2017 AP Computer Science A Syllabus

Mary Davis
University Christian High School

1 Welcome to APCS

Welcome to AP Computer Science at University Christian High School! This year marks the 1st year APCS is being offered here and I hope you share in my excitement and enthusiasm as we embark on a semester deep dive into software development using the Java programming language.

2 Course Description

AP Computer Science is an approximate equivalent to a one- to two-semester, university-level introductory computer science curriculum. By taking this class, students will learn to

- design, implement, debug, and document computer programs;
- write programs using the Java programming language, an industry standard;
- design and implement modular software components that can be tested, integrated with others, and reused;
- represent information in an object-oriented manner; and
- read and understand APIs, which enable you to learn how to use other language features on your own later.

3 Unit Outline

The list that follows is an approximate ordering of topics covered in this course.

Week 1: Syllabus, Procedures, FRQ Practice, File Reading Basics, Zip Code Example, Rapid Review1: Java languange	Week 2: Rapid Review 2: Classes & Objects; Rapid Review 3: Strings; Magpie Lab; Assessment	Week 3: Rapid Review 4: Data Structures; Pictures Lab; Assessment
Week 4: Inheritance & Polymorphism; MC Practice; FRQ Practice; Assessment	Week 5: Abstract Classes and Interfaces; MC Practice; FRQ Practice; Assessment	Week 6: Recursion; MC Practice; FRQ Practice; Assessment
Week 7: Sorting & Searching Algorithms; MC Practice; FRQ Practice; Assessment	Week 8: Elevens Lab; MC Practice; FRQ Practice; Assessment	Week 9: Practice Exam 1; Analysis of Practice Exam 1
Week 10: Barron's Chapter 1: Intro Language Concepts; Barron's Chapter 2: Classes & Objects; Assessment	Week 11: Barron's Chapter 3: Inheritance; Assessment	Week 12: Barron's Chapter 4: Standard Classes; Barron's Chapter 5: Program Design; Assessment
Week 13: Barron's Chapter 6: Arrays and Array Lists; Barron's Chapter 7: Recursion; Assessment	Week 14: Barron's Chapter 8: Sorting & Searching; : Barron's Chapter 9: Labs; FRQ Practice; Assessment	Week 15: Practice Exam 2: Analysis of Practice Exam 2
Week 16: Practice Exam 3: Analysis of Practice Exam 3		

4 Resources

4.1 Textbook

The textbook for this course is *Big Java*. The course problem sets will have required reading from the textbook and **you are responsible for doing the reading!** In fact, if you do not read the required sections in the textbook before attempting to complete parts of each problem set, you will find yourself spending far too much time figuring out how to answer the questions. The textbook is clearly written and succinct, so you won't have to read many pages to learn what is necessary.

4.2 AP Review Book

Every student is required to purchase Barron's AP Computer Science Review Book.

4.3 Oracle's Java 7 API — and the AP subset

The entirety of Java's built-in libraries is explained in detail on Oracle's website. The portion of Java that you'll necessarily need to succeed in this course and on the AP exam can be found here: http://www.cs.duke.edu/csed/ap/subset/doc/. That link will prove very useful, so bookmark it!

4.4 Software

All of the software that will be used in this course is available for free, and much of it can be run on multiple platforms (e.g., Windows 2000/XP/Vista, Mac OS X, and Linux). You should have everything you need from CS 2.

- **jGrasp** — a student grade IDE
- **Oracle's Java Developer Kit (JDK)** — command-line tools that allow for compiling and executing Java programs; combined with a text editor like Notepad++, this makes for a very light and fast Java development platform

4.5 Computers

Students are required to have a working computer each day in class. AP CS is the first class of the day, so computer should be charged overnight and ready to work on battery power.

4.6 Managing Your Data

Students are required to use Google Drive to store and turn in working programs. Students will need to build a filing system inside their shared Google folder, labeled by weeks: Week 1-2, Week 3-4, etc. "I forgot to upload" will not be accepted as an excuse for late work. (no late work will be accepted).

