

LSC Use Only
Number _____
Action _____
Date _____

UWUCC USE Only
Number #21
Action _____
Date _____

CURRICULUM PROPOSAL COVER SHEET
University-Wide Undergraduate Curriculum Committee

I. TITLE/AUTHOR OF PROPOSAL

Course/Program Title: Co 362 Unix and C
Suggested 20 character course title: Unix and C
Department: Computer Science
Contact Person: John A. Cross x4478
E-Mail: jacross

II. If a course, is it being proposed for:

- ☒ Course Approval/Revision Only
☐ Course Approval/Revision and Liberal Studies Approval
☐ Liberal Studies Approval Only (course previously approved by Senate.)

III. APPROVALS

John A. Cross
Department Curriculum Committee
Chae H. Kim Fe B
College Curriculum Committee

Gary L. Bertubing
Department Chairperson
W. J. Calan
* College Dean

Director of Liberal Studies
(where applicable)

Provost (where applicable)

* EACH COLLEGE DEAN MUST CONSULT WITH THE PROVOST BEFORE APPROVING CURRICULUM CHANGES. APPROVAL BY COLLEGE DEAN INDICATES THE PROPOSED CHANGE IS CONSISTENT WITH LONG RANGE PLANNING DOCUMENTS, ALL REQUESTS FOR RESOURCES IN THE PROPOSAL CAN BE MET, AND THE PROPOSAL HAS THE SUPPORT OF THE UNIVERSITY ADMINISTRATION.

III. TIMETABLE

Date Submitted:
to LSC _____
to UWUCC _____

Semester to be
implemented: _____

Date to be
published
in Catalog _____

CATALOG DESCRIPTION

CO 362 - UNIX* and C

3c-01-3sh

Prerequisites: CO 310 or Permission of Instructor

This course provides a general introduction to UNIX and C including historical roots, current practice, prospects for the future, and hands-on skills. Students will develop understanding of the features, syntax, and "culture" of UNIX and C through lectures, reading, and projects. UNIX and C will be studied both separately and in synergistic combination. Upon completion of the course, students will have immediately practical awareness and some skills. They will also be provided with a foundation for further coursework in operating systems, compiler construction, and/or graduate study.

*UNIX is a trademark of Bell Laboratories. UNIX is an operating system and C is a system programming language.

DESCRIPTION OF CURRICULAR CHANGE**I. Catalog description - attached.****II. Course syllabus****A. Catalog Description - attached.****B. Course Objectives.**

1. Students will develop knowledge and skills appropriate for current practice in both the use of UNIX and C and system design and implementation decisions relative to UNIX and C technology. This will include an understanding of basic terminology, design concepts, language syntax and technical features of UNIX and C.
2. Students will demonstrate an understanding of the history, technological context, and culture of UNIX and C which is appropriate for interpreting current events and making educated guesses about future events.
3. Students will be prepared for and intrigued by undergraduate, graduate or other continuing study in operating systems and/or system programming.
4. Students will be introduced to state-of-the-art topics in powerful system programming environments. At this time these are windows, multitasking, workstations, the object-oriented programming paradigm and open systems.

C. Detailed Course Outline.**1. UNIX (15 lecture hours and 1 exam hour)**

- | | | |
|----|--|-------------|
| a. | History and Introduction | <1.0 hour> |
| b. | Getting Started | <1.5 hours> |
| | 1. Keyboards and mice | |
| | 2. Logging on and off | |
| | 3. Passwords | |
| | 4. Help | |
| | 5. The "ed" line editor | |
| | 6. The "emacs" editor (see c below) | |
| | 7. Regular expressions | |
| | 8. Listing the current directory | |
| | 9. Deleting, renaming, and copying files | |
| | 10. Mail | |
| c. | The "emacs" and "vi" editor | <1.5 hours> |
| d. | Basic utilities | <1.0 hours> |

