Computer Science Department
cs.salemstate.edu

SYLLABUS **Fall, 2012**

CSC 278 Scripting Techniques **4 credits**
Prerequisite(s): CSC 202J.

Instructor: Beifang Yi **Office**: MH 211A **Phone**: (978) 542-7246
email: byi@salemstate.edu **Hours**: T & R (10:50-1:30) **Web Site**: http://cs.salemstate.edu/~byi/
T (4:30-6:00pm, 10:00pm-10:30pm), W & F (12:50-1:20)

Section	Time	Room	Final Exam
01	W & F 3:00pm-4:15pm	MH 206	**Wednesday 12/19, 11:00am-1:00pm** **MH 206**
L21	W & F 1:30pm-2:45pm	MH 202	

Catalog description:

This course presents rapid application development techniques (RAD) and their implementation using modern scripting languages. It examines methods for defining problems and their solutions, including task analysis and the development of design criteria. The course investigates the design of modern scripting languages, emphasizing the use of their particular attributes for developing solutions to complex problems. Three hours of lecture and three hours of scheduled laboratory per week, plus programming work outside of class.

Course Goals:

The purpose of this course is for students to:

- CG01: develop a greater understanding of the software development process;
- CG02: understand the role of scripting languages in software development;
- CG03: present the use of scripting languages in developing web applications;
- CG04: examine the use of scripting languages for the rapid development of scientific applications;
- CG05: implement one or more large-scale projects using a scripting language.

Course Objectives:

Upon successful completion of the course, a student will be able to:

- CO1: determine if a scripting language is appropriate for a particular problem and if so, to select a specific language that is well suited to the particular problem;
- CO2: select appropriate software engineering methods and RAD techniques for developing applications using a scripting language;
- CO3: select appropriate algorithms, data structures, and language features for the solution of a complex problem;
- CO4: understand and employ common types of code reuse;
- CO5: understand and utilize open-source modules for complex problem solving in computational science or artificial intelligence;
- CO6: produce clear documentation for problems assigned in the course and their solutions.

Course Topics:

- Software engineering environment and methodologies
    - The development of task requirements and software specifications.
    - Introduction to effective methods for software development and prototyping.
    - Tools for project planning and revision control

    - Review programming language development, with an emphasis on the need for scripting languages. Compare and contrast high level languages, such as C++ and Java, with scripting languages, such as JavaScript and Python.

- Developing Web-Base Applications Using Scripting Languages
    - The Document Object Model (DOM)
    - DOM Scripting
    - Frames, forms, cookies, and alarms
    - Processing events
    - Dynamic HTML

- Developing Scientific Applications Using Scripting Languages
    - Tools for application development: IDE's, interpreters, compilers, and applet builders
    - Examination of a popular language currently used for scientific applications
    - Cross-platform GUI Development
    - Multi-threading using scripting languages
    - According to student and instructor interest, examine an application of scripting techniques in one or more of the following areas:
        - Artificial intelligence
        - Image processing and computer vision
        - Computer graphics/gaming
        - Visualization
        - Data processing for the world-wide web applications
        - Scientific computing using existing libraries

Text(s): (required)

- **Learning Perl the Hard Way**, Allen B. Downey, Green Tea Press, www.greenteapress.com/perl/index.html
- **Object-Oriented JavaScript: Creating scalable, reusable high-quality JavaScript applications and libraries,** Stoyan Stefanov, Packt Publishing, 2008. (ISBN 1847194141).
- **The Quick Python Book**, Daryl Harms, Kenneth McDonald, Manning Publications, 2011 (ISBN 1884777740).

Additional references:

- **Dive Into Python 3**, Mark Pilgrim, Create Space, 2010. (ISBN: 1441437136). *Available cheap at bookstore*.
- **DOM Scripting: Web Design with JavaScript and the Document Object Model**, Jeremy Keith, friends of Ed, 2005. (ISBN: 1590595335).
- **Pro JavaScript Design Patterns,** Ross Harmes, Dustin Diaz, 2008. (ISBN-13: 978-1-59059-908-2).
- **Secrets of the JavaScript Ninja,** John Resig, 2010. (Available electronically from manning.com as an "early release").
- **jQuery UI 1.7: The User Interface Library for jQuery,** Dan Wellman, Packt Publishing, 2009. (ISBN: 1847199720).
- **Programming in Python 3: A Complete Introduction to the Python Langauge**, Mark Summerfield. Electronic Products, Pearson, 2009. (ISBN-13: 978-0-13-712929-4).
- **Beginning Python Visualization: Crafting Visual Transformation Scripts,** Shai Vaingast, Apress, 2009. (ISBN-13: 978-1-43-021843-2).

