Foundations in Programming and Problem Solving I

Course Description:
Catalog: A concentrated introductory programming course at the graduate level. Intended for students with little background in computer science who wish to program a computer in support of research or other academic interests.

Detailed: An introduction to Computer Science. Many important concepts underlying computer science are covered. This includes the algorithmic foundations of computer science and the expression of algorithms as pseudocode. A number of algorithms are examined including sequential search, find greatest, selection sort, and binary search. The time efficiency of algorithms and Big-O classification are discussed. Computer hardware concepts are studied including binary numbers, Boolean logic, gates, and circuits such as compare for equality and addition circuits. The construction of a CPU from basic circuits is examined. The C++ programming language is an area of emphasis.

Prerequisites: Graduate standing

Learning Objectives:
Upon successful completion of this course, the student will:
- Understand the algorithmic foundations of Computer Science and be able to express algorithms in pseudocode.
- Understand the design of basic searching and sorting algorithms (linear search, binary search, and selection sort).
- Understand the time and space efficiency of algorithms and big-O notation.
- Understand how binary numbers are represented, basic concepts of Boolean logic and logic gates, and understand the equality and addition circuits.
- Understand the von Neumann model of computer organization.
- Design and develop basic computer programs using high level programming language (sequence, selection, and iteration structures).
- Be able to design and implement programs that use arrays and functions.

Assessment of Objectives:
Assessment of objectives will be conducted through exams, laboratory exercises, and programming assignments.
Instructional Methods and Activities:
The methods and activities for instruction will include:
1. Presentation of new material and concepts in the classroom through the use of lecture, tutorials, and sample programs.
2. Classroom discussion using problem solving techniques.
3. Programming assignments to review and reinforce topics covered in the classroom.
4. Optional one-on-one discussion as needed between the student and instructor outside regularly scheduled class time.

Course Grades:
Lab Assignments: 35%
Midterm Exams (2): 30% (15% each)
Final Exam (comprehensive): 35%

Grade Ranges:
A: 90 - 100%
B: 80 - 89%
C: 70 - 79%
D: 60 - 69%
F: <60%

Please note: NO test grades will be given out via email or over the phone. Don't even bother asking.

Texts:
• Computer Science: An Overview, Eleventh edition, J. Glenn Brookshear
• Starting Out with C++, Seventh edition, by Tony Gaddis

Lab Supplies:
• Flash drive to archive your programs (optional - but strongly recommended)

Other Course-related Material:
All lecture notes, sample programs, programming assignments, and data files are online at www.sci.tamucc.edu/~ptedford.

Class Policies:
Graduate Class Conduct:
This class is a combined section with an undergraduate COSC 1435 section. The general material to be covered in sections is the same, but the graduate students will be expected to understand this material to a greater depth. The exams for the graduate and the undergraduate students will be different, with the graduate exam having more questions specifically targeted to graduate students. The undergraduate students are required to attend a lab where they will be given assistance in completing programming assignments. Graduate students are welcome to attend the undergraduate labs, but are not required to attend. There will also be additional programming assignments for graduate students, additional
requirements in the programming assignments, and the grading criteria for shared labs may be more difficult.

**Computer Security Issues:**

It is your responsibility to read, understand, and follow the Student Security Statement (www.sci.tamucc.edu/~ptedford/MiscPages/StudentSecurityStatement.pdf) handed out the first day of class that discusses computer security issues. Infractions will have a direct affect on your grade in this course.

**My Expectations of You:**

- Come to lecture and lab *on time* every day the class meets.
- Read the chapter to be discussed *before* coming to class.
- Answer the Practice Problems, Checkpoint Questions, and Review Questions in each chapter.
- Ask questions of material you do not understand.
- If I cannot explain the answers to your satisfaction, make an appointment with me to discuss the question.
- Seek help, if needed, from the Program Assistants in CI 346A.
- **DO YOUR OWN WORK!!** Do **NOT** share your work with others.
- Demonstrate integrity, maturity, and ethical behavior.

