CS 1030: Foundations of Computer Science

Administrative Details and Course Objectives
Fall 2018

Important Information

Class Website: Canvas (available through CIS)
Lectures: Mondays and Wednesdays 2-2:50 in L104 WEB
Labs (attendance required): Tuesdays and Thursdays 8:35-9:25a, 9:40-10:30a, 10:45-11:35a, 11:50a-12:40p, 12:55-1:45p, 4:10-5p, or 5:15-6:05p in L124 WEB
Instructor: D. Erin Parker, 3144 MEB, parker@cs.utah.edu
Textbook: none (reading assigned from free, online sources)

Test Dates
Test 1: Wednesday, September 26 (during class)
Test 2: Monday, November 5 (during class)
Final exam: Tuesday, December 11 1-3p

Final course grade
Weekly assignments 45%,
Test 1, Test 2, and Final Exam 40%,
Laboratory exercises and online activities 15%

Failing test/exam average → failing course grade

Getting help
The class website has details on how to see TAs and the instructor outside of class, as well as how to post questions to the class forum and contact the course staff.

Course Information

CS 1030: Foundations of Computer Science is a course for students who are interested in pursuing a computer science degree but have no background in computing and students who wish to know more about the field of computer science. This course provides a gentle introduction to the fundamental concepts of computer science. In particular, students learn problem-solving skills and apply them by writing programs in a visual and fun programming environment that is friendly to beginners. Students also study, simulate, and visualize the inner workings of a simple computer.

CS 1030 gives students a sampling of what makes computer science an innovative and exciting field, as well as, preparing them for CS 1410: Introduction to Object Oriented Programming and the rest of the computer science degree. Students may not take CS 1030 and CS 1410 concurrently.

Co-requisite: MATH 1060, 1080, 1090, or Calculus courses
Course Schedule

The CS 1030 material will be presented according to the following schedule (which is subject to change).

WEEK 1 — Getting Started
   Course overview, administrative details, problem solving introduction

WEEK 2 — Introduction to Algorithms
   How to follow and compose algorithms, problem-solving strategies

WEEK 3 — Introduction to Programming
   How to compose a computer program, solving problems (from algorithm to testing)

WEEK 4 — Repetition and Selection
   Repetition via loops, selection via if-else, testing for correctness

WEEK 5 — Lists and Searching Algorithms
   Sequential and binary searching algorithms

WEEK 6 — Procedures in Programming
   Procedures as a way of abstracting and reusing code

WEEK 7 — Representing Data: Integers
   Abstract representations, binary numbers, two’s complement for signed integers

WEEK 8 — Representing Data: Images, Sound, Floating-Point Numbers, Text
   Representing color and images, analog vs. digital, representing floats and text

WEEK 9 — Manipulating Data: Boolean Logic
   Manipulating bits: why and how, logic gates, boolean algebra, truth tables

WEEK 10 — Manipulating Data: Circuits
   Everyday circuits, circuits for addition and selection

WEEK 11 — A Simple Computer
   Von Neumann model, data and control paths, machine language

WEEK 12 — Levels of Abstraction in Programming
   Assembly language and assemblers, high-level languages and compilers

WEEK 13 — Limitations of Computing
   What computers can and cannot do, and why

WEEK 14 — Transitioning to CS 1410 and Beyond
   Brief introduction to Java, how to major in Computer Science and Computer Engineering

WEEK 15 — Course Wrap-Up and Review
   CS 1030 wrap-up, review for final exam
Course Materials

Textbook and online notes. Reading will be assigned from online notes and articles linked from the class website. There is no required textbook.

Website. The class website is a Canvas course available through CIS. It is always under development, with updates to the class schedule, lecture notes, laboratory exercises, assignment specifications, and much more. It is critical that students become familiar with the class website right away, and plan to visit it several times a week, at a minimum.

Lecture notes. The instructor will often make use of slides and other documents during lecture. These documents will be posted on the class website following the lecture; however, such posted documents may not represent completely the material covered in class. Students who must miss class are strongly encouraged to check with a classmate.

Laboratory practice. Twice weekly lab sections will meet at the times listed in L124 WEB. Exercises to be completed during lab will cover topics recently discussed in lectures, allowing students to practice applying these concepts. New students should create a College of Engineering lab account at https://webhandin.eng.utah.edu/cade.

Announcements. Important course announcements will be posted to Canvas and done so sparingly. Make sure to set up Canvas notifications appropriately to receive the announcements in a timely manner. Ideally, each student should receive an email notification as soon as an announcement is posted.

Personal computers. Students may use their own computers for completing homework assignments; however, broken tools or computers, or network connectivity issues are not sufficient basis for a homework deadline extension. Plan ahead and use the lab computers if your own computer is not working.

Student Evaluation

Weekly assignments. The instructions for each assignment and its due date will be posted on the class website at least one week before it must be submitted. Assignments are submitted online via the submission tool located beneath the instructions for each assignment. It is the student’s responsibility to ensure the successful and timely submission of each programming assignment — start early and follow the instructions carefully. Corrupted or missing files will not be grounds for extensions. Double-check your submissions, and save a digital copy of all of your work in your CADE account. No assignments will be accepted late, except in the case of a documented medical emergency.