Useful Web Sites:

- www.w3schools.com/ A website where you can learn nearly everything about web-related topics
- www.gossland.com Tutorial on Perl for Windows
- www.packtpub.com Publisher for required book; get an ebook here.
- www.diveintopython3.org html and pdf versions of required book.
- www.netbeans.org IDE software used in course.
- www.enthought.com Visualization software used in course
- www.python.org The official Python site
- www.sagemath.org Open source mathematics software written in Python
- http://icant.co.uk/articles/from-dhtml-to-dom/ From DHTML to DOM scripting
- http://www.scribd.com/doc/82260230/Secrets-of-the-Javascript-Ninja Secrets of the JavaScript Ninja
- http://getpython3.com/#resources Python free resources
- http://www.apachefriends.org/en/xampp.html A webserver solution package

- http://www.webbasedprogramming.com/ Web based Programming Tutorials (Perl, MySQL, PHP,…)
- http://www.webbasedprogramming.com/Perl-5-By-Example/ Perl 5 By Example
- http://people.csail.mit.edu/pgbovine/python/ Learn Python in the Brower
- http://pyx.sourceforge.net/ Python Graphics: PyX
- http://www.pygame.org/news.html PyGame
- http://rene.f0o.com/mywiki/PythonGameProgramming Python Game Programming in 6 hours
- http://anh.cs.luc.edu/python/hands-on/3.1/handsonHtml/index.html Hands-on Python
- http://cgkit.sourceforge.net/ Python Computer Graphics Kit

Software:

- Stawberry Perl 5.12.2. Installed on laboratory machines and available free from www.perl.org/downloads.
- Python 3.1.2. Installed on laboratory machines and available free from www.python.org/downloads.
- Enthought software. Installed on laboratory machines and available free from www.enthought.com.

Cell phones:

Turn the ringer off, or, better yet, ***turn the phone off***.

Class Attendance:

Class attendance is highly recommended. You are responsible for all materials presented in class, quizzes, examinations, and other announcements. No excuses of any nature will be construed as relieving you from the responsibility for completion of the work assigned. Each student is responsible for completing all course requirements and for keeping up with all that goes on in the course.

Scheduled Lab Attendance:

Regular laboratory attendance and participation is ***strongly*** **recommended**. Lab will be used to review or present software tools, to discuss and investigate Java implementation details that time may not permit to be fully explored during the scheduled lecture period, for design and implementation drills, for occasional short lab (programming) exercises, to assist with design and debugging problems that arise in longer lab / project exercises, and to check/examine/grade the exercises and homework. Each student is responsible for completing all course requirements and for keeping up with all that goes on in the laboratory portion of the course regardless of absences.

Final Grade:

Final grade will be determined using the following grading weights:

Programming/project assignments	40%
semester project	15%
quizzes	10%
midterm examination	10%
final examination	25%

Attendance is not used to calculate the final grade: Please note that you are at all times responsible for assignments and materials presented in class.

The following table shows how the course work is assessed against the course objectives:

Assessment Mechanism / Course Objective matrix

	Programming/Project Assignments	Midterm Examination	Semester Project	Final Examination
CO01	✓	✓	✓	✓
CO02	✓	✓	✓	✓
CO03	✓	✓	✓	✓
CO04	✓	✓	✓	✓
CO05	✓		✓	
CO06	✓		✓	

Programming/Project Assignments:

Programming languages are best learned by doing. The (programming) project assignments are intended to exercise concepts and programming constructs presented in class. These assignments will be distributed either in class and or the laboratory period, they will also be sent out to the class email list, and posted on the class's Canvas site.

Semester Project:

One programming project, larger in scope than the assignments, will be assigned as the Semester Project. Students can choose any topics and programming languages introduced in the class as their final/semester projects upon the *instructor's permission*. For this assignment, the following materials should be submitted by their indicated due dates respectively : (1) semester project proposal (a brief description of which topic and what it to be done and expected results), (2) semester project progress report (of what has been done and what will be done and temporary results), and (3) semester project report (in addition to the class presentation, students are required to submit a formal project write-up/report, programming code, and presentation documents).

