

CPS 150: Algorithms and Programming I/Fall 2004

The goal of CPS 150 is to introduce the fundamental concepts of problem solving and structured computer programming, including analysis of numeric and non-numeric problems, algorithm design and implementation using a compiled language, debugging, and testing. The approach is to expose students to these concepts through two and a half hours of lecture and two and a half hours of lab per week (4 sem. hrs). The programming language used in the course is C++ in the MS Visual Studio.NET environment (2003 release; available from <http://software.udayton.edu>).

Meeting Times	MW, 5:50–7:05p, Miriam 215 (lecture) TTh, 5:50–7:05p, Miriam 21A (lab)
Instructor	Dr. Saverio Perugini, 229–4079, Anderson 145 saverio@notes.udayton.edu
Office Hours	Mondays, Wednesdays 2–4p, or walk in any time; and by appointment.
Graduate Assistant	Karthik Ganesan Pillai, 229–2199, Miriam 21 ganesakz@notes.udayton.edu
Office Hours	Tuesdays, Thursdays 10:30a–noon
Course Webpage	http://academic.udayton.edu/SaverioPerugini/courses/cps150/

If you need adaptations or accommodations because of a disability (learning disability, attention deficit disorder, psychological, or physical), if you have emergency medical information to share with the instructor, or if you need special arrangements in case the building must be evacuated, please meet with the instructor ASAP.

Pre-requisites: There are no formal pre-requisites. However, students are expected to have strong analytical skills or an interest in solving mathematical/logic puzzles.

Book: ‘Problem Solving with C++: The Object of Programming’ [PSwC++] by Walter Savitch, Addison-Wesley, *Fourth Edition*. Make sure you get the fourth edition of PSwC++, not the fifth! This textbook is required and we will give regular reading assignments from it. The author maintains a website at <http://www.aw.com/savitch>, containing lots of helpful material. In addition, we may use supplementary material distributed in class for various topics as they become appropriate. More details on these will be provided as the course progresses.

Evaluation: There will be approximately 10–15 homeworks, which will primarily entail programming from a problem specification, but may also involve a mix of algorithm design, theoretical questions, and small programming puzzles. All assignments are due in class, before class starts, and will be either submitted as a printout (please no handwritten submissions) or collected electronically from your CPS account. *No late submissions will be accepted*—it is just not fair to your classmates who submitted the assignment on-time. In addition, an assignment submitted after class begins on the due date is considered late and thus will not be accepted. If you cannot make it to class that day, arrange to submit the assignment early. Also, remember that computer equipment (hard disks, printers) and network infrastructure fail and such failure is not the basis for an excuse. Maintain multiple backups of your work, budget your time appropriately, and stay abreast of deadlines.

There will be 3 exams and a comprehensive final. All exams are in class, and closed book,

closed notes. Detailed breakdown: homeworks (25%), labs (15%), quizzes (15%), exams (30%), final (15%). The instructor typically grades all exams individually while the GA grades the homeworks and quizzes. The instructor and lab assistants will typically collaboratively grade the labs in the lab. If you have an assignment which you feel has been graded incorrectly, contact the instructor within one week of the date that the assignment was made available for pickup (not from the time you actually claimed it or learned your score), and we can discuss a re-grading if appropriate. Re-grades will not be considered after one week. Save all submitted and graded work (marked up hard copy) until the end of the term. Final letter grades of A, A-, B+, B, B-, C+, C, C-, and D will start approximately at 93, 90, 87, 83, 80, 77, 73, 70, and 60, respectively. Please note that there is typically no correlation between the number of years of experience with programming in C++ (or substitute your favorite programming language here) and your grade in CPS 150. It is not unusual for students with no experience with computers or programming to perform exceptionally well. Likewise, it also is not uncommon for students with several years of programming experience from high school or industry to perform poorly.

Attendance: Required at all examinations. Notify me at least one day in advance for any unavoidable exceptions. Make-up exams may be given in cases of serious (documented) illness or other emergencies at the instructor's discretion. Please note that any make-up exam may be different than the original. Any test which was missed and unexcused will count as zero points toward the final grade. No make-up quizzes or labs will be given. However, the lowest two quiz scores and the lowest two lab scores will be dropped. If you have an unexcused absence from lecture, it is your responsibility to find out what material (discussions, assignments, handouts) you missed by consulting the course webpage and asking your classmates. First-year students (<30 semester hours) are required to adhere to the attendance policy stated in the student handbook (available at <http://www.udayton.edu/~studev/studenthandbook/>):

“...the allowable number of absences in the first term ... will be equal to twice the meeting times a week [in our case, 8] ... A student exceeding this number will be referred to the student's dean for possible counseling and appropriate action” (ref. 2004–05 UD Student Handbook, p. 37).

Labs: Include a stepwise assignment which often requires the lab assistant or instructor to periodically check on student progress. The initial portion of lab may include a succinct (10–15 minute) lecture and may cover material not discussed in the course lecture. Labs must be completed in the lab and will be due at the end of the lab, unless otherwise indicated by the instructor or lab assistants. Students are encouraged to use the entire lab period to complete the lab. Students who complete the lab early may leave or use the remaining time for CPS 150-related work only.

