Computer Science
or AP® Computer Science Principles
Period 2 (9:50–10:40) or Period 3 (10:40–11:30) • 2023-2024
Mr. Holzer • tholzer@ahsmail.com

"Every discovery in science and art, that is really true and useful to mankind has been given by direct revelation from God, though but few acknowledge it. It has been given with a view to prepare the way for the ultimate triumph of truth, and the redemption of the earth from the power of sin and Satan. We should take advantage of all these great discoveries, the accumulated wisdom of ages, and give to our children the benefit of every branch of useful knowledge, to prepare them to step forward and efficiently do their part in the great work."
—Brigham Young, 1862

Course Description

Computer Science is a required course for all 9th – 11th graders at American Heritage School – Salt Lake City. This hands-on, project-based, introductory course is designed to help you learn the fundamentals of computer programming, regardless of whether you have had previous programming experience. If you can tell the difference between a computer and a toaster, you have all the prior experience needed to succeed in this course.

Computers are powerful tools — arguably some of the most powerful that our race has ever invented. The purpose of this course is to put the full power of the computer firmly within your grasp. By learning how to program a computer, you will gain the ability to wield the computer as a tool to accomplish whatever your life work entails, including blessing yourself, your family, and the world. Contrary to popular assumption, computer programming is a fairly straightforward skill to learn and is well within the grasp of every student at American Heritage. Further, whether or not you become a professional computer programmer, this class is for you, because computer programming skills can be applied *to every field of work* in which you will be engaged. For these reasons, we believe that everyone should learn computer science and thereby master the use of the computer as the powerful tool that it is — to bless yourself, your family, and others, and to build the kingdom of God on the earth.

In this course, you can expect to:
1) Participate in readings, discussions, and activities that will give you a broad understanding of computing, including the legal, ethical, social, safety, privacy, and spiritual implications that computers and computer technologies have on us. In a world that contains more computers than human beings and that requires any given person to interact with dozens of computers every single day, these are essential topics for us to discuss.
2) Develop core computer programming skills. Not only do these skills allow you to program computers, but they will also help you think analytically — a skill that can be applied to much of our life's decision making. We will use Python in this class.
3) Learn to use specialized programming tools that will make computer programming a useful skill to you, regardless of what your life's mission entails. Here is a sampling of some of the things you will learn to do.
   a. Analyze and visualize data — Data is present in nearly every field of study, and computers are especially good at processing large sets of data and displaying them in useful ways.
   b. Create graphical user interfaces (GUIs) — This skill will enable you to create computer programs that others can easily use and interact with.

c. Create computer graphics — Being able to draw and animate windows, shapes, and pictures on a computer screen opens the door to basic web development, app development, game development, and related fields.

d. Create simulations — Simulating real-world situations, especially those that are difficult or dangerous to study directly, is another great strength of computers, and is applicable to nearly all fields.

e. Understand the basics of artificial intelligence and machine learning — All reputable projections predict that artificial intelligence is going to become much more powerful very quickly. Therefore, understanding the function, strengths, and limitations of this technology will simply become more and more important. Though the inner workings of machine learning can be very complex, our class will engage in projects and experiments that illuminate the basics. For instance, students will be given the opportunity to train a few machine learning models in class.

4) Use all of the skills you have learned throughout the year to create a computer program in an area that interests you and that will bless others.

For those students who want to learn more computer science, this course serves as a preparation for and prerequisite to AP Computer Science A, which will likely be offered in the 2024-2025 school year.

Course Requirements

<u>Homework Assignments</u> — Complete and turn in the weekly homework assignment.

In the last class of each week, all required readings and homework assignments for the next week will be handed out to you in class. These materials will also be posted to Canvas, insofar as that is allowable by copyright law. If you lose or damage the printed copy that is provided to you, you will be responsible to print another copy from Canvas.

All homework assignments will be due at the beginning of the last class of the week and must be turned in in class. Electronic submissions will only be accepted if arranged for in advance. You are encouraged to begin working on your homework assignment early, so that you can ask questions and get help in class if you get stuck.

<u>Readings and Class Discussions</u> — Prepare for and participate in class discussions.

