Syllabus

Spring Semester, 2022

General Information

Course Description

Introduction to Programming II: Learn how to use the C++ language effectively, to write readable, and efficient programs. Learn how to use the abstractions in the language, as well as writing your own abstractions. Learn about the static type system, and the C++ compilation model. Learn how to use simple data structures, and write some of your own.

Objectives

The focus of the course will be twofold:

- Learn the C++ language, with a focus on using the available functionality in the STL.
- Learn how these tools work under the hood, so you can design your own data structures and algorithms.

Pandemic Specific Information

In-Person Labs

Lab sections 1-8 are meeting in-person in the STEM building once MSU allows in-person classes to resume. According to University policy (as well as health expert recommendations), proper mask wearing is required while in the building. Students who fail to properly wear a mask will be given one warning before being removed from the lab and disciplinary actions being filed.

Alternate Lab Options

All students (regardless of section) are allowed to attend a synchronous Zoom lab section (see Piazza for details) if they would prefer without prior permission. All students can also
earn lab credit by earning all the points on the Coding Rooms assignment associated with the lab by Friday, 10pm.

**Attending In-Person**

If there is any reason you should not be attending the in-person lab (symptoms, exposure, quarantining, anxiety, etc.), either attend the Zoom synchronous lab, or complete the lab independently. There is no need to inform course staff of which option you are choosing each week.

**Course Materials**

- $15 Payment to Coding Rooms (see here)
- Recommended (but not required) textbook:
- You must have a desktop or laptop capable of running Zoom and screen-sharing.
- You must have a phone or tablet capable of running Zoom with a camera that can show your workspace while taking exams.
- You must have a consistent internet connection for streaming video through Zoom, and watching lecture videos.

**Lectures**

All lectures are distributed via pre-recorded videos available each week on the course website. You are welcome to work ahead of the schedule, however, some videos will be updated as the semester progresses. Watching the videos closely (I recommend taking notes) is essential to being able to complete the labs, assignments, and exams in this course.

**Laboratory Sessions**

Lab section 1-8 will be conducted in STEM 2201. Performing the work assigned in your laboratory section is *mandatory*. Repeatedly failing to earn credit for your laboratory section will reduce your final grade (see below). Lab section 730 is by default asynchronous online.

The lab sessions are designed to be learning tools that complement the lectures, as well as collaborative experiences where students work with each other and the Teaching Assistant to complete the exercise. Students who are late to lab might not receive credit for that session. Attendance is based on a good faith effort, i.e., the TAs will decide if you have put in a sincere effort to participate actively, collaborate with others, and attempt to complete the exercise.
If your lab falls on a day when the university isn’t holding classes, you will be given credit for the lab. However, you are responsible for reading the lab and being comfortable with the content covered even if you aren’t required to complete it (lab content will appear on homework assignments and exams).

Section 730: Students that are assigned to this section, or those who wish to not attend their in-person section, can complete the Coding Rooms assignment by Friday at 10pm for lab credit. There will also be an optional, weekly, synchronous, Zoom lab session offered for those who would like to complete the lab with a parter and with TA supervision as mentioned previously.

Communication

In-person communication can be done at help room. For all other circumstances, communication with the instructors should be done through Piazza (with a public or private post). Emails will not be responded to.

All class announcements will be made through Piazza, as well as direct messages from the instructors. The instructor reserves the right to modify course policies, scheduling, and assignment specifications as necessary.

Grading

A total of 1000 points can be accrued over the course of the semester. There will be no extra credit.

<table>
<thead>
<tr>
<th>Points</th>
<th>GPA</th>
</tr>
</thead>
<tbody>
<tr>
<td>900-1000</td>
<td>4.0</td>
</tr>
<tr>
<td>850-899</td>
<td>3.5</td>
</tr>
<tr>
<td>800-849</td>
<td>3.0</td>
</tr>
<tr>
<td>750-799</td>
<td>2.5</td>
</tr>
<tr>
<td>700-749</td>
<td>2.0</td>
</tr>
<tr>
<td>650-699</td>
<td>1.5</td>
</tr>
<tr>
<td>600-649</td>
<td>1.0</td>
</tr>
<tr>
<td>0-599</td>
<td>0.0</td>
</tr>
</tbody>
</table>
Exams

There will be a midterm and a final exam. The midterm is worth 100 points, and the final is worth 150 points.

Projects

There will be 3 projects over the course of the semester, contributing a total of 300 points to your final grade. Each project will be due at 10 PM on a Thursday. The projects will include the design and implementation of solutions using C++. Project submission and grading will be done via Coding Rooms.