**Attendance:**

If you want to succeed in this course, your attendance and active participation is crucial. Attendance is taken every day the class meets. If you are not in the room and in your seat *before* the lecture starts, you will NOT be counted as present that day. This may have serious implications on any financial aid you might be receiving. Although grades will not be directly based on attendance, regular attendance and active participation increases your chances of successfully completing this course. You are expected to know all material presented in class. *Turn off all cell phones when you enter the classroom!* *If you are caught texting, surfing the Web, playing games on your cell phone or laptop, or otherwise participating in distracting behavior, YOU WILL BE TOLD TO LEAVE THE CLASSROOM for the remainder of that class period.* Such behavior will **NOT** be tolerated.

**Preparing for Lecture:**

Class topics will follow the order of topics given in the schedule posted on the course’s Web page (http://www.sci.tamucc.edu/~ptedford/CurrentCourses.html). You should read ahead to prepare for each class. Be prepared to study 2 - 3 hours for every hour you spend in class.

**Class Problems:**

You may be asked to work in groups in class to solve problems similar to those that will appear on exams. You are expected to actively participate in these activities. In addition, you may occasionally be asked to write answers to group problems on the board.

**Academic Integrity/Plagiarism:**
University students are expected to conduct themselves in accordance with the highest standards of academic honesty. Academic misconduct for which a student is subject to penalty includes all forms of cheating, such as illicit possession of examinations or examination materials, falsification, forgery, complicity or plagiarism. (Plagiarism is the presentation of the work of another as one’s own work.) This includes NOT sharing code or answers for the individual lab assignments! In this class, academic misconduct or complicity in an act of academic misconduct on an assignment or test will result in sanctions ranging from being given a zero for the assignment(s) involved to being given an F for the course. For additional information, see the “Assignments” section under “Lab Policies”.

Assignments:
Lab work will be assigned on a regular basis. Please refer to the "Lab Policies" section below for specific information and instructions about the lab assignments.

Exams:
You MUST read the text to do well in this class. As much as one third of the material on the tests may be information in the text not discussed in class. Exams may contain multiple choice, true-false, fill-in, short answer, and/or programming questions.

Make-up exams:
Makeup exams will not be given under normal circumstances. If you notify me immediately that serious, unavoidable, documentable (e.g., with a letter from your doctor) circumstances have arisen, I will discuss options for replacing the missing grade. For example, I may allow the grade earned on the comprehensive final to replace the grade for the missed exam. You must take at least one of the midterm exams in this course. In other words, only one midterm exam may be missed due to unforeseen circumstances. Excused absences due to school sponsored activities, religious observations, family rituals, etc. should be discussed with me in advance.

Lab Policies:
Attendance:
Lab attendance is not required. However, if you would like some one-on-one assistance, please feel free to attend the lab. If you want to surf the Web, play video games, or watch videos on YouTube, do it on your own time. Don’t do it in lab.

Assignments:
A list of assigned lab work and the criteria on which the labs will be graded is available from the lab assignments list (http://www.sci.tamucc.edu/~ptedford/CurrentCourses.html) on the first day of class. The assignments will be similar to those assigned to the undergraduates in COSC 1435. However, since this is a graduate course, each of your assignments will have additional components. While there are due dates (see below), there is no actual "assigned date" for each lab. That is, you may start working on the lab assignments immediately. Here are some guidelines you need to follow in order to do well on the labs:

- The assignments are to be completed individually. You may ask each other for general advice, but do NOT share final answers and/or source code unless you have been told
to do so by me. Be sure to protect your programs. **Sharing or giving your work to others is grounds for an immediate "F" for the assignment involved. Submitting another person's work as your own is grounds for an immediate "F" in the course.**

- Plan on spending 5 - 10 hours working on your lab assignments. You cannot learn and understand the material by simply sitting through lectures.
- **Be sure to keep backup copies of ALL your programs!** Storage media have been known to fail. *Not having a backup copy of your work is NOT an acceptable excuse for submitting a late lab.*
- Keep in mind that when we grade your programs, we open the source code in Microsoft Word to comment/critique it. Save yourself some points (and us some frustration) by **not** extending your source code beyond column 73 in the text editor you use to type your code. Anything extending beyond column 73 tends to result in "word wrapping" when opened in Word. Word wrapping makes your code difficult to read and, as such, will result in your losing points unnecessarily.
- Do not resubmit a lab until you have received a grade for the previous submission. For example, if you turn in lab 6, but know it was not complete, wait until you get the results back before resubmitting it again.
- Each assignment is to be submitted using the dropbox as outlined on the Dropbox Instructions page. Lab assignments will **not** be accepted for grading if they are submitted via email, as a "hard copy," or on disk. *If you have problems using the dropbox, ASK FOR HELP!*