4.7 Course Website and email

A website has been developed for this course and is available at http://uchsdavis.weebly.com. There you will find all handouts, problem sets and course news. We will also be trying out Google Classroom. This site will also allow you to maintain contact with me and your classmates, both during and after school hours. You can ask questions in a forum manner and help others with their questions. Once students have registered for accounts on the website, a course mailing list will be created. That email list will be used to broadcast changes to due dates, reminders, and notifications of when grade updates are posted on MSP.

5 Teacher's Expectations

It is expected that you will be learning from various places, including lessons, reading, and your own tinkering with code. The degree to which you rely on each is a matter of your own learning style. I respect that everyone has their own style, but I also expect that you draw experience from *all* of those sources.

When in class, it is extremely important to be paying attention during lessons and explanations. That means that you might have to stop work on a problem momentarily to hear what I say. *And it certainly presupposes that you are not using a computer for any kind of non-class activities.* Your attention is considered a matter of educational necessity and politeness to me and your peers; your lack of it will be considered a failure to gain participation credit when it comes time to compute your grade.

All students agree to. . .

- respect your classmates, teacher, and equipment;
- be on time and working immediately, which entails staring the *Daily WarmUps*;
- make progress on the current problem set on a **daily basis**, even in cases when there is no strict due date for interim sections;
- read all assigned sections from the textbook, which are listed throughout the problem sets;
- supply your own effort for your own work (i.e., not giving others your work or vice versa);
- provide *appropriate* assistance to your peers (refer to prior bullet point!), as not everyone will learn all topics at the same speed or understand abstract concepts in quite the same way;
- operate computers in an ethical manner, especially when personal information is involved; and
- LEARN!

6 Grading

Grades will typically be updated on a one- to two-week basis, with grades being posted on MSP.

6.1 Points Allocation

WarmUps 5%; In-class programming/Problem Sets 40%; Quizzes 40%; Programming HW 10%; Binder 5%

6.2 Late Work

This is an AP College Level Class - No late work will be accepted, unless you are absent. Contact me for RARE exceptions.

7 Problem Sets

7.1 Overview

Just as formal assessments measure what you have learned, formative assessment — in the form of *problem sets* — will let me measure how your learning is progressing and how much effort you are putting forth all along. At any given time, there will generally be a problem set you are expected to be working on. Generally, lessons will conclude with a listing of problem set sections you are expected to work on and complete for homework. Expect that some class time will be used to check up on your problem set progress; this is how you prove that you are doing your homework all along before the final due date for a given problem set. *Checkpoint scores* may be given for certain problem set sections, and these values will be used when tallying up your final problem set score.

7.2 Structure

Problem sets can consist of any of the following:

- *Introductory Notes* are provided to prepare you for the material that will be presented. Sometimes, this text is above and beyond what your textbook provides, so it is important thatyou take the time to read all I have written.
- *Required reading* tells you which sections of the textbook you need to read **before** attempting to proceed further in the problem set.
- *Book problems* are those problems from the textbook or Review Book that you are required to work on.
- *Teacher-written problems or FRQs* are those problems that either don't appear in the textbook or are written in a different manner.
- *Programming assignments* generally describe some programming task for which you will be asked to provide either (a) written source code, (b) demonstrate a working program, or (c) both. **These assignments** typically have a set point value and **are significant to your problem set grade!**

7.3 Quality of Work

Just as grammar, spelling, neatness, and overall presentation of a deliverable matter in industry (and academia!), those elements do count for your problem set grade. Sloppy work that isn't well written does not reflect well on your effort level. Just as this very document is well formatted and clearly written, I expect the same from you. Those who would judge you based on your work product — teachers, managers, prospective clients, etc. — are often gatekeepers to your future success. That said, it doesn't hurt to impress. (By the same token, one can go overboard by spending too much time on aesthetics and too little on content.)

8 Getting Help

If you are having difficulty with a particular concept or part of a problem set, you are highly encouraged to do the following, in order.