- e. System File Structure <1.0 hours>
 - 1. Hierarchical organization of directories
 - 2. Terminology
 - 3. Directory Commands
 - 4. Access restrictions
- f. UNIX Shells as Programming Languages <4.0 hours>
 - 1. Versions: Bourne, C, Korn, Bash
 - 2. Commands
 - 3. Standard Input and Output
 - 4. Pipes
 - 5. Filters
 - 6. Running a program in the background
 - 7. Special characters in file names
- g. More on the emacs editor <1.0 hour>
- h. More on Utilities <2.0 hours>
- i. Discretionary time and topics <2.0 hours>
- 2. The C Programming Language (15 lecture hours and 1 exam hour)
 - a. Getting Started <1.0 hour>
 - 1. A first program, printf
 - 2. Compile and execute
 - 3. Examples of variables, expressions, and assignment
 - 4. #define, scanf, while
 - 5. A program with input and a loop
 - 6. Redirecting the standard I/O files
 - b. Fundamentals <1.5 hours>
 - 1. Character set and language elements
 - 2. Identifiers and keywords
 - 3. Data types, conversions, casts
 - 4. Operators and separators
 - 5. Comments and constants
 - 6. Strings
 - 7. Exercise on operators
 - i. Precedence
 - ii. Increment
 - iii. Implicit operand
 - c. Flow of Control <1.5 hours>
 - 1. Conditional expressions
 - 2. Compound statements
 - 3. The null statement
 - 4. if statements
 - 5. Looping
 - i. while
 - ii. for
 - iii. do
 - 6. Sample programs
 - d. Functions and Program Structure <1.5 hours>
 - 1. Function invocation and definition
 - 2. Call by value / call by address
 - 3. Program structure

- 4. Builtin functions
 - e. Character Processing <0.5 hours>
 - f. The Arithmetic Data Types <1.0 hours>
 - g. Branching and Nested Flow of Control <1.5 hours>
 - 1. The break and continue statements
 - 2. The switch statement for cases
 - 3. Nested flow of control
 - 4. The conditional operator
 - 5. Storage classes
 - h. Functions and Pointers:
 - Call by Reference <1.0 hour>
 - i. Arrays <1.5 hours>
 - j. Recursion <2.0 hours>
 - k. Structures and Dynamic Storage
 - Allocation . . . <2.0 hours>
 - 3. Unix and C: Synergism (10 lecture hours)
 - a. C in a UNIX environment <1.0 hours>
 - i. cb "prettyprint"
 - ii. make and makefile for developing modularized code
 - iii. lint static code checker
 - iv. prof run-time analysis
 - b. File I/O <1.0 hours>
 - c. Gnu software and the Free Software Foundation <0.5 hours>
 - d. System Administration <1.0 hours>
 - e. C++ <1.5 hours>
 - f. awk <2.0 hours>
 - g. Discretionary time for extra lectures as needed and additional topics such as multitasking, multiuser, and real-time applications; and current awareness.
 - 4. Final Examination <1.0 hour>
- D. Evaluation Methods.

Points will be awarded for student performance on out-of-class assignments, in-class activities, two one-hour exams, and a final exam. Exam 2 will not be comprehensive, but the Final Exam will be comprehensive. There will be roughly 325 points for examinations and 225 points for other graded activities, especially hands-on projects. Point scores will be used to determine grade cutoffs. The grade cutoffs will reflect the instructor's best judgment of what letter grade a particular point total represents. Grade cutoffs will be no higher than 90%-80%-70%-60% of the possible point total for grades A through D. A course grade will be determined by each student's point total relative to the grade cutoffs, plus the personal judgment of the instructor for students who are at a borderline score.

The primary factor for deciding a borderline score will be exam scores; a secondary factor will be evidence of intellectual involvement with the concepts of the course as evidenced from

classroom participation. However, because of the importance of hands-on experience with UNIX and C in this course, satisfactory completion of ALL hands-on projects will be required for a passing grade.

Students may check their grades throughout the academic session by using the "grade-check" program on the academic vax. Students will receive direction on how to use this program in class. Note that a "percentile" score is a ranking, and your overall "percent" score is what your instructor bases your grade on. The final course grade will also be available through grade-check.