**Assignment Due Dates:**
Assignments are to be submitted by 11:59 PM on the due date. Due dates are listed with each assignment. If you want an A on any lab, **it must be submitted on time.** If you are submitting a lab that is less than complete, **you must have it checked by me to make sure it is an acceptable submission.** If you submit a lab late, your grade for that lab will drop according to the following schedule:

<table>
<thead>
<tr>
<th>Late Submission</th>
<th>Possible Grade</th>
</tr>
</thead>
<tbody>
<tr>
<td>1 - 2 days late</td>
<td>Highest possible grade is 89</td>
</tr>
<tr>
<td>3 - 4 days late</td>
<td>Highest possible grade is 79</td>
</tr>
<tr>
<td>5 - 6 days late</td>
<td>Highest possible grade is 69</td>
</tr>
<tr>
<td>7 or more days late</td>
<td>Highest possible grade is 50</td>
</tr>
</tbody>
</table>

No labs will be accepted after Tuesday, May 1, 2012. There will be a lab due the final week of lecture. If an unavoidable, documentable emergency arises, please discuss it with me. However, under normal circumstances, if you have not completed your assignment by the due date, you should submit the **approved** work you have done for partial credit. For the sake of your grade, you should ALWAYS turn in SOMETHING that shows you've attempted to solve the problem. **Be sure to keep ALL graded material.**

**Resubmitting Lab Assignments:**
I allow you to resubmit lab assignments if you are not satisfied with the grade you have earned on a lab. However, there are a couple of "ground rules".
The grade of a lab you want to resubmit must be less than a 90.
The original version of the lab was submitted on time (that means by 11:59 PM on the due date). Labs that were submitted late the first time are NOT eligible to be resubmitted!
The material originally submitted must be one of the following: 1) typed original pseudo-code (not pseudocode that was included in the original lab assignment), 2) the source code for a partially completed program that attempts to solve the problem, or 3) the source code for the program to solve the problem that will not compile and/or execute. In other words, you must have originally submitted something that indicates you made an honest effort to solve the problem. Submitting a ".cpp" file with your name on it and the words "Will resubmit later" does not count as an honest effort. If you are submitting a lab that is less than complete, you must have it checked by me to make sure it is an acceptable submission. If you do not have incomplete worked checked for admissibility prior to submission, you may not be eligible to resubmit that lab. All decisions about what is eligible for resubmission are final.

Do not resubmit a lab until you have received a grade for the previous submission. For example, if you turn in lab 6, but know it's not complete, wait until you get the results back before resubmitting it again.

Do not resubmit a lab unless you have made the changes/corrections necessary to improve your grade! Resubmitting a lab without making any corrections will not earn you any extra points!

Only one lab per directory will be accepted. You may submit more than one lab at a time, but each must be in a separate directory.

Be sure to include ONLY files/documents that have been modified or corrected along with a list of what has been corrected/modified. Don't forget to include a new executable file, too, if needed!

This is VERY important: FOLLOW THE INSTRUCTIONS FOR RESUBMITS IN THE DROPBOX INSTRUCTIONS VERY CAREFULLY!

No resubmits will be accepted after 11:59 PM on Tuesday, May 1, 2012!
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. For assistance and/or guidance in the grade appeal process, students may contact the Office of Student Affairs.

**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.

If you are a returning veteran and are experiencing cognitive and/or physical access issues in the classroom or on campus, please contact the Disability Services office for assistance at (361) 825-5816.

**Course Outline (See online schedule for more details)**

**NOTE:** Until Jan. 31, all lectures will be based on the Brooksheur text. Beginning on Jan. 31 and continuing until about Apr. 12, lectures will refer to the Gaddis textbook. Lectures notes with (G) after the chapter number refer to the Gaddis text.