Keeping in Touch: Expect to receive important announcements via e-mail. All e-mail will be sent to your university Lotus Notes account only. If you have set forwarding from there (e.g., to Hotmail), it is your responsibility to ensure that it works. Make sure to check your e-mail at least once a day, and more frequently if there is something special going on. There also is a listserv (mailing list) for distributing announcements to the entire class which will be used.

Workload: CPS 150 is not easy! It is a challenging course and moves at a very fast pace. It will appear deceptively simple at first but, unless you start early, you will be unable to complete the assignments. The topics covered in this course progressively build on each other and it is easier to keep up than catch up. The programming assignments will involve a large amount of thought

and design, so plan your schedule accordingly. Spending a minimum of 10–12 hours of independent work (outside of class) each week studying the concepts and exercising your problem-solving and programming skills is required to do well in this course. I strongly encourage you to see me to discuss any problems you may have before you are tested on the material or before deciding to drop the course. Having said all of this, CPS 150 will be exciting, extremely fun, and can be your claim to fame. Mastery of the concepts studied will provide you with a strong foundation in computing and pave the way for success during the rest of your computer science curriculum and beyond.

Electronic Accounts and Lab Facilities: You will be assigned an account on the CPS undergraduate lab network, which is to be used only for this course. All accounts (e.g., e-mail) given to you by the university, including your CPS account, are not yours, but rather property of the university, and therefore, must be used responsibly. Please refer to the ‘Policy on Fair, Responsible, and Acceptable Use of Electronic Resources at UD’ at <http://www.udayton.edu/~udit/ethics/policy.htm> for details. I will have read-only access to all files stored in your account. You can access your account from the Miriam Hall 21A, Anderson 131, or Anderson 135 labs. You also can access your account remotely using FTP (more on this later). When not reserved for another class session or meeting, you may work on your CPS 150 assignments from these labs outside of regularly scheduled class time. Please respect the rights of the faculty and other students by not attempting to work in a lab when another class is in progress. In addition, the printers found in the CPS labs are reserved for use related to CPS course work only. Information about the CPS labs, including hours of operation, is available at <http://www.udayton.edu/~cps/facilities.html>.

Classroom Etiquette: Students are expected to conduct themselves in a professional manner. Talking to other students during lecture is disruptive to other students around you and forces the instructor to speak louder. All classes will begin promptly at 5:50p. We will expect you to be in your seat and ready to begin work then. If you are forced to arrive late or leave early, please sit near the door so as not to disrupt others. Sleeping, browsing the web, reading/writing e-mail, or using instant messengers or similar things is not permitted in class. Such behavior is insulting to the instructor and graduate/lab assistants, discourteous to students around you, and shows a lack of moral fortitude. In addition, we do not permit the use of laptop computers in lecture, with the exception of a documented disability which requires the use of a computer for note-taking; and please turn off all mobile phones, pagers, and similar devices prior to entering class. Lastly, wearing hats will not be allowed in class.

Ethics: One of the main goals of this course is to enable the student to master the fundamentals of problem-solving and computer programming. To achieve this goal, students must work on their assignments individually. It is acceptable to discuss the goal of an assignment, but not how to achieve that goal, with classmates. Questions related to solutions must be directed only to the instructor and graduate/lab assistants. Do not make any assumptions as to who can provide help on an assignment. Working with others to derive a common solution to a homework assignment or copying, reading, or sharing source code (programs) or homework solutions in whole or part from someone else, or their disk or account, even if unprotected, constitutes cheating and is prohibited. Homework assignments may be subject to scrutiny by a system (or systems) which measure(s) software similarity (cheating detection). Also, never make assumptions regarding which references, such as webpages, books, and articles may be used to help lead you to a solution. When in doubt, consult the instructor. The policy on academic dishonesty set forth in the student handbook (pp. 34–36)

will be strictly enforced in this course. Evidence indicating a violation or violations of the university policy or the policy stated above will be dealt with severely as per the student handbook and may result in a zero for the assignment in question or an F in the course.

No student should ever feel that they must resort to academic dishonesty. Please come talk to us if you are struggling with the course or have a problem completing an assignment on time. There is plenty of help available to you from acceptable sources (instructor, graduate/lab assistants) and we are all more than happy to help you. It is much easier to explain a poor grade to parents, a potential employer, or graduate admissions committee than a violation of the university policy on academic dishonesty. Remember, no grade is worth your integrity; and it is never wise to borrow from integrity to pay expediency. Honesty in your academic work will develop into professional integrity. The faculty and students of the University of Dayton will not tolerate any form of academic dishonesty.

Tentative Schedule:

Week	Topic
1	Introduction to computer systems and programming
2	Variables and data types; input and output; arithmetic and logical operators
3	If statements; nested if statements
4	Loops
5	Loops (contd); pre-defined functions
6	Programmer-defined functions; procedural abstraction
7	Void functions; reference parameters
8	File Input/Output
9	Structures; classes and ADTs
10	Operator overloading; friend functions
11	Separate compilation; namespaces; arrays
12	Arrays in functions
13	Array sorting and searching; arrays and classes
14	Arrays and classes (contd)
15	Two-dimensional arrays
16	Character strings