# COP 3502 Syllabus
# Computer Programming I - Fall 2010

**Course Prerequisites:** COP 3223 and MAC 1105
**Class Time:** MWF 9:30 – 10:20 AM
**Class Location:** HEC-125
**Course Web Page:** http://www.cs.ucf.edu/courses/cop3502/fall2010/

**Lecturer:** Jonathan Cazalas
**Office:** HEC – 250
**Email:** jcazalas@eecs.ucf.edu
**Office Hours:** MW 10:30 AM – 12:30 PM

**TA Office Hours:** TBD, and will be posted on the course website

**Course Description:** from the UCF catalog description, "Problem solving techniques, order analysis and notation, abstract data types, and recursion." Now, in English, this class is a follow up to the COP 3223 material, in which you learned (ideally) the syntax and use of major constructs of the C language (conditional statements, loops, functions, arrays, pointers, strings, structures, and file I/O). This course now focuses on algorithmic design, analysis of running time, a variety of abstract data types (new data structures), and lastly, but definitely not least important, recursion.

**Course Textbook:** Data Structures, Algorithms & Software Principles in C by Thomas A. Standish. ISBN-13: 978-0201591187.

**Recitations/Lab Sessions:** In addition to the main lecture, this course also includes recitation sections. One of the primary purposes of the recitation is to reinforce the information presented in the main lecture. Additionally, due to the amount of material that must be presented in this course, there will be times where new material is presented in the Labs and not discussed during the main lecture. For this reason, **recitation attendance is mandatory**, resulting in 10% of your final grade. Attendance will be taken, with each day counting as 1%. In order to **earn** your attendance point, you must be present throughout the lab (physically and <u>mentally</u>) and actively paying attention and/or working on the given problems. If you come more than 10 minutes late, leave early, or are playing on a Laptop, phone, game system, etc., you will not get credit for the lab. There will be 13 or 14 lab sessions over the semester. You should go to ALL of them. However, once you have attended 10 labs, you will have received the full 10% (attending more will not earn you extra credit).

**Programming Assignments:** There will be six programming assignments. Each assignment will be introduced in class and posted on WebCourses. All homework assignments are to be turned in through WebCourses by 11:55 PM on the day they are due. The official time a program is submitted will be determined by WebCourses. (Your wristwatch or cell phone time when you hit the submit button is not valid.) Due to possible server issues, it is **strongly suggested** that you attempt to submit programs **at least three hours before the actual time it's due**. All programming for assignments is to be done in C and must compile and run using DevC++. Details on the usage of Dev-C++ will be provided.

***Late Assignment Policy:*** Late homework assignments will be accepted for the first five homework assignments, but will be assessed a late penalty. In particular, assignments will be accepted up to 48 hours after the due date of the assignment. If an assignment is less than 24 hours late, a 10% penalty will be assessed. If an assignment is in between 24 and 48 hours late, a 25% penalty will be assessed. Assignments will not be accepted beyond 48 hours after the due date. ***No late assignments will be accepted on the last programming assignment.***

***Community Service Opportunity:*** In lieu of the last assignment (program 6), you may perform 5 hours (or more) of community service. **If you take this option, then you will automatically get a 100 for program 6.** In order to get this credit, you must complete the community service and turn in the signed requisite form by 12:30 PM on Wednesday, November 10th. I strongly suggest you do this early to guarantee you do not miss the deadline; remember, **this is a FREE 100 for Program 6.** There will be NO exceptions to this deadline. All grades for the community service will be posted under the column P6 two weeks before the last program is due.

**Exams:** There will be two midterm exams and one final exam. Students will be allowed one page (8-1/2" x 11" or smaller) of notes for the two midterms and 2 pages of notes for the final. However, calculators will NOT be allowed for any of the exams. As the material in this course builds on itself, each exam can be considered "cumulative", and material from the beginning of the semester is certainly not off-limits for the 2nd Midterm. And of course, the Final exam is cumulative as well.