Laboratory exercises. Every Tuesday and Thursday, students complete a lab exercise reviewing the material covered recently in lecture or preparing for an upcoming assignment. Often, a worksheet accompanies the lab exercise and is due at the end of the designated lab period. No lab exercise / worksheet may be made up for credit, except in the case of a documented medical emergency.

Tests and final exam. Tests will be given during class meetings on September 26 and November 5. The final exam is cumulative and will take place December 11 1-3p. No test or the final exam may be taken at a different time for any reason other than a documented medical emergency.
Final course grade. If the average score for Test 1, Test 2, and Final Exam is 65% or lower, the final course grade will be no higher than a D+. Otherwise, the final course grade will be based on assignments (45%), Test 1, Test 2, and Final Exam (40%), and laboratory exercises and online activities (15%).

Regrades. Students who wish to appeal a score on an assignment, a lab, an online activity, or a test must do so within one week of receiving the score and use the Regrade Request Form posted on the class website.

Drop scores. Students may miss a deadline or a lab session for a reason that is not granted an exception. Therefore, to allow for such an occurrence, the lowest score earned on an assignment, and the lowest two scores on lab exercises are dropped from the record of each student at the end of the semester. Students should plan to use the “drop scores” judiciously — there is only one for an assignment and two for labs. No test or final exam scores are dropped.

Letter grades. The following table is used to associate numerical scores with the corresponding letter grade. Note the lack of rounding.

<table>
<thead>
<tr>
<th>Numerical Score</th>
<th>Letter Grade</th>
</tr>
</thead>
<tbody>
<tr>
<td>93 ≤ X ≤ 100</td>
<td>A</td>
</tr>
<tr>
<td>90 ≤ X &lt; 93</td>
<td>A-</td>
</tr>
<tr>
<td>87 ≤ X &lt; 90</td>
<td>B+</td>
</tr>
<tr>
<td>83 ≤ X &lt; 87</td>
<td>B</td>
</tr>
<tr>
<td>80 ≤ X &lt; 83</td>
<td>B-</td>
</tr>
<tr>
<td>77 ≤ X &lt; 80</td>
<td>C+</td>
</tr>
<tr>
<td>73 ≤ X &lt; 77</td>
<td>C</td>
</tr>
<tr>
<td>70 ≤ X &lt; 73</td>
<td>C-</td>
</tr>
<tr>
<td>67 ≤ X &lt; 70</td>
<td>D+</td>
</tr>
<tr>
<td>63 ≤ X &lt; 67</td>
<td>D</td>
</tr>
<tr>
<td>60 ≤ X &lt; 63</td>
<td>D-</td>
</tr>
<tr>
<td>X &lt; 60</td>
<td>E</td>
</tr>
</tbody>
</table>

Getting Help

To get help understanding course material, students may see the Teaching Assistants during TA Help Hours, see the instructor during Office Hours, post a question to the Discussion forum, or send email to the course staff at teach-cs1030@lists.utah.edu. See the How to Get Help in CS 1030 link on the class website for details.

Course Guidelines

CS 1030 laptop policy. Students are expected to engage with the instructor and classmates during class meetings. Students are permitted to use a laptop or mobile device to take notes. Use of a laptop or mobile device for any other purpose is not permitted, and students who do so will be asked to leave the classroom.

UofU Student code. All students are expected to maintain professional behavior, according to the University of Utah Student Code at www.regulations.utah.edu/academics/guides/students/studentRights.html. Students should read the Code carefully and know that they are responsible for the content. According to Faculty Rules and Regulations, it is the faculty responsibility to enforce responsible classroom behaviors, beginning with verbal warnings and progressing to dismissal from class and a failing grade. Students have the right to appeal such action to the Student Behavior Committee.

Other policies and guidelines. CS 1030 students are also bound by the following policies and guidelines:

- CS 1030 Academic Misconduct Policy (www.cs.utah.edu/~parker/1030_Fa18_amp.pdf)
The following are expected outcomes for a student completing CS 1030.

- The student will be able to solve (simplified versions of) scientific problems using skills such as following and composing algorithms, as well as, using variables, assignment, selection, repetition, and lists. Furthermore, the student will be able to implement solutions in a graphical programming environment that is friendly to beginners (e.g., Scratch).

- The student will be able to relate the concepts of data representation, logic gates and Boolean algebra, and digital circuits to the fundamental workings of a Simple Computer (made up of RAM, registers, an ALU, and a control path). Furthermore, the student will be able to “program” the Simple Computer (simulated using the Logisim tool) to solve simple problems in a variety of ways (including direct manipulation of the control path, machine-language instructions, assembly-language instructions, and high-level programming-language instructions).

- The student will be able to explain abstraction as a crucial mental technique of devising simple models for complicated things by selectively ignoring details. The student will be able to describe abstraction at work in the Simple Computer and in problem-solving techniques.

- The student will be able to discuss what computer science is, what computer scientists do, and what makes computer science a fun and challenging field.

The expected outcomes will be achieved by covering the CS 1030 material according to the schedule on page 2 of this document.