Submission Deadlines/Late Penalties:

There are specific due dates/times for any assignments (written assignments, programming projects, semester project) and these assignments should be completed by the deadlines. **A penalty of 2%** will be applied for late submission for each day (including weekends and holidays). No missed presentations will be made up unless under extreme circumstances with advanced notification of the instructor and/or certain supporting documentation.

All the assignments must be submitted at Canvas—the course online management system. Students may not turn in your assignment through emails or in printed copy format. Canvas will keep your last submission (usually, your new submission will automatically be replaced by the old one for the same assignment; thus it is the ***latest*** submission that will be graded by the instructor, whether it has been submitted on-time or late.). Extension of deadlines will be granted under exceptional circumstances (such as medical emergencies) and with supporting documents—in such cases, students must obtain emails from the instructor as the proof of deadline extensions.

All the assignments will be announced/given in class and/or through course website.

Semester project presentation cannot be made later than the presentation time except under emergent situations.

No assignment submissions will be accepted after the final examination.

Exams/Quizzes:

The midterm will be held approximately around week 8 depending on class progress. Check the final examination time and place indicated above (the table at the beginning of the syllabus). . **Note:** Make-ups are given for examinations only under exceptional circumstances and with documented circumstances.

Missed Tests:

Tests (exams and quizzes) may not be made up except for *documented emergency* situations. If a test must be made up, arrangements must be made with the instructor to take the test before it is discussed in class (usually within a week of the test being administered). Students who know in advance that they must be absent on an exam day for an excusable reason should notify the instructor prior to the exam day. Students who are absent on the day of the exam for an excusable reason should contact the instructor immediately following their absence. Makeup work will be permitted *only when* the instructor is presented with acceptable documentation for acceptable absences. It is your responsibility to notify your instructor of any excused absence as far in advance as possible.

Due Dates:

- **There will be a 2% penalty for each day (including weekends, holidays) an assignment (projects, assignments, etc.) is late**; penalties accrue at 12:00 midnight of the assigned due date.
- **No assignments will be accepted after the final examination**.

Study Groups:

While I strongly encourage study groups, I require that each student hand in his/her answers in her/his own words - if two answers are highly similar to each other, neither will receive credit.

When working on your programming projects, you may discuss with others the project topics, the algorithms and methodologies related to the project; but when you work on writing the code, this coding work should be 100% of your own work. **If two answers/written code segments come out exactly the same or highly similar, neither will receive credit and/or further actions will be taken** (such as reporting to the department and/or university). Given the nature of most of the projects, homework questions and writing assignments, it will be almost impossible for two people to come up with highly similar answers UNLESS they copy.

Academic Integrity:

Academic Integrity Policy and Regulations can be found in the University Catalog and on the University's website (http://catalog.salemstate.edu/content.php?catoid=13&navoid=1295#Academic_Integrity). The formal regulations are extensive and detailed - familiarize yourself with them if you have not previously done so. A concise summary of and direct quote from the regulations: "Materials (written or otherwise) submitted to fulfill academic requirements must represent a student's own efforts". *Submission of other's work as one's own without proper attribution is in direct violation of the University's Policy* and will be dealt with according to the University's formal Procedures.

"Salem State University is committed to providing equal access to the educational experience for all students in compliance with Section 504 of The Rehabilitation Act and The Americans with Disabilities Act and to providing all reasonable academic accommodations, aids and adjustments. Any student who has a documented disability requiring an accommodation, aid or adjustment should speak with the instructor immediately. Students with Disabilities who have not previously done so should provide documentation to and schedule an appointment with the Office for Students with Disabilities and obtain appropriate services."

In the event of a University declared critical emergency, Salem State University reserves the right to alter this course plan. Students should refer to http://www.salemstate.edu for further information and updates. The course attendance policy stays in effect until there is a university declared critical emergency. In the event of an emergency, please refer to the alternative educational plans for this course located at http://cs.salemstate.edu/~byi/2012Fall/CSC278/emergency/index.html. Students should review the plans and gather all required materials before an emergency is declared.

Please remember that if, for any reason, you decide to drop this course, you **MUST** do so officially through the Registrar's office. The last day to withdraw from a course this semester is **November 26th**.

Note: This syllabus represents the intended structure of the course for the semester. If changes are necessary, students will be notified in writing and via all regular class communication mechanisms (class discussion, emails, and/or the instructor’s website at http://cs.salemstate.edu/~byi/).