective:

2. **Rationale for proposal**:

   Networking is a fundamental part of modern computer science and many applications use networking in some form. The ubiquity of network access makes the study of networks and distributed computing necessary for computer science students.

3. **Justifications for (answer N/A if not applicable)**

   Similarity to other courses: There is some similarity to MIS3200, which has the catalog description "Study of business telecommunications, teleprocessing, and computer networks. Installing, configuring and administering local area networks." and MIS4700, which has the catalog description "Internetworking concepts, advanced internetwork design, security, and administration. Enterprise network design, management and user administration. WAN data analysis and troubleshooting." However, both of these classes have a focus on the configuration and administration of networks and no focus on distributed computing. The proposed course includes distributed computing and primarily deals with the analysis of networks and the algorithms utilizing them, not the administration.

   Prerequisites: Networking protocols and distributed computing inherently makes use of concurrency as studied in CSM 3980. As such, the understanding gained from CSM 3980 is essential to success in this course.

   Co-requisites: N/A

   Enrollment restrictions: N/A

   Writing active, intensive, centered: N/A

4. **General education assurances (answer N/A if not applicable)**

   General education component: N/A

   Curriculum: N/A

Instruction: N/A

Assessment: N/A

5. Online/Hybrid delivery justification & assurances (answer N/A if not applicable)

Online or hybrid delivery justification: N/A

Instruction: N/A

Integrity: N/A

Interaction: N/A

Model Syllabus (Part II)

Please include the following information:

1. Course number and title

CSM 4980 Networking and Distributed Computing

2. Catalog description

An overview of the concepts and algorithms in networking and distributed computing. Topics include protocol stacks, link, network, transport, and application layers, network management, network architecture, network abstractions, distributed algorithms, distributed environments, the client-server model, the peer-to-peer model, and remote procedure calls. There will be case studies and analysis of existing systems.

3. Learning objectives.

Students will be able to:

- Understand the layered structure of a typical networked architecture [QR-4, CT-3, GGL-1, GGL-2]
- Identify the different types of complexity in networks (edges, core, etc.) [QR-4, QR-5, GGL-1]
- Implement simple client-server socket-based applications [QR-6, GGL-1, GGL-2]
- Understand how packets are forwarded in an IP network [CT-3, GGL-1]
- Understand the scalability benefits of hierarchical addressing [QR-4, CT-3, GGL-1, GGL-2]
- Understand how frames are forwarded in an Ethernet network [QR-4, CT-3, GGL-1]
- Understand and Implement at least one approach to the multiple access problem [QR-4, QR-6, CT-3, GGL-1, GGL-2]
- Compare and contrast current approaches to congestion [QR-4, QR-5, CT-6, GGL-1, GGL-2, GGL-3]
- Explain why synchronization constructs such as simple locks are not useful in the presence of distributed faults [QR-4, QR-5, CT-6, GGL-1, GGL-2, GGL-3]
- Write a program that performs any required marshaling and conversion into message units, such as packets, to communicate interesting data between multiple hosts [QR-6, GGL-1, GGL-2]
- Measure the observed throughput and response latency across hosts in a given network [QR-1, GGL-1, GGL-2]
- Explain why no distributed system can be simultaneously consistent, available, and partition tolerant [QR-4, QR-5, CT-6, GGL-1, GGL-2, GGL-3]

- Implement a simple server [QR-6, GGL-1, GGL-2]
- Explain the tradeoffs among overhead, scalability, and fault tolerance when choosing a stateful v. stateless design for a given service [QR-4, QR-5, CT-6, GGL-1, GGL-2, GGL-3]

4. Course materials.

Computer Networks, Fifth Edition: A Systems Approach, Larry L. Peterson, Bruce S. Davie

Distributed Systems: Principles and Paradigms 2nd Edition, Andrew S. Tanenbaum and Maarten Van Steen

5. Weekly outline of content.

Week 01: Overview of Networking and Distributed Computing, common APIs
Week 02: Measuring network performance, classes of links, reliable transmission
Week 03: Basic internetworking, routing, routing performance
Week 04: Advanced internetworking, multicasting
Week 05: End-to-end protocols
Week 06: Remote procedure calls, Exam 1
Week 07: Client-server model, peer-to-peer model
Week 08: Congestion-avoidance, quality of service
Week 09: Logical time, vector clocks
Week 10: Global state and snapshot algorithms
Week 11: Message ordering, Exam 2
Week 12: Termination detection
Week 13: Distributed mutual exclusion algorithms
Week 14: Deadlock detection in distributed systems
Week 15: Distributed shared memory
Week 16: Final exam

6. Assignments and evaluation, including weights for final course grade.

Weekly Homework (10%)
Programming Assignments (30%)
Two in-class Exams (30%)
Final Exam (30%)

Graduate students will be expected to do the same work as the undergraduates. In addition, the exercises and projects will include more challenging problems, focusing on theoretical exercises requiring additional reading. Weekly homework assignments will include two additional graduate-student-only exercises, and exams will include an additional graduate-student-only question.

7. Grading scale.

Final grades will be given on a standard A, B, C, D, F scale:

90-100% A 80-89% B 70-79% C 60-69% D Below 60% F

8. Correlation of learning objectives to assignments and evaluation.

	Weekly Homework (10%)	Programming Assignments (30%)	Two in-class Exams (30%)	Final Exam (30%)
Understand the layered structure of a typical networked architecture [QR-4, CT-3, GGL-1, GGL-2]	x	x	x	x
Identify the different types of complexity in networks (edges, core, etc.) [QR-4, QR-5, GGL-1]	x		x	x
Implement simple client-server socket-based applications [QR-6, GGK-1, GGL-2]		x		
Understand how packets are forwarded in an IP network [CT-3, GGL-1]	x	x	x	x
Understand the scalability benefits of hierarchical addressing [QR-4, CT-3, GGL-1, GGL-2]	x	x	x	x
Understand how frames are forwarded in an Ethernet network [QR-4, CT-3, GGL-1]	x	x	x	x
Understand and Implement at least one approach to the multiple access problem [QR-4, QR-6, CT-3, GGL-1, GGL-2]		x		
Compare and contrast current approaches to congestion [QR-4, QR-5, CT-6, GGL-1, GGL-2, GGL-3]	x		x	x
Explain why synchronization constructs such as simple locks are not useful in the presence of distributed faults [QR-4, QR-5, CT-6, GGL-1, GGL-2, GGL-3]	x		x	x
Write a program that performs any required marshaling and conversion into message units, such as packets, to communicate interesting data between multiple hosts [QR-6, GGL-1, GGL-2]		x		
Measure the observed throughput and response latency across hosts in a given network [QR-1, GGL-1, GGL-2]	x	x		
Explain why no distributed system can be simultaneously consistent, available, and partition tolerant [QR-4, QR-5, CT-6, GGL-1, GGL-2, GGL-3]	x		x	x
Implement a simple server [QR-6, GGL-1, GGL-2]		x		
Explain the tradeoffs among overhead, scalability, and fault tolerance when choosing a stateful v. stateless design for a given service [QR-4, QR-5, CT-6, GGL-1, GGL-2, GGL-3]	x		x	x

Date approved by the department or school: November 28, 2016
Date approved by the college curriculum committee: January 20, 2017
Date approved by the Honors Council *(if this is an honors course):*
Date approved by CAA: February 23, 2017 CGS: