CSC 310.010 - Procedure Oriented Programming Languages, Spring, 2009

CSC 310.010 meets MWF 9-9:50 in Old Main 140.
Dr. Dale E. Parson, Old Main 252, parson@kutztown.edu, 484-646-4296
Office Hours: M 5-6, Tu 10-11, W 10-12, Th 2-3, and by appointment

Topics covered include basic concepts and special facilities in higher level programming languages, including scope of declarations, storage allocations, data structure variety, binding time of constituents, and control organization with emphasis on programming in several representative languages. **Prerequisite:** CSC 237.


**Grading** (on an A-90, B-80, C-70, D-60 scale)

<table>
<thead>
<tr>
<th>Component</th>
<th>Percentage of Grade</th>
</tr>
</thead>
<tbody>
<tr>
<td>Midterm</td>
<td>25% of grade</td>
</tr>
<tr>
<td>Non-cumulative final</td>
<td>25% of grade</td>
</tr>
<tr>
<td>Programs</td>
<td>50% divided equally among the programming assignments</td>
</tr>
</tbody>
</table>

**Programming assignment grading criteria**

- 5% Concise, accurate documentation of the operations of an abstract data type interface signature, preconditions, postconditions, invariants, appropriate use of Javadoc
- 10% Clean separation of specification of interface from concrete module implementations
- 5% Documentation of client modules (including tests) that use ADT modules
- 30% Correct implementation of central data structure or algorithm
- 20% Adherence to written specification of assignment, including coding standards
- 5% Willingness to ask questions and clarify any uncertainties about assignments
- 5% Concise, structured, and accurate documentation of implementation
- 10% Inclusion of test code, data & results showing coverage of common and corner cases
- 10% Successful execution against my private evaluation test cases

*The academic integrity policy is at* [http://cs.kutztown.edu/pdfs/AcademicIntegrityPolicy.pdf](http://cs.kutztown.edu/pdfs/AcademicIntegrityPolicy.pdf)  
*Your first reading assignment is to read the above policy statement.  
You may openly discuss ideas, algorithms, pitfalls, and the use of programming tools.  
You may not share code, test drivers or test data except within groups for group projects.  
Group projects, when assigned, have documented partitioning of student responsibilities.  
I will be using the Moss detection system at [http://theory.stanford.edu/~aiken/moss/](http://theory.stanford.edu/~aiken/moss/).*

Class attendance is not mandatory, but I will be teaching fundamentally from experience, using the textbook and some slides as additional sources of information. You are responsible for all material covered in class, including technical information, coding standards and conventions, verbal specification of assignments, and your questions about topics that are not clear to you. Please, there should be no classroom conversations, cell phones, text messaging, eating, sleeping, listening to music or other disruptions of the class. **Consistent lateness to class or disruptive or obscene language will result in a reduction to a student’s grade.** Thank you.
There are five programming assignments in the plan.

P₁ requires you to write a multi-function Python program that utilizes run-time interpretation.

P₂ requires you to use a parser generator to create a parser for translating sentences in a grammar.

P₃ requires you to construct, maintain and use symbol table data structures for the symbols of P₂.

P₄ requires you to construct a virtual machine that can run P₂/P₃ programs.

P₅ involves extending P₂/P₃/P₄ to support additional data types and referencing environments.

Python’s home is at http://www.python.org/. Any of the 2.x versions is OK. Do not use version 3!

/opt/csw/bin/python on bill is Python version 2.5.1. Assignments must run on bill.