Enrollment in this section is an implicit acknowledgment of the requirements, regulations, and policies set forth in this document.

Objectives: The goal of CPS 343/543 (3 sem. hrs) is to study historical and current issues in the design, implementation, and application of programming languages. The approach is to comparatively assess language design and implementation options and understand their influence on programming practice and methodology. Examples will be provided from a host of computer languages such as FORTRAN, Pascal, COBOL, C, C++, Java, Scheme, FORTH, PROLOG, Modula, Ada, and Smalltalk.

Students can expect a survey of programming paradigms, including representative languages, a treatment of language definition and description methods, an overview of features across all languages, and discussion of implementation strategies.

Meeting Times  M W, 3:00–4:15p, Miriam 213
Instructor      Dr. Saverio Perugini, 229–4079, Anderson 145,
saverio.perugini@notes.udayton.edu
Office Hours    MW, 1:00–3:00p, or walk in any time; and by appointment.
Graduate Assistant Wei Fan (Roger), 229–2199, Miriam 21,
hsuweifz@notes.udayton.edu
Office Hours    T Th, 1:30–3:30p, and by appointment.
Course Webpage  http://academic.udayton.edu/SaverioPerugini/courses/cps543/

If you need adaptations or accommodations because of a disability (e.g., 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 or otherwise, please meet with the instructor ASAP.

Pre-requisites: CPS 350: Data Structures and Algorithms. No exceptions. In addition, students are expected to have strong algorithm design and programming skills and the ability to code up decent sized projects from specifications.

Required Textbook: *Concepts of Programming Languages* by Robert W. Sebesta. Addison-Wesley, Sixth Edition, 2004. ISBN: 0-321-19362-8. Make sure you get the sixth edition of the book, not the fifth. We will regularly assign reading and problems from the book. Historically, students who regularly read the book and independently solve problems from it (beyond those assigned in class) are more successful in the course than those who do not. The textbook is broad in coverage and will be a useful addition to your personal library. In addition, I will recommend books, resources, and manuals for a variety of languages, such as Pascal, Scheme, ML, PROLOG, and Smalltalk as they became appropriate. 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 homeworks, which will involve a mix of theoretical questions and small programming assignments. In addition, there will be four large programming assignments which will each involve coding up a project in a different language from a specification.
Each project as well as homework questions which require source code will be graded according to not only correctness, but also design, efficiency, style, and documentation. All assignments are due in class, before class starts, and will be either submitted as a printout (please no handwritten submissions) and/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 on a due date, arrange to submit the assignment early. However, since completing graded work is critical to your comprehension and mastery of the material, you are encouraged to complete any assignments which you do not submit on time. Also, remember that computer equipment (hard disks, printers) and networks fail and such failure is not the basis for an excuse. Maintain multiple backups of your work, budget your time appropriately, and stay apprised of deadlines.

There will be a midterm exam and a comprehensive final exam (2:00–3:50p, May 4, Miriam 213; no exemptions). All exams are in class, and closed book, closed notes. Detailed breakdown: homeworks (15%), programming projects (40%), quizzes (5%), midterm exam (15%), final (25%). The instructor typically designs all homeworks, projects, quizzes, and exams. In addition, the instructor typically grades all exams while the GA grades the homeworks, projects, and quizzes. 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 (in electronic form and marked-up hard copy) until the end of the term, especially since some of the assignments may build on previous assignments. 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.

**Addendum for Graduate Students:** Each homework assignment or quiz may contain a few extra questions for graduate students. Exams will contain a few extra questions for graduate students. In addition, graduate students will be required to code a fifth project or write a term paper of publishable quality (more on this later).

**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. Note that any make-up exam may be different than the original. Any exam which was missed and unexcused will count as zero points toward the final grade. No make-up quizzes will be given. However, we will drop the two lowest quiz scores. We encourage you to attend and actively participate in all class sessions. Students who regularly attend and contribute to class are historically more more successful in the course than those who do not. 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.