E. Required Textbook, Supplemental Books and Readings.

Kelley, Al, and Pohl, Ira. A Book On C. Benjamin/Cummings Publishing Company, Reading MA (1989).

Sobell, Mark G. A Practical Guide to the UNIX System, Second Edition. Benjamin/Cummings, Reading, MA (1989).

F. Special Resource Requirements.

No additional costs for the student are anticipated. IUP will provide students with access to all the necessary computer hardware and software. No special laboratory fee is associated with this course. Students will run UNIX and C on the Microvax II in Stright 107. Ultrix, a DEC version of UNIX, will be installed on the Microvax for the duration of this course. Students will also be provided with an academic VAX account for mail and grade-checks. Students may also run C on the academic VAX or "Turbo C" on the Computer Science microcomputer network. The academic VAX also has a UNIX "shell" which may be used for class activities.

The Computer Science Department encourages students to acquire sufficient personal computing resources to access the computing facilities for this course through the IUP Chief network. Your instructor, Microcomputer Support Services (X4000, Stright G-1), and the IUP Co-op Bookstore will provide free consultation to students who are considering the purchase of personal computing resources.

G. Bibliography.

Burkhard, Walter A. C For Programmers. Wadsworth, Belmont, CA, 1988.

Jalics, Paul J., and Heines, Thomas S. "Transporting a Portable Operating System: UNIX to An IBM Minicomputer." Communications of the ACM (CACM). December 1983, Volume 26, Number 12, pp. 1067-1073.

Kelley, Al, and Pohl, Ira. C By Dissection. Benjamin/Cummings Publishing Company, Reading MA (1989).

Kernighan, B.W., and Ritchie, D.M. The C Programming Language, Second Edition. Prentice-Hall, Englewood Cliffs, NJ, 1988. <This is the best-known book among professionals. It is an excellent book, and it is only slightly harder to learn from than the required text.>

MIPS, The Magazine of Intelligent Personal Systems. January, 1989.

Mullish, Henry, and Cooper, Herbert L. The Spirit of C: An Introduction to Modern Programming. West Publishing, New York, NY, 1987.

O'Malley, Christopher. "UNIX, Tomorrow's Operating System?" Personal Computing, June 1988, pp. 100-108.

Peters, James F., III. UNIX Programming Methods and Tools. Harcourt Brace Jovanovich, Publishers, New York, NY, 1988.

Ritchie, Dennis M., and Thompson, Ken. "1983 Turing Award Lectures." Communications of the ACM (CACM). August 1984, Volume 27, Number 8, pp. 758-763.

Tondo, Clovis L., and Gimpel, Scott E. The C Answer Book, Second Edition. Prentice Hall, Englewood Cliffs, New Jersey, 1989. <Note: this book contains solutions to exercises in Kernighan & Ritchie.>

III. COURSE ANALYSIS QUESTIONNAIRE

A. Details of the Course

1. This course fulfills the academic need for students to know the terminology, technical details, practical skills, system considerations, and historical context of Unix and C. Many professional computing activities require specific knowledge of Unix and C, the industry trend is toward more Unix and C applications, and most graduate programs require them. This course is designed as a Junior-level elective for Computer Science Majors and Minors and MIS majors. This course is

NOT proposed for inclusion in any Liberal Studies course list.

2. This course is not designed as a prerequisite to other courses and no other courses or programs will require changes in their content. We specifically considered the desirability of using Unix and C in other upper-level courses, especially CO 420 Modern Programming Languages, CO 424 (* not yet in the catalog *) Compiler Construction, CO 430 System Programming, and CO 432 Operating Systems. However, we were unable to work out a way to require it as a prerequisite or teach Unix and C in those courses.
3. This is a traditional type of course for the Computer Science Department. Students learn by lectures, reading, and doing projects. Up to 40% of the grade may be based on hands-on projects.
4. This course was offered in Spring 1989 as a 1-credit Special topics and in Spring 1990 as a 3-credit Special Topics. In both cases, this course was the first of our Special Topics courses to "fill" during early registration.
5. This course is NOT proposed as a dual-level course.
6. This course may NOT be taken for variable credit.
7. The University of Pittsburgh and Johns Hopkins University offer similar courses.
8. The content of this course cannot be incorporated into existing courses because there are more than 3 credits worth of things to learn in this course and because our existing courses already have plenty of concerns other than Unix and C. Note that Unix and C are practically an entry level requirement for graduate study in computing.