Throughout the year, class discussions will be held on important topics related to computer science. You will be expected to carefully prepare for each discussion by reading, annotating, and thinking about a specific article or talk that will be provided to you. On the day of the discussion, you can expect to discuss the article in small groups and as an entire class, and you should be prepared to be called on to share your insights and impressions. You will receive a grade for your participation in these discussions; if you carefully prepare, you will receive a good grade.

<u>Python Reference Guide</u> — Create your own personal reference guide for Python programming.

As Python programming concepts are presented and practiced in class, you will be expected to capture them in what will become your own personal reference guide. This will be a major focus of our class: nearly every day you will make, annotate, and refine these notes by hand, and then twice per term you will type them up. This ever-growing reference guide will be the primary source you will consult when working on programming projects. Your reference guide will be inspected and graded periodically.

<u>Programming Projects</u> — Complete each project by the due date.

We learn best by doing. For this reason, this course includes a series of programming projects which will help you apply the programming concepts that you are learning. Written instructions will be given to you for each project, along with a clear due date. Time in class will be provided for you to work on these projects and get help, but you should also expect to spend some time outside of class for each project. Completed projects will be submitted on Canvas.

<u>Term Projects</u> — Complete each project by the due date.

Each term, you will complete a project to help you learn about an important aspect of computer science. Except for your final project in Term 4, these projects will not involve programming, but will instead invite you to interview others, research topics, give presentations to the class, and/or write reports. Some class time will be given to work on these projects, but they will mostly be completed outside of class.

<u>Memorization</u> — Memorize one prophetic quote per term.

Each term, a prophetic quote related to our topics of study will be provided for memorization. We will review the quote periodically in class to aid in the memorization process, but it will be your responsibility to memorize the quote, complete the accompanying reflection, and pass it off. You may pass off your memorization during any class work time; I will simply ask you to repeat the quote from memory and turn in your reflection.

<u>Class Notebook</u> — Maintain your class notebook.

Throughout the year, you will be expected to maintain a class notebook containing your homework assignments, your annotated readings, your Python reference guide, the final source code and report for each of your projects, and reflections on what you are learning. Your best work is expected, including good handwriting and neat organization. Your notebooks will be inspected and graded a couple times each term.

<u>Science & Engineering Fair</u> — Submit and present a project at the annual Science & Engineering Fair.

Every year, all students at American Heritage School – Salt Lake City participate in the annual Science and Engineering Fair. This year, the Fair will be held on November 3, 2024. Completion and submission of a science or engineering project will be required of every student. Clear guidelines will be given in class. Though you will have periodic opportunities to ask questions and get help in class, the project will need to be completed at home. Deadlines for different stages of the project will be given in your weekly homework to ensure that you are on track with your project.

Grading

Grades will be assigned according to the standard American Heritage breakdown, as follows:

A = 100-93%   A- = 92-90%   B+ = 89-87%
B = 86-84%   B- = 83-80%   C+ = 79-77%
C = 76-74%   C- = 73-70%   D+ = 69-67%
D = 66-64%   D- = 63-60%
F = 59-0%
IN = Incomplete

A total of 1000 points will be awarded per term. Completion of class requirements will contribute to your grade in the following manner:

300 points: Programming Projects
200 points: In-Class Checkpoints
150 points: Homework assignments
150 points: Python Reference Guide
100 points: Term Project(s)
100 points: Notebook, Discussions, and Memorization

Clear instructions will be given for every assignment. If you follow the instructions carefully and do your best work, you will get a good grade.

<u>In-Class Checkpoints</u>
In-class checkpoints are each worth 4 points. On some checkpoints, there is an optional "stretch goal" provided for those who would like to earn 1 point of extra credit. The checkpoints will be graded in the following manner:

5 — Flawless + stretch goal
4 — Flawless
3 — Functional, but incomplete
2 — Functional, but incomplete
1 — Not functional

Course Materials and Equipment

Laptop and Software — Laptops will be available for use during every class period. These laptops cannot be taken home. However, each student is issued a Microsoft 365 account, which includes online access to Microsoft programs and Cloud storage space. You are encouraged to store your coding projects and documentation in the Cloud, so you can access them and continue working on them at home. You are also welcome to bring a USB flash drive to class for the purpose of transferring files between school and home.

