
CS 277

Systems Programming

Spring 2006
MWF 2:20 - 3:30*
HOS-123)

<http://www.cs.moravian.edu/cs277>

CS 277. Systems Programming (1u).
C language syntax and structure. C programming techniques. Emphasis on structured design for medium to large programs. Unix operating system fundamentals. Unix utilities for program development, text processing, and communications. Prerequisite: CS 120.

Instructor

Stephen Corbesero
PH-213, 610-625-7786
corbesero@cs.Moravian.edu
<http://www.cs.moravian.edu/corbesero>

Goals

1. To learn the facets of the Unix operating system as a software development platform.
2. To learn and apply the C language in application and systems programming contexts.
3. To be able compare and contrast Unix operating systems features with other operating systems.

Text

The required texts for the course are Unix in a Nutshell: System V Edition, 4/e by Gilly and Advanced Programming in the Unix Environment, 2/e by Stevens, and the C Pocket Reference from O'Reilly.

In addition, a good ANSI C reference is The C Programming Language, 2/e by Kernighan and Ritchie.

Assignments, Programs, and Tests

Homework and programming assignments will mostly consist of C programs and Unix shell scripts. Tests will include of three hour quizzes, a final, and many short quizzes.

Computer Resources

The primary computer resource will be the various Mo-CoSIN Sun workstations running the Solaris version of the

Unix operating system, but students are free to utilize other (equivalent) computers for developing their programming assignments. However, all programming assignments, unless explicitly stated otherwise, must work correctly and be submitted on the Suns.

Prerequisites

Students are expected to have a fundamental background in programming and computer usage. Some basic understanding of machine organization and architecture understanding (binary, bits, bytes, addresses, etc.) is also helpful

Grading Scale

Homework and Quizzes	15 %
Programs	35
Hour Quizzes	25
Final	25
Total	100 %

Grading Policies

- Attendance is strongly encouraged, and pop quizzes, which would count in the homework category, have spontaneously occurred. **YOU ARE RESPONSIBLE FOR EVERYTHING THAT OCCURS IN CLASS.** A grade may be changed up to two weeks after an assignment, program, or test is returned. After the final exam, no grades may be contested.
- Short quizzes will be given on every Friday, unless there is an hour quiz. These quizzes should take no more than 10 minutes, and are CLOSED BOOK, CLOSED NOTES.

Homework

Each homework will be graded out of a possible 100 points. Late homework will be penalized according to scale for programs, below, unless a solution has been posted or presented in class, then there will be no credit.

Programs

- Each program will typically be graded out of 100 points, and then the score will be weighted to reflect its complexity. Programming assignments will consist of small (10-200 lines), medium (200-600), and large (600+ lines) endeavors.
- Programs will be graded on correctness, style, and documentation.

- Late programs will be accepted but will be penalized according to the following table.

on-time	score modifier
no more than 1 class-day late	100 %
up to 1 class-week late	90 %
> 1 class-week late	50 %
	ZERO

- Unless explicitly stated otherwise, programs are due electronically at 11:59pm on the due date.
- I will report to your advisor as soon as you fail to submit a second programming assignment, and I will continue to report for each one for each missed program thereafter.

Tests

No makeup exams will be given. Students missing an hour quiz, in a properly excusable fashion, will be graded out of the remaining percentages. Unless explicitly stated otherwise, the hour quizzes and final are open book, open notes.

Important Dates

Jan 16	M	First day of classes Martin Luther King Day
Jan 24	T	Last Day to Add/Drop
Feb 10	F	Hour Exam I
Feb 24	F	Midterm
Mar 5–12	S – U	Spring Break
Mar 17	F	Hour Exam II
Mar 31	F	Last Day to Withdraw with a W
Apr 4	T	College Day of Service
Apr 12	W	Hour Exam III
Apr 14–17	F–M	Easter Break
Apr 28	F	Last Day of Classes
Apr 30	S	Reading Day
May 1–6	M – S	Final Examinations
May 13	S	Commencement

Policy on Academic Dishonesty

All work, unless explicitly stated in the problem definition, is to be an individual effort. Students are encouraged to work together so long as the final submission has a single, clearly identifiable author. If necessary, violations of this will be dealt with as a case of academic dishonesty. Please refer to the “Collaboration Policy” statement,

The Dean of Students has far more experience handling these matters, so I reserve the right to assign all cases to him.

Policy on Computer Abuse

Overall, I encourage creative and curious play on computer systems in general, but I will not tolerate flagrant abuses of other user’s rights. Any such cases will be dealt with very severely.

W and WP/WF Grades

If a student drops a course after the last day to withdraw with a “W”, I must assign a “WP” or a “WF” to indicate whether the student was passing or failing at the date of withdrawal. When making that judgment, I will take into account factors such as class attendance, program submission punctuality, etc.

Topics

- C programming
 - Syntax, statements, data types, operators
 - The standard libraries
 - Separate compilation
 - Machine architecture concerns
- Software development issues
 - Text editing (ed, ex, vi, emacs)
 - Automatic program testing
 - Source tools
 - * Maintenance (indent, cxref, cflow)
 - * Debugging (gdb, gprof)
 - * Version control (RCS)
- Unix operation
- Shell programming
- Unix/C utilities
 - make
 - m4
 - lex/flex, yacc/bison
- Text Processing
 - awk, sed, diff, grep, perl
 - nroff, troff, tbl, eqn, pic
 - T_EX, L_AT_EX
- Unix systems programming
 - System calls
 - Signals
 - Processes and inter-process communication
 - Files and filesystems