All assignments (projects and homework) are to be done individually. Discussing problem-solving strategies with other students is encouraged, but as soon as the discussion turns to a particular implementation (for instance specific code), the work must be your own. You can make private posts on Piazza asking for help. See the Academic Honesty section below.

Labs

You can fail to earn credit for a lab twice without penalty. For each lab missed beyond two, you will receive a 0.5 GPA reduction.

For example, if a student was previously earning a 4.0, but missed 4 labs, then their overall course grade would be reduced to a 3.0 \((4.0 - (2 \times 0.5) = 3.0)\).

Homework

A homework assignment will be due every Monday and Thursday at 10 PM. Each homework assignment is worth 30 points. There will be 17 assignments given, but only the top 15 assignments will count toward your point total (the two lowest-scoring assignments will be dropped). Thus, you can earn a total of 450 points from the homework assignments.

All assignments (projects and homework) are to be done individually. Discussing problem-solving strategies with other students is encouraged, but as soon as the discussion turns to a particular implementation (for instance specific code), the work must be your own. You can make private posts on Piazza asking for help. See the Academic Honesty section below.

Re-Grades

We are often generous when we first grade something, so please be sure that we did, indeed, make a mistake before you submit your request. The entire assignment will be regraded, not just a specific part. All requests for re-grades must come within one week from the return of
the graded item. No requests will be considered thereafter, so please be sure to review your returned assignments promptly.

Honors

An Honors Project will be released midway through the course for those that are enrolled in the Honors College and wish to receive Honors credit. You’ll need to:

- Fill out an Honors Option Agreement Form.
  - You shouldn’t fill out the form until the Honors Project has been released, as you need to provide a detailed description of the work to be performed.
- Complete the Honors Project with full points by the given due date.
- Earn a 3.5 or 4.0 in the course.

The Honors Project is not worth any points, nor is it extra credit. We will not provide assistance in completing it, as the Honors Project is meant to be solved independently of additional instruction.

Extensions

Make-ups for examinations may be arranged if your absence is caused by a documented illness or personal emergency. A written explanation (including supporting documentation) must be submitted to your lecture instructor; if the explanation is acceptable, an alternative to the examination will be arranged. When possible, make-up arrangements should be completed in advance.

A student who is unable to complete an assignment by the specified due date because of illness or a personal emergency should contact their lecture instructor. If the student’s explanation is acceptable, the assignment due date will be extended (or their lecture instructor will make other appropriate arrangements).

A student is allowed to miss up to two laboratory sessions without penalty, so illness or personal emergency can be accommodated without needing to notify us. If a long-term illness or personal emergency consumes more than two, the student needs to contact their lecture instructor.

Academic Honesty

Spartan Code of Honor
As a Spartan, I will strive to uphold values of the highest ethical standard. I will practice honesty in my work, foster honesty in my peers, and take pride in knowing that honor in ownership is worth more than grades. I will carry these values beyond my time as a student at Michigan State University, continuing the endeavor to build personal integrity in all that I do.

**Collaboration**

Each assignment submission is electronically and manually compared to all other solutions (in this course and previous offerings) to identify similar solutions.

Under no circumstances should you share any part of an assignment solution with another student. Simply showing your solution to another student almost guarantees detection by the plagiarism software. Past experience shows that a student who asks to “look at” your solution will copy parts of it or pass it along to someone else who copies it.

Plagiarism (unsourced use of other’s intellectual property) is not allowed. However, citing and using other’s works is generally fine (please ask if uncertain) as long as the material wasn’t made specifically for solving the assignments in this class. The use of material or code from a student that previously took the course is an explicit instance of academic dishonesty. Intellectual collaboration with other students in the class is allowed, but each student should write (and not share) their own code. If a student submits code that they don’t understand, such is also an act of academic dishonesty.

There is one exception to the above policy: labs. The lab assignments are intended to be solved collaboratively, and you are allowed to share your solutions with other students. However, you may only share with your partner to allow everyone the opportunity to solve the assignment. Please refrain from making public posts about lab solutions while the lab is currently active, so that other students have the opportunity to think through it.

**Examples Of Academic Misconduct**

A goal of this course is to teach professionalism. Any instance of academic dishonesty will be viewed as evidence that this goal has not been achieved, and will be grounds for receiving a final grade of 0.0. Examples of academic dishonesty include (but are not limited to):

- Copying another student’s code or exam answers.
- Using code implemented by someone else intended to solve this class’ assignments (i.e., don’t get someone else to do the assignment for you).
- Using code independently implemented by someone else without attributing credit (i.e., you can use tools, libraries, or code snippets from the web, but only with proper citation).
• Writing code that deceptively passes the test cases, but doesn’t solve the problem given. In other words, abusing automatic grader mechanisms to gain unearned points (hard-coding).