Articles — The following articles will be read and discussed in class, either in full or in part.
- "'A century of living': In celebration of his 99th birthday, President Nelson reflects on innovation, growth, political breakthroughs" (Church News, 10 September 2023).
- "I, Chatbot" by Sara Smith Atwood and ChatGPT (Y Magazine, Spring 2023)

Talks — The following talks from leaders of The Church of Jesus Christ of Latter-day Saints will be read and discussed in class, either in full or in part.
- "Things As They Really Are" by Elder David A. Bednar (CES Fireside, 3 May 2009)
- "To Sweep the Earth as with a Flood" by Elder David A. Bednar (BYU Campus Education Week, 19 August 2014)

Books — Selected portions of the following books will be read.
- *Blown to Bits: Your Life, Liberty, and Happiness After the Digital Explosion, Second Edition*, by Abelson, Ledeen, Lewis, and Seltzer (Pearson Education, Inc., 2021)
- *Clean Code: A Handbook of Agile Software Craftsmanship*, by Robert C. Martin (Pearson Education, Inc., 2008)
- *Debugging: The 9 Indispensable Rules for Finding Even the Most Elusive Software and Hardware Problems*, by David J. Agans (AMACOM, 2002)

Other materials will likely be identified and become part of the required reading list throughout the year. These changes will be communicated clearly to you and to your parents.

Class Policies

Respect for People — The highest standard of respect for others is expected in our class: for your classmates, for all faculty and staff, and for any volunteers who may assist us in any way. I promise to treat you with the same level of respect, for I view the opportunity to interact with you in class as a profound privilege.

Computer Use — You will have almost daily access to the school laptops. You will be expected to use them respectfully and only for the purposes outlined in class. Additionally, you should take the utmost care to keep your computer safe from Internet viruses and other malware.

Phone Use — In accordance with the policies of American Heritage, phone use in class is not permitted.

Due Dates — All work is due on the set due date. You should plan ahead, knowing that more often than not, newly coded computer programs will not initially function correctly, requiring additional time for debugging and testing. However, if you feel that you need more time, talk with me before the due date. I will be happy to make special arrangements with you, but only if you talk with me beforehand. Otherwise, all late work will incur a 20% late penalty. The only exceptions to this policy are the weekly homework assignments and the in-class checkpoints; those may be submitted late for full credit.

Cleanliness — We all need to work together to keep our classroom clean. It is expected that everyone will help to clean up at the end of each class. Once you have finished cleaning up your own work area, please look for opportunities to help others clean up as well.

<u>Group Work</u> — Collaboration is an essential skill in the field of computer science, and you will nearly always be allowed to collaborate with your peers when completing class projects. However, you must complete each project yourself; you cannot turn in anyone else's work.

Questions
Do I need to be able to program at home?

Yes. Time will be provided in class for you to work on your programming projects, but you will usually need to spend some additional time at home to complete them. This will require that you install the latest stable version of Python and a programming text editor on a computer which you can access outside of class. If this poses any problems for you, please talk with me; I will be happy to make accommodations for you.

The installer for the latest version of Python (currently Python 3.11.5) can be downloaded from https://www.python.org/downloads/. You are allowed to use any programming text editor you wish, but I would strongly suggest that you use Visual Studio Code, as this is the editor we will use in class. You can download Visual Studio Code from https://code.visualstudio.com/. A step-by-step guide for installing both Python and Visual Studio Code will be provided to you during the first week of class. If you run into any problems with installation, please reach out to me early so we can get them resolved.

May I use an IDE (integrated development environment) at home?

You should not use an IDE to complete your programming projects at home. There are several reasons for this:

1) IDEs often provide features which do part of the work of programming for you (auto-filling commands, correcting typing errors, highlighting syntax errors, and so forth). At the early stages of programming, learning how to deal with these problems yourself is a crucial skill to learn. Though IDEs are amazing tools for increasing the productivity of programmers, you should not be dependent upon them.
2) Learning to use the extra features provided by an IDE can distract from learning how to actually program, especially in the beginning of your programming journey. For those interested in learning to use an IDE for debugging, compiling, and releasing software, these skills will be taught in AP Computer Science A, which will be offered beginning in the 2024-2025 school year.
3) IDEs are usually large, complex software packages. As this is a beginning computer science course targeted at every student, my intention is to show you that computer programming does not require the use of complicated software. I believe this will make it more likely for you to continue using the skills you learn in this class long after the end of this school year.

