Name of Instructor: Mario A. Garcia  
Course title: Survey of Programming Languages  
Course number: COSC 3353  
Office phone number: 825 3478  
E-mail address: mario.garcia@tamucc.edu  
Office number and building: Center for Instruction 331  
Office hours: M W F: 9.00 – 10.30  
MWF 8.00 – 8.50 Am BH 223  

Course Description  
A study of selected programming languages for students familiar with programming.  
Students will write programs in a broad variety of languages. Prerequisite COSC 2437  

Students Learning Outcomes:  
After completing this course, the student should be able to:  
Have a higher-level understanding of traditional and non-traditional programming  
language constructs  
1. Comprehend and Apply Functional programming languages (LISP)  
2. Comprehend and Apply Web programming languages (Ruby, Perl, PHP, JavaScript)  
3. Comprehend and Apply Imperative programming languages (Python)  
4. Comprehend and Apply Multi-paradigm programming languages C#  
5. Comprehend and Apply Visual Languages – Visual Basic  
6. Comprehend and Apply with Mac – Objective C  

Graded Activity  

<table>
<thead>
<tr>
<th>Assessment Mechanism</th>
<th>Percentage</th>
</tr>
</thead>
<tbody>
<tr>
<td>LISP exam</td>
<td>25%</td>
</tr>
<tr>
<td>Selected Programming Language Group Presentation</td>
<td>25%</td>
</tr>
<tr>
<td>Programming Assignments</td>
<td>50%</td>
</tr>
</tbody>
</table>

Grading Scale  

<table>
<thead>
<tr>
<th>Grade</th>
<th>Scale</th>
</tr>
</thead>
<tbody>
<tr>
<td>A</td>
<td>90-100%</td>
</tr>
<tr>
<td>B</td>
<td>80-89%</td>
</tr>
<tr>
<td>C</td>
<td>70-79%</td>
</tr>
<tr>
<td>D</td>
<td>60-69%</td>
</tr>
<tr>
<td>F</td>
<td>0-59%</td>
</tr>
</tbody>
</table>

Grading Notes and Comments  

Incompletes: An incomplete will only be granted in the case of serious illness. Written proof of the illness and a recommendation for an incomplete will be required from both the Dean of Students office as well as from a doctor. A grade of incomplete is never issued to give a student more time to complete assignments or improve a grade. The final determination as to whether or not an incomplete should be issued rests solely with the professor.  
Note: An 89 is a B
*Notice to Students with Disabilities*: Texas A&M University-Corpus Christi complies with the Americans with Disabilities Act in making reasonable accommodations for qualified students with disabilities. If you suspect that you may have a disability (physical impairment, learning disability, psychiatric disability, etc.), please contact the Services for Students with Disabilities Office, located in Driftwood 101, at 825-5816. If you need disability accommodations in this class, please see me as soon as possible.

**ACADEMIC ADVISING**: The College of Science and Technology requires that students meet with an Academic Advisor as soon as they are ready to declare a major. The Academic Advisor will set up a degree plan, which must be signed by the student, a faculty mentor, and the department chair. The College's Academic Advising Center is located in Driftwood 203E, and can be reached at 825-3466.

*** Grade Appeal Process. As stated in University Rule 13.02.99.C2, Student Grade Appeals, a student who believes that he or she has not been held to appropriate academic standards as outlined in the class syllabus, equitable evaluation procedures, or appropriate grading, may appeal the final grade given in the course. The burden of proof is upon the student to demonstrate the appropriateness of the appeal. A student with a complaint about a grade is encouraged to first discuss the matter with the instructor. For complete details, including the responsibilities of the parties involved in the process and the number of days allowed for completing the steps in the process, see University Rule 13.02.99.C2, Student Grade Appeals, and University Procedure 13.02.99.C2.01, Student Grade Appeal Procedures. These documents are accessible through the University Rules Web site at [http://www.tamucc.edu/provost/university_rules/index.html](http://www.tamucc.edu/provost/university_rules/index.html). For assistance and/or guidance in the grade appeal process, students may contact the Office of Student Affairs.

Optional Texts
3. The little lisper second edition. Daniel P. Friedman and Matthias Fellisen. SRA

COURSE OUTLINE
Introduction -
Topic 1: LISP August 24 - September 23
    LISP Exam – September 26

Topic 1: Ruby
    Programming Language Class Presentation: October 3

Topic 2: Python
    Programming Language Class Presentation: October 10
    Ruby programming assignment due. October 10
Topic 3: Perl
  Programming Language Class Presentation: October 17
  Python programming assignment due. October 17

Topic 4: C#
  Programming Language Class Presentation: October 24
  Perl programming assignment due. October 24

Topic 5: PhP
  Programming Language Class Presentation: October 31
  C# programming assignment due. October 31

Topic 6: Visual Basic
  Programming Language Class Presentation: November 7
  PhP programming assignment due. November 7

Topic 7: Objective C
  Programming Language Class Presentation: November 14
  Visual Basic programming assignment due. November 14

Topic 8: Java Script
  Programming Language Class Presentation: November 21
  Objective C programming assignment due. November 21

**Group Presentation (25%)**
- Each team will consist of two students.
- Each team will have to learn about the programming language assigned including its history, main application areas and the main instructions to use the language: Read, Write, Storage (arrays), control structures, decisions, repetitions, arithmetic operations.
- Each team will prepare a powerpoint for presentation according to the schedule. The powerpoint must include all the items described in the previous bullet.
- Each team must find three programming assignments where all the instructions (previous bullet) have to be applied. The problems must be approved by the instructor.

**Programming Assignments. (50%)**
Each team of two students will apply the concepts of Extreme programming to implement the programming assignments that each group will assign. Extreme programming is based on designing, implementing, testing, in groups of two. There is going to be one week to complete each programming assignment after each presentation. Only one solution will have to be uploaded in the assignments section of Blackboard.