- Re-read the question in case you missed important details.
- Use debugging techniques you are taught in class: insert print statements to show variable state at various places, test tricky lines of code in a small program where you control the conditions, etc.
- Make sure you try to understand any compile-time errors that are generated.
- Look carefully, line by line, for tricky syntax errors: missing semicolons, unmatched curly braces, missing a capitalization, etc.
- If in the computer lab, ask a classmate to have a look at your code. When outside of class, use the web site's problem set posting as a place to post a question.
- Ask me. Office hours (after-school and lunchtime tutoring) will be announced, and it's expected that you will come to them to get help with topics/questions that are confusing for you. This time is also good for getting programming assignments checked off in your problem sets.

9 Collaborative Work Policy

Part of becoming a computer scientist, software architect, developer, QA person, etc., entails working with peers in some capacity. I *encourage* students to work together to simulate that real-world experience. In addition to the preparation for what comes later, you will learn how to interpret the code of others, what good comments in code are (and are not!), and the importance of good programming style. The most direct advantage is this: Working with others can streamline your time spent being stuck and debugging your code.

The main disadvantage of working with others can be the temptation to breach a code of ethics that I make my students uphold. For example, let's say you're having a hard time figuring out how to write a program that finds the n^{th} Fibonacci number in a recursive manner. You ask your friend, who is also in the class, "Hey, what's wrong with my code?" After she has a look, she decides that it's all wrong and not really worth trying to fix. She suggests, "You might want to start over. Here, have a look at what I have." Sounds innocent enough, but you can probably make a few small mental leaps between where this story ends and having lines of her code embedded within your own.

In these days of instant messaging, email, web forums, etc., opportunities to *borrow* code abound. Do yourself a favor and **make sure your discussions with others stick to methodology**, or **how to do something**, not adapting specific chunks of others' code.

10 Academic Honesty Policy

Universities have established policies for dealing with academic dishonesty, and you will be held to a similar standard while in this class. **Under no circumstances are you permitted to present anyone else's work as your own.** I understand that it can be frustrating when you can't get the right answer, but there is more than enough support available to help any student who is having difficulty. Penalties for academic dishonesty may include — but are not limited to --- grade reductions, referrals to school administration, and notification being sent to the colleges to which you apply.

One contributing factor to academic dishonesty is falling behind on your work, a.k.a. Procrastination. If you fall too far behind, it becomes *impossible* to catch up without cheating. I would rather you come talk to me about the situation instead. First, I will respect your honesty. Second, I may be willing to make an accommodation to help you get back on track, like pushing back a due date for some agreed-upon grade penalty. Of course, if you have some legitimate issues preventing you from getting your work done, I will be willing to work out an arrangement with you provided that you do the responsible thing .

Parent Letter for AP Computer Science, 2017

Dear Parent/Guardian:

I hope that you have had an opportunity to read over this syllabus and discuss it with your son or daughter. If you have any questions or concerns, please don't hesitate to email me at *mdavis@uchigh.com*

Grades for this class will be made available on MSP. Course materials will be posted to our class website or Google Classroom.

Students will need the following supplies for the class: 1 3 ring binder, several 2 pocket folders, a supply of graph paper for handwriting programs and taking notes, or a graph-paper notebook for the same reason. We use the whiteboards almost everyday, so I request students donate a pack of dry erase markers for the class to use, chisel tip, if they are able. Students are also required to purchase Barron's AP Computer Science Review Guide - $10.69 From Amazon.

So that I may be in touch with you with grade updates and such, please provide your contact information to me by filling out the form below, separating it from this letter at the dotted line, and having your son or daughter return it to me. Thank you in advance!

Sincerely yours,

Mary Davis
University Christian High School

.....*cut here*.....

Parent Information for AP Computer Science, 2017

Student's Name: _____

Your Name(s): _____

Home Phone: _____ Best time to call: _____

Work Phone: _____ Best time to call: _____

Cell Phone: _____ Best time to call: _____

Email Address: _____

How often do you check email? _____

Signature: _____ Date: _____

Please write any comments/concerns you have on the back of this form. Thank you!