Why Python?

Many different programming languages could have been selected for this course, each with different strengths and weaknesses. JavaScript, HTML, and CSS are heavily used in web development; Java is used to write cross-platform applications and mobile apps; and C and C++ are standard languages often used for programming embedded devices. Any of these languages could potentially be used to introduce you to computer programming. However, after careful consideration, Python has been selected as the programming language for this course because of its versatility, universal support, and ease of use.

This course is designed to be taken by all students, to equip them with enough programming knowledge that they could use computer programming in any other area of interest throughout their lives. After taking this class, you may choose not to learn more programming. With that in mind, I asked myself the following question: <u>If a student were to learn only one programming language, which would be the easiest for them to learn and provide the greatest long-term usefulness to them?</u> With these criteria in place, Python seemed to be the best choice.

Using Python will get you creating meaningful programs quickly, without getting lost in some of the more detailed complexity inherent in other languages. While those details are certainly important for those who choose to continue learning computer programming and want more control, I don't want students to give up due to unnecessary complexity right at the beginning of their programming experience. To employ the use of an analogy, a music teacher generally would not choose to teach a toddler how to play the organ; instead, the toddler would start by learning the piano, where he does not need to bother with the complexity of pedals and stops and multiple registers. In much the same way, Python does not require the immediate understanding of many low-level concepts that can be difficult for beginners to grasp.

But please don't misunderstand me: Python is not a play language! To extend the use of our music analogy a little further, just because the piano is usually the first instrument keyboard musicians learn does not mean that it is an inferior instrument. It is not a kazoo or a plastic recorder! It is a real instrument and can produce incredible music — and for many musicians, it is their lifelong instrument of choice. Likewise, Python is not an overly simplified programming language useful only for beginners (like Scratch and other drag-and-drop languages): it is a real programming language, used by real computer programmers to create real (and really cool!) things. Not convinced? Did you know that YouTube's backend was written mostly in Python? Same with Netflix and Pinterest. Google's and Facebook's backends also use some Python. Python is widely used by scientists and engineers for data analysis and visualization, and the quickly growing field of artificial intelligence and machine learning has adopted Python as its de facto standard. In fact, for two decades, Python has been ranked as one of the most popular and widely used programming languages in the world, and by all indications, this will not change any time soon.

Of course, if you enjoy learning Python in this class, you might want to add more programming languages to your skillset. Starting next year (2024-2025) we will likely offer AP Computer Science A, which will give you the opportunity to learn Java and get deeper experience with object-oriented programming. Rest assured that the skills and thinking patterns you learn in this class will be easily translated to Java and most other programming languages you might decide to learn.

Will I be encouraged to use artificial intelligence tools?

You will not be encouraged to personally use artificial intelligence tools developed by others. However, as part of our classroom discussions on this important topic, demonstrations of certain uses of artificial intelligence will be given in class, including the use of large-language models, like ChatGPT, and certain AI image generators. The purpose of these demonstrations is to facilitate the analysis of what artificial intelligence is, what its current capabilities and limitations are, and what world views and attitudes it reflects. Additionally, you will have the opportunity to employ some simple tools to train a machine learning model; the purpose will be to understand how machine learning works.

Just like computers and the Internet, AI is a powerful tool — and, by nearly all reputable projections, will quickly become much more powerful. As is true of all powerful tools, AI can be used for both good and bad. Under no circumstances should you use ChatGPT or other artificial intelligence tools to complete your assignments for you. Doing so is dishonest and will hurt you in the long run, for it will rob you of the very learning, thinking, writing, and feeling opportunities that you have come to American Heritage to obtain. There are, however, appropriate ways to use AI, which we will discuss as part of this course.

Will I be encouraged to use search engines to find information online?

Yes. The use of technology is generally discouraged during class time at the Salt Lake Campus of American Heritage School, as it often distracts from the opportunity to interact with other people — both your teachers and fellow students. However, technology and the Internet are such huge parts of our society and carry such great potential for good that *in this particular class*, you will not only be encouraged to use these tools, but also taught appropriate ways to use them. At no time, however, will "surfing the Internet" be tolerated. Before any Internet research is performed, we will carefully discuss appropriate Internet use, how to identify reputable sources of information, and which search engines and websites are most likely to provide wholesome, accurate, legal information and images. During class periods in which we conduct Internet research, your computer screen will at all times be visible to me and other class members.

