R2R
THE ROADMAP
TO REDESIGN

Precalculus: Virginia Tech

(For a list of topics covered in the course, please see http://www.theNCAT.org/R2R/AcadPrac/Topics/VT_Precalc_Top.pdf.)

Precalculus is a one-semester, three-credit course that uses the following materials:

Commercial Materials (Optional)

- Software: *Mathematica*

Publisher: Wolfram Research
http://www.wolfram.com/

Description: *Mathematica* is a software package that seamlessly integrates a numeric and symbolic computational engine, graphics system, programming language, documentation system, and advanced connectivity to other applications. *Mathematica* notebooks form the basis for the precalculus course at Virginia Tech. Student learning exercises are recorded in these notebooks and then processed into web pages via an engine developed at Virginia Tech. Although students are not required to purchase this package to take this course, they are encouraged to do so.

How to obtain: A variety of licensing and purchasing options are available. See *Mathematica* web site for more details. At Virginia Tech, students purchase *Mathematica* through a subsidized student license.

Materials developed by Virginia Tech

- **Supplemental Lectures**

Description: Thirty-one videotaped lectures describe course concepts and calculations. Each 15-minute lesson was written and recorded by Virginia Tech faculty. The tapes are available via an index using QuickTime streaming video.

Technical Requirements: QuickTime Player and web browser with broadband internet.

How to preview: http://www.emporium.vt.edu/math1015/qtvideos.html

How to obtain: Contact John Rossi at Rossi@calvin.math.vt.edu.

- **Lesson Pages**

Description: Forty interactive lesson pages present concept information and problems to be solved by the students. Each lesson includes a written discussion of the topic, exercises to work through, a lesson problem on that topic and links to supplemental lectures.

Technical Requirements: Web browser with QuickTime plug in and an xml/xslt-like engine to fill this role. When edits to online materials are necessary, only the original *Mathematica* notebooks are modified (no HTML editing). Affected lesson pages are automatically regenerated. Thus error correction and updating or improving lessons can be done routinely. The pages themselves require no more than generic javascript and style-sheet support.

How to preview: http://www.emporium.vt.edu/math1015/1015/Resources/lessons/index.html

How to obtain: Contact John Rossi at Rossi@calvin.math.vt.edu.

- **Quiz and Exam Questions**

Description: All quizzes and exams were written at Virginia Tech. Because the testing engine generates questions based on the objectives for that quiz or exam and because the problem values are generated randomly, the same question bank is used for practice and graded quizzes and proctored exams. See below for a description of the testing engine.

Technical Requirements: Web browser and ID for access.

How to preview: Contact John Rossi at Rossi@calvin.math.vt.edu.

How to obtain: Contact John Rossi at Rossi@calvin.math.vt.edu.

- **Testing System and Core Testing Engine**

Description: The testing system at Virginia Tech was designed to scale for very large numbers of students and to provide concurrent testing using an SQL database. (Over 550,000 assessments were administered during the fall 2003 semester.) The testing system has 4 main components: 1) core engine, 2) data storage, 3) problem generation and 4) reporting.

The core engine has the following functions:

1) Delivery of an assessment to a student browser, subject to its defining parameters (e.g. proctored or not, location restriction, type (practice, quiz, test, etc), time restrictions, etc.) It is managed through entities called “tickets.”
2) Management of assessment in progress including information about the state of the test at any moment. The usual functions of navigation and security are handled here as well.
3) Scoring of tests and display of results.
4) Recording of all student activities to a permanent database. Once results data have been entered, they may not be altered.
5) Student logins using university-wide IDs (interacts with the enterprise directory)
6) Management of test queues of unused assessments for all the course sections registered to the system.
7) Display of past (i.e. taken) assessments on demand by student.
8) Addition and modification of assessment tickets for student access per instructor requests.
9) Addition and modification of student enrollment data.

Technical Requirements: Java program with server-side java pages.

How to preview: Contact John Rossi at Rossi@calvin.math.vt.edu.

How to obtain: Contact John Rossi at Rossi@calvin.math.vt.edu.

- **Data Storage**

Description: All student information, course and section data, unused and used assessments, ticket data regarding the delivery of assessments and access data are stored in an SQL database (Oracle) with no outside access (test engine access only).

Technical Requirements: SQL Oracle database.

How to preview: Contact John Rossi at Rossi@calvin.math.vt.edu.

How to obtain: Contact John Rossi at Rossi@calvin.math.vt.edu.

- **Problem Generation**

Description: The problems that make up the assessments are not the conventional type normally found in test databanks. Stored questions are “question stubs'” that are actually programs. Whenever a question corresponding to a particular stub is called for by the assessment creation process, the stub is fetched and executed creating an instance of the question. These are assembled into graded and practice quizzes and other assessments according to definitions provided by the instructor. Stubs typically have millions of variations. The question stub programs have been carefully written to test the particular goals and objectives which define the course content. The question stub programs are written so that they must be worked out afresh in each instance. Thus, the same stubs can be used for practice quizzes, quizzes, tests and exams. All questions are multiple choice. This component is implemented using *Mathematica*. Because the testing system is not documented, its portability is very limited.

Technical Requirements: Contact John Ross at Rossi@calvin.math.vt.edu.

How to preview: Contact John Rossi at Rossi@calvin.math.vt.edu.

How to obtain: Contact John Rossi at Rossi@calvin.math.vt.edu.

- **Reporting**

Description: Student test results are extracted to a reporting system and processed using logic and weighting prescribed by each section instructor. These data are stored in a separate SQL database (Oracle) and available on demand by students or instructors. Any adjustments instructors wish to make are done here rather than within the test engine proper. Student standing is always current and always available.

Technical Requirements: Web browser and SQL Database.

How to preview: Contact John Rossi at Rossi@calvin.math.vt.edu.

How to obtain: Contact John Rossi at Rossi@calvin.math.vt.edu.

- **Course Web Page**

Description: The instructor in each course/section maintains a course web page which includes course schedules and deadlines, supplementary resources (if available), information about help sessions and other information which is current for the course. Everything here is stored in ordinary web pages.

Technical Requirements: Internet browser with web access and web server.

How to preview: http://www.emporium.vt.edu/math1015/index.html

How to obtain: Contact John Rossi at Rossi@calvin.math.vt.edu.