• Using sources whose purpose is to provide assignment solutions (e.g. Chegg, YahooAnswers).

• Distributing course content without instructor permission.

• Having another person that isn’t an instructor or TA aid you in implementing a solution to an assignment.

• Submitting a solution that you don’t understand / can’t explain to an instructor.

• Using assignment solutions provided to a previous iteration of this course.

• Providing false information to the instructor about matters related to the course.

• Facilitating another student in any of these activities.

Real Instances Of Academic Misconduct From This Course

The following stories are real cases of academic dishonesty that I’ve encountered in my classes. I’m describing them here so that it is clear to each of my students what is allowed in my classes. In all cases of academic dishonesty, I will file an Academic Dishonesty Report and usually fail or severly reduce the student’s grade.

• A student found a solution written by another student (who took the class in a previous year). That student modified the solution (changed variable names, restructured code) and submitted the assignment.

• A student worked closely with another student on an assignment and they showed each other their code in the computer lab. Although no files were exchanged, nor was there any copy/pasting, their solutions converged enough to be detected by my plagiarism detection software.

• A student found a solution to a similar problem in the class written by a third-party (someone not associated with the class). The student submitted a solution based on the found solution, without attribution to the original source.

• A student was posting homework questions to StackOverflow and soliciting for other’s to solve their assignments for them, instead of just asking the instructors privately for assistance.

How To Provide Attribution

You must provide attribution if you make use of sources beyond the material given to you in this class. The attribution should be commented in your code and/or added to a README file included with your assignment. Please ask if you are uncertain as to if a source is allowed to be used in your assignment.
Example solution with attribution:

# The assignment is to read in a string that looks like "010 001 0101"
# and print out sum of the binary numbers with zero padding.

input_str = input()
bin_strs = input_str.split()

def convert_bin_str_to_int(bin_str):
    # Discovered int function from:
    # http://stackoverflow.com/questions/8928240/convert-base-2-binary-number-str
    return int(bin_str, 2)

# In discussion with Grant King, we found that the map function
# would be useful in converting both numbers
ints = map(convert_bin_str_to_int, bin_strs)

# This source helped me determine how to print binary numbers with padding:
# http://stackoverflow.com/questions/13599638/how-to-return-a-number-as-a-binary
print("{0:06b}".format(sum(ints)))

Sources which don't require attribution:

- Class-related materials (lectures, example code).
- Documentation from the linked reference websites (python.org, cppreference.com, cplusplus.com).

Everything else, including conceptual discussions with other students should be attributed.

Department Policy Concerning The Use Of Chegg and Similar Sites

This course has adopted the Chegg and Similar Sites policy. Submission of student work
(e.g. assignments and/or exam solutions) based on those found on Chegg, Brainly,
Quizlet, and other similar websites will result in an Academic Dishonesty Report (ADR)
and an automatic failing grade of zero (0.0) for the course. The ADR for students
personally posting questions from assignments or exams to these sites will request
additional sanctions.

Extenuating Circumstances

The goal of this class is for you to learn. If you find that anything is coming in your way of that
goal, please talk with us about it. We plan to keep the class flexible to the learning styles that
seem to work best for the students, so feedback is always appreciated. In particular, due to
the ongoing global pandemic, we understand that you might need additional support. Please
don’t hesitate in reaching out to us for anything that might affect your learning.

Grief Absence Policy

If there are unfortunate circumstances that would lead you to have unexpected absences,
MSU has a Grief Absence Policy. Please contact the Associate Dean, and we will make every
effort to aid you in continuing the class after we receive confirmation from the administration.

Resource Center for Persons with Disabilities

Michigan State University is committed to providing equal opportunity for participation in all
programs, services and activities. Requests for accommodations by persons with disabilities
may be made by contacting the Resource Center for Persons with Disabilities at 517-884-
RCPD. Once your eligibility for an accommodation has been determined, you will be issued a
verified individual services accommodation (“VISA”) form. Please present this form to me (Dr.
Nahum) at the start of the term, or as early as possible.

Religious Observances

Accommodations will be made for religious observances if requests are made well in
advance. Since religious observances are usually known at the beginning of class and
assignment due dates are in the schedule, we expect students to request accommodation at
the beginning of the semester. See MSU’s Religious Observance Policy for more details.

Acceptable Use Policy

All students are expected to be responsible users of the computer system and laboratory
classrooms provided for this course. Account usage guidelines published by the Department
of Computer Science and Engineering are posted under the CSE AUP.