The Internet has become the main repository for official programming documentation. Finding information, examples, and tutorials to teach yourself new programming concepts is a very helpful skill when programming a computer. In the initial stages of this course, I will aim to give you as much information as I can, thus minimizing the amount of Internet searching that must be done. However, during the final project, you will likely need to obtain additional information from the Internet. You will be pointed to good sources, so that in most cases, you will not need to resort to conducting broad Internet searches. But in some cases, conducting an Internet search will be the most efficient way for you to obtain the information you need. If you or your parents are concerned, please contact me. I will be very happy to work with you and to provide alternative activities or resources.

Can I prepare to take the AP® Computer Science Principles test?

Yes — if you are willing to commit some extra time outside of class. In this course, we will cover most of the content required to succeed on the AP® Computer Science Principles exam (please note that this is different from

AP® Computer Science A, which we will begin offering in the 2024-2025 school year, as noted in other places in this syllabus). However, in order to make this course more relevant to all students — especially those students who will not pursue technical careers — we will not cover all required content or do AP exam preparation in class. Instead of spending the final months of the year preparing to take the AP exam, we will devote that valuable class time to creating student projects in students' areas of interest.

If you are interested in taking the AP exam and are willing to spend extra time learning, preparing, and taking the test, I will gladly get you ready for the test. The normal curriculum will get you about 75% of the way there; I will provide extra learning material and practice exams to give you the other 25%. Here is how it will likely look:

1) <u>Select the AP course option</u> — By September 15, you will need to send me an email declaring that you intend to take the AP exam. I will not try to influence you one way or the other; that is a decision that you can make in consultation with your parents and the Holy Ghost. The cost to register for the exam is $98. If you select this option and take the AP exam, your transcript will be changed at the end of the year to show that you took AP® Computer Science Principles.

2) <u>No difference for the first two terms</u> — There will likely be very few (if any) differences for you in the first two terms of our class. You will complete the same course work as all non-AP students.

3) <u>Extra homework problems in the second two terms</u> — Starting in January, you will begin receiving an extra section of problems on your weekly homework. These will help you learn topics not covered in class. These will sometimes be supplemented with videos or articles for you to read.

4) <u>After-school "Create Performance Task" preparation</u> — One of the unique aspects of the AP® Computer Science Principles test is that it requires the completion and submission of a project before the exam day. This project — called the "Create Performance Task" — is a collaborative project that is mostly completed during class time and will make up 30% of your AP score. Twice during the month of February, you will be required to attend extra class meetings after school in which you will be prepared for this project. We will work together to determine the best day(s) of the week to meet after school, so as to minimize conflicts with extra-curricular, family, and church activities.

5) <u>Create Performance Task completion</u> — The College Board (the organization that administers the AP exams) requires that at least 9 hours of class time be provided for students to work on the Create Performance Task. As we will not be taking time for this during our normal class time, we will identify eight 1.5-hour sessions of after-school time during the months of March and April when all AP students will remain after school to work on their Create Performance Tasks. The Task is a collaborative project in which you will create a computer program with a group of other students, so committing to be present for these 9 hours will be very important. Since these 9 hours must be considered "class time," attendance will be taken and a grade will be awarded for the completion of the Create Performance Task. The submission deadline for the Create Performance Task is April 30, 2024, at 11:59 PM Eastern Time (9:59 PM Mountain Time). Please note that you will not be allowed to work on this project during our regularly scheduled class time, and you will still be expected to complete the normally scheduled class projects that will be going on during that time.

6) <u>AP Exam preparation</u> — After you submit your Create Performance Task, you will have two weeks to prepare for the AP exam. During those two weeks, I will pull you and all of the other AP students together once or twice during our regularly scheduled class time to give you some pointed test preparation. You will also be provided with practice exams and other preparation material to complete at home. I will review your scores on these practice exams, help you identify areas where you could improve, and answer any questions you will have. Even though my focus in this class will need to be on our normal class curriculum, I will absolutely be there to help you be as prepared for the exam as you can possibly be.

7) <u>AP Exam</u> — The AP exam will be administered on May 24, 2024, at 12:00 noon Mountain Time. We hope to administer the exam at our school, but we will let you know the exact location as time gets closer.