**Quizzes:** Quizzes will consist of a small number of basic questions on material that has been covered recently, with the goal of forcing students to keep up with the material. Quizzes will be announced in class and administered via WebCourses. They will normally be due the following day by 11:55 PM with no late exceptions. If you miss class and, therefore, miss the announcement, and happen to have not checked your WebCourses, then you will miss the quiz. There will be no makeup quizzes. It is your responsibility to go to class and to check WebCourses for announcements.

**Tentative Grading Procedures:** The final letter grade will be based upon the six items listed below. Plus/minus grades may be issued, when deemed appropriate. The grading scale will be based on the class average, standard deviation, and overall difficulty of the assignments and exams.

| Item | Percentage |
|---|---|
| Lab Attendance | 10 |
| Quizzes | 10 |
| Homework Assignments | 30 |
| Exam #1 | 15 |
| Exam #2 | 15 |
| Final Exam | 20 |

Also, in order to pass the class you must earn at least a 40% on the final exam. (Thus, if you have a 75% in the course but earn a 30% on the final, you still get a C- in the course even though your percentage may qualify for a B.)

**Important Dates:**

**Labor Day:** Monday, September 6th (no class)
**First Midterm:** Friday, October 1st
**Withdrawal Deadline:** Friday, October 15th by 11:59 PM
**Second Midterm:** Friday, November 5th
**Thanksgiving:** November 25 – 27 (no class on Friday the 26th)
**Last day of class:** Monday, December 6th (and we will hold a lab session)
**Final Exam:** Friday, December 10th, bright and early, at 8:00 AM*

\* The official, UCF start time is 7:00 AM.  However, the final exam is not three times longer than your regular midterms, for which you have 50 minutes.  As such, the Final exam will start at 8:00 AM, and you will have one hour and fifty minutes.


**Other Important Course Policies:**

**1)**  The TAs are your first, second, and even third points of contact regarding the programming assignments.  If you have any questions at all regarding the assignment, solving the program, how to code it, syntax errors, you name it, please contact the TAs.  There are plenty of TA office hours throughout the week, and I strongly suggest you take advantage of them.  You can also email them with your questions, but understand that they may not respond immediately.  If you want help via email, start your assignment early.  Finally, the TAs will be grading the assignments.  Therefore, any and all questions you have regarding your grade should be directed to them.  If you feel your grade was unfair and you were not satisfied after contacting the TA, please come to my office hours to discuss.

**2)**  Cheating will not be tolerated. **If a student is caught cheating, then the grade on that assignment for all students knowingly involved (the person providing answers as well as the one taking the answers) will be a -25%.** (**Note, this is less than 0%.**)  Since discussion of concepts with other students is often helpful, cheating must be more clearly defined. In particular, the following items are cheating: copying a segment of code of three lines or more from another student from a printout or by looking at their computer screen, taking a copy of another student's work and then editing that copy, and sitting side by side while writing code for assignments and working together on segments of code. In all of these situations, **BOTH people responsible**, the one from whom the three lines of code are taken as well as the person who takes those lines of code are engaging in academic misconduct. For example, if someone makes an electronic copy of their code accessible to ANYONE in the class (except for themselves) before 48 hours after an assignment is due, they are automatically culpable of academic misconduct. It does not matter if the recipient of the code doesn't use it, uses it a little, or copies it directly. Furthermore, based on the severity of the case, the entire course grade for the student may be lowered an entire letter grade. ***If you get stuck on an assignment, please ask either a TA for help instead of getting help from another student.*** Part of the learning process in programming involves debugging on your own. In our experience, when a student helps another student with an assignment, they rarely allow the student getting help to "figure out" problems on their own. Ultimately, this results in a lack of debugging experience for the student receiving help. The goal of the TAs and instructors is to provide the facilitation necessary for students to debug and fix their own programs rather than simply solving their problems. **But, you are encouraged to work together on any non-graded programs to enhance and expedite the learning process.**

