Knight Foundation School of Computing and Information Sciences

Course Title: Advanced Unix Programming **Date:** 04/04/2012

Course Number: COP 4604

Number of Credits: 3

Subject Area: Computer Systems	**Subject Area Coordinator:** Shu-Ching Chen **email:** chens@cis.fiu.edu
Catalog Description: Unix overview: files and directories, shell scripting and systems programming; Unix tools; Internals: file systems, process structure; Using the system call interface; Interprocess communication.	
Textbook: Advanced Programming in the UNIX Environment – Second Edition Richard Stevens and Stephen Rago Addison Wesley (ISBN: 0201433079)	
References: Unix for the Impatient, 2nd Edition Abrahams and Larson Addison Wesley (ISBN: 0201823764)	
Prerequisites Courses: COP 4338	
Corequisites Courses: COP 4610	

Type: Elective for CS (Systems group).

Prerequisites Topics:

- C programming and pointers
- Multithreading and serialization

Course Outcomes:

1. Mastery of the basic UNIX process structure and the UNIX file system
2. Mastery of simple UNIX filters
3. Familiarity of UNIX pipes and redirection, UNIX environment, traps, signals, filter parameters, filter options, UNIX contentions, and Regular Expressions
4. Mastery of at least one Shell scripting language
5. Familiarity of Perl scripting and C systems programming
6. Familiarity with Interprocess Communication using pipes, shared memory, semaphores and messages

Knight Foundation School of Computing and Information Sciences
COP 4225
Advanced Unix Programming

Relationship between Course Outcomes and Program Outcomes

BS in CS: Program Outcomes	Course Outcomes
a) Demonstrate proficiency in the foundation areas of Computer Science including mathematics, discrete structures, logic and the theory of algorithms	
b) Demonstrate proficiency in various areas of Computer Science including data structures and algorithms, concepts of programming languages and computer systems.	1,2,3
c) Demonstrate proficiency in problem solving and application of software engineering techniques	4,5,6
d) Demonstrate mastery of at least one modern programming language and proficiency in at least one other.	4,5
e) Demonstrate understanding of the social and ethical concerns of the practicing computer scientist.	
f) Demonstrate the ability to work cooperatively in teams.	
g) Demonstrate effective communication skills.	

Assessment Plan for the Course & how Data in the Course are used to assess Program Outcomes

Student and Instructor Course Outcome Surveys are administered at the conclusion of each offering, and are evaluated as described in the School's Assessment Plan: https://abet.cs.fiu.edu/csassessment/

Knight Foundation School of Computing and Information Sciences
COP 4225
Advanced Unix Programming

Outline

Topic	Number of Lecture Hours	Outcome
• Using UNIX o UNIX history, basic concepts, getting started o basic operations on files, file tools, editors o networks and communications	10	2,3
• Programming o Shell programming: Bourne, csh o UNIX utilities: grep, etc. o Perl scripting o System programming in C	12	4,5
• Systems programming o file i/o, files and directories o process control o interprocess communication	14	1,6

Course Outcomes Emphasized in Laboratory Projects / Assignments

	Outcome	Number of Weeks
1	Statistical evaluation of Unix files Outcome: 1,2,3	2
2	Shell script design Outcomes: 4	2
3	Perl script and C system programming Outcomes: 5	2
4	Interprocess communication Outcomes: 6	2

Oral and Written Communication

No significant coverage

Written Reports		Oral Presentations	
Number Required	Approx. Number of pages	Number Required	Approx. Time for each
0	0	0	0

Knight Foundation School of Computing and Information Sciences
COP 4225
Advanced Unix Programming

Social and Ethical Implications of Computing Topics

No significant coverage

Topic	Class time	student performance measures

Approximate number of credit hours devoted to fundamental CS topics

Fundamental CS Area	Core Hours	Advanced Hours
Algorithms:		**1.0**
Software Design:		**1.0**
Computer Organization and Architecture:		**1.0**
Data Structures:		
Concepts of Programming Languages		

Theoretical Contents

Topic	Class time
Boolean algebra	1.0

Problem Analysis Experiences

1. Semaphores

Solution Design Experiences

1. Regular Expression Construction
2. Design of Scripts

Knight Foundation School of Computing and Information Sciences
COP 4225
Advanced Unix Programming

The Coverage of Knowledge Units within Computer Science Body of Knowledge[1]

Knowledge Unit	Topic	Lecture Hours
OS8	File structure, file I/O, basic operations on File	6
OS12	Scripting, passing parameters to scripts, shell scripts, awk and perl scripts.	8

[1]See https://www.acm.org/binaries/content/assets/education/cs2013_web_final.pdf for a description of Computer Science Knowledge units