Redesigning Computer Programming I Using Multi-level Online Modules for a Mixed Audience

Nira Herrmann, Jeffrey Popyack, Bruce Char, Paul Zoski, Chris Cera

Department of Mathematics and Computer Science, Drexel University, Philadelphia PA 19104

Supported by Drexel University,
the Ramsey McCluskey Family Foundation,
the National Science Foundation (DUE 0089009), and
the Pew Learning and Technology Program of the Center for Academic Transformation

The Drexel Environment

- All students required to have own computer since 1983-84
- Huge growth in computer science and related majors
  - §1996-97: 300 CS majors + 280 IS majors
  - §2000-01: 825 CS majors + 710 IS majors + 20 Digital Media majors + 456 CE majors
- Moderate growth in faculty
  - § 1996-97: 9.5 FTE in CS 2000-01: 13 FTE in CS
- Increasingly varied student backgrounds in computing

Instructional Objectives for Redesign

- Personalize course presentation for individual students
- Minimize resources, particularly faculty, needed in course
- Create modules with multiple knowledge levels to serve multiple constituencies for the course: CS, CE, DM, IS
- Allow for multiple entry points into course material depending on previous knowledge

Old versus New Class Organization

Old course
2 lecture hours + 1 lab hour per week
Individual lab projects in a standard computer lab
Some online materials (instructor-created)

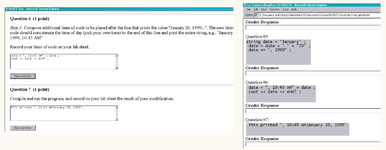
New course
1 lecture hour + 1 lab hours per week
Individual and group lab projects
Substantial on-line support: chat groups, asynchronous discussion, on-line assignment submission

Modular Organization

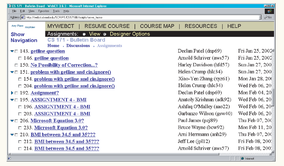
- Course material divided into modular units
- 3 modules equivalent to 1 credit of course work
- Typical courses are 3 credits: 9 modules per full course (10 week term)
- Modules to include
  - § pre-recorded lectures on module topics
  - § homework assignments
  - § lab assignments, group and individual
  - § enrichment materials

The course on WebCT

- WebCT with links to separately hosted course Web site
- Campus Pipeline will be Drexel campus portal to WebCT, Student Info. System (BANNER)
  - § Automatically creates webCT accounts for registered students
  - § Allows seamless communication between Drexel's record system and course WebCT site
- Automated Grading Support Software
  - § Performs rapid downloads of assignments from WebCT
  - § Sorts assignments by lab section for TAs to grade
  - § Grades can be sorted by lab or lecture section
- WebCT Functionality
  - § Online submission of assignments
  - § Online labs and quizzes



Lab worksheets are filled out and graded on-line in webCT.

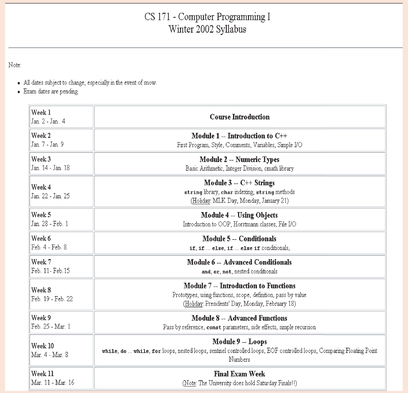


WebCT supports chat rooms and on-line discussion groups

Plans for evaluation

Pilot Study
- Format of pilot
  - § Two traditional lecture sections
  - § One redesign section
  - § Same faculty teaching both groups
- Common quizzes and exams
- Common individual assignments
- Redesign section also does group assignments
- Evaluation will include
  - § Comparison of exam performance
  - § Follow-up on performance in sequel course
  - § Standard student course evaluations
  - § Evaluation by participating faculty
  - § Focus groups to get student feedback
- Institutional Review Board review
  - § Informed consent for focus groups
  - § Anonymity of evaluations

Course Syllabus



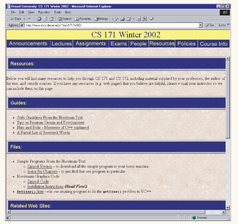
CS 171 - Computer Programming I
Winter 2002 Syllabus

Note:
- All dates subject to change, especially in the event of snow.
- Exam dates are pending.

Week 1 Jan. 2 - Jan. 4	**Course Introduction**
Week 2 Jan. 7 - Jan. 9	**Module 1 – Introduction to C++** First Program, Style, Comments, Variables, Simple I/O
Week 3 Jan. 14 - Jan. 18	**Module 2 – Numeric Types** Basic Arithmetic, Integer Division, cmath library
Week 4 Jan. 22 - Jan. 25	**Module 3 – C++ Strings** `string` library, `char` indexing, `string` methods (Holiday: M.L.K. Day, Monday, January 21)
Week 5 Jan. 28 - Feb. 1	**Module 4 – Using Objects** Introduction to OOP, Howtoman classes, File I/O
Week 6 Feb. 4 - Feb. 8	**Module 5 – Conditionals** `if`, `if - else`, `if - else if` conditionals,
Week 7 Feb. 11- Feb. 15	**Module 6 – Advanced Conditionals** `and`, `or`, `not`, nested conditionals
Week 8 Feb. 19 - Feb. 22	**Module 7 – Introduction to Functions** Prototypes, using functions, scope, definition, pass by value (Holiday: Presidents' Day, Monday, February 18)
Week 9 Feb. 25 - Mar. 1	**Module 8 – Advanced Functions** Pass by reference, `const` parameters, side effects, simple recursion
Week 10 Mar. 4 - Mar. 8	**Module 9 – Loops** `while`, `do - while`, `for` loops, nested loops, nested controlled-loops, EOF controlled loops, Comparing Floating Point Numbers
Week 11 Mar. 11 - Mar. 16	**Final Exam Week** (Note: The University does hold Saturday Finals!)

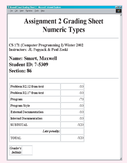
Lectures

A page of a lecture notes. These are used as presentation slides in the live and recorded lectures.



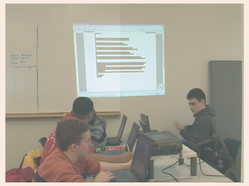
Lectures and other course materials are available from the class web site.

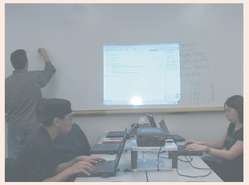
Assignments

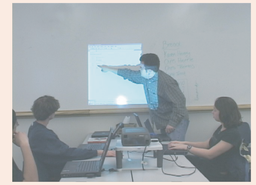


Conventional assignments are posted in webCT. Students submit their solutions electronically to webCT. They are timestamped and available for upload by graders working for the course. An coversheet generator for the course (shown above) allows the instructor to standardize the marking information

Supervised group problem-solving sessions







Wireless networked laptop computers in clusters of five, with one projector per cluster. The walls are covered with "whiteboard wallpaper" for board work and projection.