Instructor Information

<table>
<thead>
<tr>
<th>Name: Sam Midkiff</th>
<th>Website: <a href="https://engineering.purdue.edu/~smidkiff">https://engineering.purdue.edu/~smidkiff</a></th>
</tr>
</thead>
<tbody>
<tr>
<td>Office: EE 310</td>
<td>Office hours: See Piazza (<a href="https://piazza.com/class/jqjy1unszpi18u">https://piazza.com/class/jqjy1unszpi18u</a>). I will usually hang around for a few minutes after class to answer questions.</td>
</tr>
<tr>
<td>Email: <a href="mailto:smidkiff@purdue.edu">smidkiff@purdue.edu</a></td>
<td>Email and piazza are the way to reach me. Put 30862 in the subject line.</td>
</tr>
</tbody>
</table>

In general, questions should be submitted to piazza so that all students have the benefit of the question and answer. If you send technical questions by email I may well post the question and answer to piazza.

Teaching Assistant Information

Sara Aghajanzadeh and Austin Ketterer are the Graduate TAs for the class. Undergraduate TAs are Rtvik Sriram Bharadwaj, Shuham Bhokare, Whutao Wang, Sruthi Reddy and Ziyu Guo. The TAs will handle the programming project, and all questions related to the programming project and programming homeworks should be directed to them or Piazza. I will answer Java and C++ questions, although these can be directed to the TAs as well.

TA office hours will be available on Piazza.

Course Description

This course is intended to teach you C++, Java, and some details about their implementation. It will include two large projects, homeworks, and many of them involve programming. There will be quizzes that will not be graded for correctness, but will serve as a way to take roll and offer you practice test problems. These will be given randomly, typically at the beginning or the end of class.

Prerequisites (if needed)

You should be competent in programming a non-object oriented language and have taken ECE 26400.

Learning Outcomes
i. an understanding of the concepts of inheritance and polymorphism. [None]
ii. an ability to overload operators in C++. [1,2]
iii. an understanding of the difference between function overloading and function overriding. [None]
iv. an ability to incorporate exception handling in object-oriented programs. [None]
v. an ability to use template classes and the STL library in C++. [None]
vi. an ability to write object-oriented programs of moderate complexity in C++. [1]
vii. an ability to write object-oriented programs of moderate complexity in Java. [1]
viii. an ability to write programs with multiple threads and use synchronization among threads. [None]

Suggested Text

How to Succeed in This Course
You will succeed in this class using the same strategies that work for most Comp E classes: attend class and pay attention, ask questions and talk to Prof. Midkiff or the TAs if you don’t understand concepts, understand the last lecture’s material before going to the next lecture and attend class. And start on the projects as soon as you can. Programming, like most things in life, usually takes longer than you think it will.

Policies
General Course Policies
• Please try and arrive on time. With 200+ students, having a small percent coming in late will be disruptive.
• Computers can be used, don’t talk on cell phones and mute them.
• It’s a large class, but feel free to ask questions.

Grading
Grade ranges are:
90 – 100: A
80 – 90: B
70 – 80: C
60 – 70 D
Below 60: F
When assigning grades, the cutoffs may be lower than what is shown. But if you make in the ranges above you will get at least the grade indicated. A- may be given to the top Bs, B- to the top Cs, etc. A+ may be given to the top As, B+ to the top Bs, etc.

**Tests:** 3 tests will be given and each will be worth 14% of your grade, for a total of 42%.

**Projects:** The C++ project will be worth 25% of the grade and the Java project will be worth 20% of your grade, for a total of 45%

**Homework:** Homworks, in total, will be worth 7% of your grade. At least 10 homeworks will be given.

**Quizzes:** In-class quizzes will be worth 6% of your grade. 10% of the quiz scores will be dropped.

**Academic Dishonesty**
I dislike cheaters. It harms your fellow student, it creates extra work for me, usually at a bad time in the semester. If I catch you cheating, I will, at a minimum, flunk you on the assignment, turn you in to the department and turn you in to the Dean of Students. I may flunk you, depending on how severe I judge the cheating to be.

- You may discuss the projects with fellow students. You must not share code.
- You may discuss homeworks together. You may collaborate on homeworks. Remember that the primary purpose of homeworks is to help you understand concepts and prepare for tests: not doing the work yourself and understanding what you turn in will only hurt you.
- In class quizzes are not graded for correctness, but to keep the room quiet during quizzes do the work on your own.
- Test will be open notes and open book. Make-up tests will be closed notes and closed book unless announced otherwise. Use of electronics is prohibited on all tests. You must do your own work on tests.