<table>
<thead>
<tr>
<th>Date</th>
<th>Topic</th>
</tr>
</thead>
<tbody>
<tr>
<td>1/12/12</td>
<td>Syllabus; Chap. 0 – Introduction</td>
</tr>
<tr>
<td>1/17/12</td>
<td>Chap. 0 (cont.); Chap. 5 – Algorithms and Pseudocode</td>
</tr>
<tr>
<td>1/19/12</td>
<td>Chap. 5 (cont.) – Algorithmic Discovery; Iterative Structures; Sequential and Binary Searches</td>
</tr>
<tr>
<td>1/24/12</td>
<td>Chap. 5 (cont.): Insertion and Selection Sorts</td>
</tr>
<tr>
<td>1/26/12</td>
<td>Chap. 5 (cont): Efficiency and Correctness</td>
</tr>
<tr>
<td>1/31/12</td>
<td>Chap. 2 (G): Program Structure and Design, Input Statements, Types, Variables</td>
</tr>
<tr>
<td>2/2/12</td>
<td>Chap. 3 (G): Program Structure, Output Statements, Operator Precedence</td>
</tr>
<tr>
<td>2/7/12</td>
<td>Chap. 4 (G): Conditional Control Structures</td>
</tr>
<tr>
<td>2/9/12</td>
<td>Chap. 4 (G): Conditional Control Structures (cont.), Chap. 5 (G): Repetitive Control Structures</td>
</tr>
<tr>
<td>Date</td>
<td>Event</td>
</tr>
<tr>
<td>-----------</td>
<td>----------------------------------------------------------------------</td>
</tr>
<tr>
<td>2/14/12</td>
<td>Review for Exam #1; Chap. 6 (G): Functions</td>
</tr>
<tr>
<td>2/16/12</td>
<td><strong>Exam #1: Chapter 0 &amp; 5 (B); Chapters 2 &amp; 3 (G)</strong></td>
</tr>
<tr>
<td>2/21/12</td>
<td>Chap. 6 (G): More Functions</td>
</tr>
<tr>
<td>2/23/12</td>
<td>Chap. 6 (G): More Functions</td>
</tr>
<tr>
<td>2/28/12</td>
<td>Chap. 6 (G): More Functions</td>
</tr>
<tr>
<td>3/1/12</td>
<td>Chap. 6, 7 (G): More Functions, Arrays</td>
</tr>
<tr>
<td>3/6/12</td>
<td>Chap. 6, 7 (G): More Functions, Arrays</td>
</tr>
<tr>
<td>3/8/12</td>
<td>Chap. 6, 7 (G): More Functions, Arrays</td>
</tr>
<tr>
<td>3/13 – 3/15</td>
<td><strong>Spring Break!</strong></td>
</tr>
<tr>
<td>3/20/12</td>
<td>Chapter 12 (G): Basic File I/O</td>
</tr>
<tr>
<td>3/22/12</td>
<td>Review for Exam 2; Chapter 12 (G): More Basic File I/O</td>
</tr>
<tr>
<td>3/27/12</td>
<td><strong>Exam #2: Chapters 4, 5, 6, 7, &amp; 12 (G)</strong></td>
</tr>
<tr>
<td>3/29/12</td>
<td>Chap. 6, (G): More Functions</td>
</tr>
<tr>
<td>4/3/12</td>
<td>Chap. 8 (G): Searching</td>
</tr>
<tr>
<td>4/5/12</td>
<td>Chap. 8 (G): Searching, Sorting</td>
</tr>
<tr>
<td>4/10/12</td>
<td>Chap. 8 (G): Sorting</td>
</tr>
<tr>
<td>4/12/12</td>
<td>Chap. 1.5: The Binary System (pages 42 – 45)</td>
</tr>
<tr>
<td>4/17/12</td>
<td>Chap. 1.1: Bits and Their Storage</td>
</tr>
<tr>
<td>4/19/12</td>
<td>Chap. 1.2 – 1.3: Main Memory; Mass Storage</td>
</tr>
<tr>
<td>4/24/12</td>
<td>Chap. 2.1 – 2.2: Computer Architecture; Machine Language</td>
</tr>
<tr>
<td>4/26/12</td>
<td>Chap. 2.3: Program Execution</td>
</tr>
<tr>
<td>5/1/12</td>
<td>Review for final exam</td>
</tr>
<tr>
<td>5/8/12</td>
<td><strong>Final Exam 4:30 – 7:00</strong></td>
</tr>
</tbody>
</table>