**3)** In order to take a make-up exam, you must request one from the instructor. The instructor will grant requests using his own judgment by applying the following general rule: "Make-up exams will only be given if the reason for missing the exam was out of the student's control." For example, being hospitalized unexpectedly is out of a student's control, but oversleeping or going to happy hour is not out of a student's control. ***If possible, it is recommended that the instructor be contacted before the exam.***

**4)** The penalty for assignments late by less than 48 hours will be waived only for circumstances described in #2 above at the instructor's discretion. Similarly, assignments will be accepted after 48 hours past the due date under these same types of circumstances. The student ***must*** contact the instructor within two days of the due date of the assignment in order to get credit for the assignment. ***TAs are NOT allowed to give extensions for assignments under any circumstances, only the instructor can.***

**5)** Both the course web page and WebCourses will be crucial elements of the course. ***It is your responsibility to check both of these before every class meeting for any updates that may be posted.*** Additionally, some clarifications may only be given in class and won't be posted online at all, so make sure you keep up with announcements in class.

## Tentative Schedule for Lectures/Assignments

| Week | Monday Class | Wednesday Class | Friday Class | |
|---|---|---|---|---|
| Aug 23-27 | Go over Syllabus + Intro PPT Slides + C-Review: Read on your own. | Programming Team Presentation + Linear vs Binary Search | Sorted List Matching Problem + Review of Pointers | Assignment 1: C-Practice (Due 9/3) |
| Aug 30 – Sep 3 | Dynamic memory allocation for arrays | Linked Lists Intro: Traversing a list, counting elements in a list, printing a list, and more | Linked List operations: Insertion into a list | Assignment 2: Linked List (Due 9/15) |
| Sep 6 - 10 | **Labor Day (no class)** | Linked List operations: Deleting nodes from a list | Linked Lists Gone Wild: Circle and Doubly-Linked Lists | |
| Sep 13-17 | Recursion 1: Intro to Recursion, examples include count down, factorial, and Fibonacci | Recursion 2: General structure, sum numbers, power, reversing a string, multiply, & Towers of Hanoi | Recursion 3: Permutations + Info on Assignment 3 | Assignment 3: Recursion (Due 9/29) |

| Sep 20-24 | Recursion 4: Recursive Binary Search & Fast Exponentiation | Base Conversion Methods | Big-O notation, algorithm analysis, time complexity problems; use of summations | |
|---|---|---|---|---|
| Sep 27 – Oct 1 | More Algorithm Analysis | Even more Algorithm Analysis + Exam #1 Review | **Exam #1** | |
| Oct 4-8 | Time complexity using recurrence relations | Stacks – applications, evaluation of postfix expressions | Use of stacks for infix to postfix; array and linked list implementation of a stack | |
| Oct 11-15 | Go Over Exam #1 | Queues: array and linked list implementation of a queue | Binary Trees, relation of height to number of nodes, tree traversals | Assignment 4: Stacks/Queues (Due 10/27) |
| Oct 18-22 | Binary search tree, searching in a BST, insertion | Deletion in BST | Various other binary tree functions | Assignment 5: BST (Due 11/17) |
| Oct 25-29 | Sorting- selection sort, insertion sort, bubble sort | Quick Sort | Merge sort | |
| Nov 1-5 | More Sorting | Exam #2 Review | **Exam #2** | |
| Nov 8-12 | Heaps & Priority Queues | More Heaps & Heapsort | Hash tables | |
| Nov 15-19 | Go over Exam #2 | free | AVL Trees, insert | Assignment 6: Heaps (Due 12/1) |
| Nov 22-26 | AVL Tree delete | free | **Thanksgiving (no class)** | |
| Nov 29 – Dec 3 | Backtracking | Introduction to Graphs | Algorithm Analysis Revisited | |
| **Dec 6 -10** | FINAL EXAM REVIEW | | **FINAL EXAM ( 8:00 – 9:50 AM)** | |