**Workload:** CPS 343/543 is not easy! While the course provides a nice mix of theory and practice, it is a programming-intensive course. The course is challenging 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. Graded work will cover concepts discussed in class. However, often you will not have seen (in class) the specific problems you will be asked to solve on graded work. In other words, you will be required to think independently for graded work and not just regurgitate what you have seen in class. Spending a minimum of 10 hours of independent work (outside of class)
each week studying and experimenting with the concepts 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 343/543 will be
exciting, extremely fun, and can be your claim to fame. It also happens to be my favorite course
to teach! Mastery of the core concepts underlying programming languages will make you a better
programmer whether you are building compilers or programming routers.

Projects: Each project will cover a language paradigm (imperative, functional, logic, and object-
oriented) and the choice of programming language for each has been carefully selected to make
the core concepts salient. The projects will involve a fair amount of thought and design, so plan
your schedule accordingly. You are also expected to be a good coder, and to choose appropriate
data structures and algorithm design strategies. Some of the projects can get unwieldy and sound
choice of data structures can be useful in completing them on time. Lastly, your projects also are
expected to adhere to sound programming principles and style guidelines (distributed later).

Office Hours: You are encouraged to come to office hours for extra help. Note however that
questions (via e-mail or office hours) regarding an assignment will not be entertained on the as-
signment’s due date. The motivation for this policy is not to reduce your access to assistance. On
the contrary, this policy is intended to encourage you to start on assignments early so that we can
resolve your doubts and answer your questions before it is too late.

Keeping in Touch: Expect to receive important announcements and clarifications via e-mail. All
e-mail will be sent to your university Lotus Notes account only. Make sure to check your e-mail at
least once a day, and more frequently if there is something special going on. It is our responsibility
to send you important information, but it is your responsibility to ensure that you receive and read
it. Therefore, if you have set mail forwarding from your Notes account, e.g., to Hotmail, it is your
responsibility to ensure that it works. There also is a mailing list for distributing announcements
to the entire class which will be used. Since it is created automatically by a central university
system, its subscribers will always match the class roster. If you do not receive a test mail from
the instructor by the end of the first week of classes, ensure that your e-mail address is properly
recorded in the university system or contact the instructor.

Course Webpage: The course webpage will serve as the central location for all course-related
items. Make it your browser’s homepage! The webpage will contain a detailed course calendar,
including deadlines and other notable dates, lecture notes, reading assignments, and links to elec-
tronic versions of important course documents, such as this handout. Homework assignments and
project specifications also will typically be distributed via the webpage.

Electronic Accounts, Lab Facilities, and Programming: You will be assigned an account on
the CPS 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 ‘UD Policy on Fair, Responsible, and
Acceptable Use of Electronic Resources’ at http://www.udayton.edu/~udit/ethics/policy.htm for
policy.html. The GA and 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 343/543 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 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.

Familiarity with high-level programming in Windows and UNIX operating systems is expected.

Classroom Etiquette: Students are expected to conduct themselves in a professional manner. Class will begin promptly at 3:00p. 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. We do not permit the use of laptop computers in class, with the exception of a documented disability which requires the use of a computer for note-taking; and please switch all cell phones, pagers, and similar devices to a silent mode during class. Lastly, wearing hats will not be permitted during exams and quizzes. Talking to other students during lecture is disruptive to other students around you and forces the instructor to speak louder. In addition, sleeping, browsing the web, reading/writing e-mail, or using instant messengers or similar things is not permitted in class. Such behavior is discourteous to your fellow students around you, insulting to the instructor and graduate/lab assistants, and shows a lack of moral fortitude. The graduate/lab assistants and I reserve the right to evict students from the classroom for sleeping, disruptive talking, or any egregious breach of classroom decorum. Those evicted will receive a zero on any graded work that day.

Ethics: One of the main goals of this course is to enable the student to master the fundamentals of programming languages. To achieve this goal, students must work on their assignments individually, unless noted otherwise. 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; always verify with the instructor first. Working with others to derive a common solution to an assignment or copying, reading, or sharing source code (programs) or solutions in whole or part from someone else, or their disk or account, even if unprotected, constitutes cheating and is prohibited. Homework assignments and projects may be subject to scrutiny by a system (or systems) which measure(s) software similarity (cheating detection). Also, all submitted solutions and code must be your own with the following permissible exceptions: code discussed or distributed in class or on our course webpage and text or code found in the course textbook. In these cases, you must include detailed comments that indicate which parts of the assignment you received help on and what your sources were. 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.