B. Interdisciplinary Implications

1. This course will be taught by one instructor.
2. Unix and C are notoriously "unfriendly". Therefore, this course would be a poor choice for a technically inexperienced student. The course prerequisite is designed to address this concern (see A1).
3. This description of this course has similarities to a Special Topics course which has been offered many times by the Finance and MIS Department. The MIS course emphasizes

UNIX system administration concerns, while the CO course covers shell programming, variant shells, C, UNIX and C together, and C++.

4. The course can be taught in late afternoon to facilitate enrollment by students with other commitments than IUP courses. If two sections are scheduled, seats can be made available to students in the School of Continuing Education.

C. Implementation

1. a. Faculty - Professors Cross, Micco, Wolfe and Watts are currently competent to teach the course. Other faculty could prepare themselves, so that faculty preparedness is not a problem. However, this course represents an additional "preparation" and requires a staffing commitment of 6 credits per year. Our current complement of 11 faculty is adequate. The lab manager is also a critical resource for this course: The student and faculty facilities for hands-on use of Unix and C must be administered by someone whose job description includes responsibility for system management. Our current "lab manager" position is adequate - but it is also essential.
- b. Space - student facilities must be in a public area with extended evening and weekend hours. This space must be staffed with a consultant who can at least "logon" to Unix and point students to appropriate system documentation. The Unix and C facility must be accessible through the "Chief", which it is. The class must be scheduled in a room where the instructor can demonstrate Unix and C.
- c. Equipment - at least 10 terminals must be available in one location for students. Dial-in access through the Chief currently exists. The current facility is adequate, but the technology is evolving and enhancements will be requested. In particular, Unix "workstations" are clearly more desirable than using Ultrix on the Microvax II. We will attempt to acquire at least six (6) such workstations (currently about \$10,000 each), but CURRENT FACILITIES ARE MINIMALLY ADEQUATE.
- d. Laboratory Supplies and other Consumable Goods - printer paper and ribbons - no significant cost and no problem.
- e. Library Materials - current arrangements are adequate.

- f. Travel Funds - Unix and C are rapidly evolving. This implies that the instructor must travel and the lab manager will have to go to occasional classes in system administration. We are currently managing to do some travel and training, but everyone at IUP must recognize that travel to keep a professor in touch with rapidly evolving technology (and related theory) is an essential part of our mission. We will continue to argue for increased financial support for faculty travel and recognition of scholarly travel in the evaluation of faculty performance.
- 2. No current resources which are essential to this course are funded by a grant. We may make use of our current supercomputing grants at the Pittsburgh Supercomputing Center, but these activities are not essential to the course.
- 3-4. We expect to offer two (2) sections of this course every Fall semester.
- 5. Twenty-five (25) students per section is a workable number. We tried thirty (30), but this places a harmful limitation on the instructor's ability to schedule occasional hands-on activities, interact on technical details in class, and consult with students outside of class.
- 6. No quotes are available from the pertinent groups: the Association for Computing Machinery (ACM), the Computer Society of the Institute for Electrical and Electronic Engineers (IEEE), and the Computer Sciences Accreditation Board (CSAB). However, the ACM plans to release a recommendation for "Curricula 90" in the next year with specific statements about the content of this course.
- 7. This is an elective course.

D. Miscellaneous

None attached.

LETTERS OF SUPPORT

A general letter was sent to the Finance and MIS Department in August, 1990. The MIS program in the College of Business has offered a similar course as a Special Topics. We have provided them with a copy of this proposal and we have invited them to respond. Their informal response is that the two courses are significantly different because they want to emphasize system administration concerns and de-emphasize programming

CO 362, Unix and C

Page 10

concerns.

CURRICULAR OFFERING/CHANGE AUTHORIZATION FORM

Attached.