More information can be found at https://www.purdue.edu/odos/osrr/academic-integrity/index.html

The Purdue Honors Pledge states: As a boilermaker pursuing academic excellence, I pledge to be honest and true in all that I do. Accountable together - we are Purdue.

**Use of Copyrighted Materials**
Unless otherwise stated, you may distribute notes, homeworks, old test and solutions from this course as long as any copyright notices remain and they are not used for any commercial purpose.

**Attendance**
I expect you to attend class. Many classes will have a quiz at some point in the class. I will drop some portion of the quiz grades, so you can miss some classes. Email me if you will need miss class for a good reason.

If you will have to miss an exam, let me know as soon as you know you will miss the exam so that we can arrange an alternate exam.

**If you are sick with a fever, avoid class – there is no need to infect the rest of us.**
If you are sick and need to miss a test go to PUSH or another doctor and get a slip saying you visited. DO NOT come to my office when you are sick to let me know you are too sick to attend class, take a test, etc.

**Missed or Late Work**

Homeworks are not counted as late until I download them to grade. After that you may lose 50% credit.

Projects will lose some number of points a day for being late.

**Grief Absence Policy for Students**
If you have a personal loss or other personal disaster strike you, please let me know and I will try to be accommodating.

**Violent or Disrespectful Behavior Policy**
I will not tolerate violent behavior, either towards me or your fellow students, and you will be dropped from class.

**Disrespectful Behavior**
I will not tolerate disrespectful behavior in class. This includes being impolite with the TAs or other students, and being disruptive in class by talking, making too much noise, viewing inappropriate material in labs or class, eating in labs, etc. The consequences for this may include having you called out in class, being banned from TA office hours, to being dropped from the class.

**Emergencies**
In the case of fire, tornadoes, an active shooter or other emergency we will either shelter in place or leave. If this is during a test, the tests must be left on your desk, and will be collected and finished at a later time, unless they are burned up or carried off in a tornado.

Should facilities necessary for class or completing projects be unavailable appropriate extensions will be given to finish work.

**Accessibility and Accommodations**
Purdue University strives to make learning experiences as accessible as possible. If you anticipate or experience physical or academic barriers based on disability, you are welcome and encouraged to let me know so that we can discuss options. You are also encouraged to contact the Disability Resource Center at: drc@purdue.edu or by phone: 765-494-1247.

**Nondiscrimination**
Every person should be treated with respect regardless of race, religion, personal beliefs or sexual orientation.

**Class Schedule**

**Test dates:**
Tuesday 09/24, 6:30 – 7:30, EE 129 & EE 117  
Thursday 10/24, 6:30 – 7:30, WTHR 200  
Wednesday, 11/20, 8:00 – 9:00PM, CL50

If you need to miss a test, contact me as soon as possible about a make-up date. Make-up exams may be closed note and closed book.

*There is no final in the class.*

**Tentative Project due dates (they won’t get earlier):**
C++ (team) project: Sunday, 11/3, 11:59PM  
Java (individual) project: Sunday 12/8, 11:59PM (the last day of dead week)

**Days classes are canceled because of evening exams.**

These may change, but are unlikely to. If you need to schedule travel and want to be sure, ask me and I’ll make a commitment one way or another.

Wednesday, Oct. 9 (Wednesday of October Break week)  
Friday, Oct. 11 (Friday of October Break week)  
Monday, Nov. 25 (Monday of Thanksgiving week)

**Topics covered:**

- Objects and Classes
- Inheritance and Polymorphism
- Encapsulation
- C++
  - Program structure
  - Class examples
• Encapsulation and polymorphism
  • Object instantiation
  • inheritance;
  • Casts and overloading
  • Static functions
  • Initializers
  • Abstract Classes
  • Parameter passing
  • References
  • Destructors
  • Object assignments
  • Copy Constructors
  • Friend functions and operator overloading
  • Containers (STL)
  • Exceptions and assertions
• Java
  • Inheritance and overloading
  • Overloading
  • Simple constructors
  • Static methods
  • variables, strings and arrays
  • Parameter passing
  • The Object class
  • Assertions and exceptions
  • Cloning
  • Threads and synchronization
  • Games in Java
  • Garbage collection
• Function resolution in Java and C++
• Smart pointers in C++
• Multiple inheritance in C++
• C++11 Threads
• Java reflection
• Java Generics
• Other Java/C++ features

Disclaimer
This syllabus is subject to change.