Programming Units
Unit 1: Basic Python Programs
<u>Discussion Topics</u>
- What are Computers?
- Computers as Tools
- Computers as a God-given Endowment for the Latter-day Work

- Brief History of Computers
- What are Computer Programs?

<u>Python Programming Modules</u>
Basic Printing, Variables, Operators, and User Input
1.0 — Setup Environment, Hello World!
1.1 — Basic Printing and Comments
1.2 — More Printing
1.3 — Numbers and Math Operators
1.4 — More Operators
1.5 — Variables
1.6 — Multiple Lines and Escape Characters
1.7 — Random Values
1.8 — Getting User Input
1.9 — Getting Numerical User Input
1.10 — Printing with Format Strings

<u>Programming Project</u>: Fill-In Story
Students will create a Python program that generates a "fill-in" story (also called a "mad lib" story). The program will begin by asking the user to enter words of different types. These words and several randomly generated numbers will be inserted into a student-written story, which will be printed to the terminal.

Unit 2: Computer Architecture and Decision-Making
<u>Discussion Topics</u>
- Computer Architecture and Function
  - Where are computers found?
  - What are the different types of computers?
- Binary Numbers

<u>Python Programming Modules</u>
Decision-Making
2.1 — If-Else Statements
2.2 — Comparison Conditions and Elif Statements
2.3 — Testing for Inequality and Multiple Conditions
2.4 — Checking User Input
2.5 — The Or and Not Operators
2.6 — Nested Conditionals

<u>Class Activity</u>: Binary Messages
Students will write an encoded message to a classmate by converting English characters into binary bits using an ASCII table and their knowledge of the binary representation of decimal numbers.

<u>Programming Project</u>: Personality Profile
Students will write a Python program that tells the user his or her personality type, based on the answers to five questions about his or her personality. There are 16 different personality types, all of which are different combinations of the five main personality traits. This project will require extensive use of if-else statements.

Unit 3: Iteration and Algorithms
<u>Discussion Topics</u>
- Repetition and Iteration
- Algorithms

- o What are Algorithms?
- o Building Blocks of Algorithms: Sequences, Decisions, Iterations
- o Process of Algorithm Development

Python Programming Modules
Lists and Loops
3.1 — List Creation and Indexing
3.2 — Manipulating Lists
3.3 — List Slicing
3.4 — Looping Over Ranges and Lists
3.5 — While Loops
3.6 — Creating Lists in Loops
3.7 — Combining Decision-Making and Iteration
3.8 — Rounding Numbers and More with Lists

Programming Project: Virtual Waiter
Students will write a Python program that displays a restaurant menu, takes an order from a customer, totals up the price for the order, and then prints out a receipt for the customer. Completing this task will require students to organize menu items and customer orders using lists, to iterate through those lists, to loop an unspecified number of times to allow the customer to order as many items as they want, and to practice the skills of algorithm development that are taught in class.

Class Activity: Max Number Algorithm
Students will work in pairs to design an algorithm that identifies the maximum number in an unsorted list of numbers. They will diagram the steps of the algorithm, specifically identifying the steps, decisions, and iterations involved. They will then team up with another group and compare the two algorithms, considering whether each algorithm will yield the same result and which algorithm will take the least number of steps to complete. Each group will then code their algorithm in Python. When the algorithms are completed, groups will give their code to another group for testing. Students will develop a list of test cases to ensure that the algorithm is working.

Unit 4: Abstraction, Modularity, and Clean Coding
Discussion Topics
- Abstraction and Modularity
- Clean Coding and Refactoring

Python Programming Modules
Functions
4.1 — Defining and Naming a Function
4.2 — Practice Creating Functions
4.3 — Input Arguments
4.4 — Return Values
4.5 — Pass, Modify, and Return Lists

Programming Project: Fact Game
Students will collaborate in groups of two to develop a program that tests a user's knowledge of a certain set of facts (a virtual flash-card game). Students will develop a number of functions to use in their program, including functions that display a question and receive the user's answer, check the user's answer, update the user's score, add questions the user got wrong to another list for further quizzing, and print the user's final score.

Unit 5: Inputs and Outputs
Discussion Topics
- Identifying Inputs and Outputs

- Improving the User Experience

Python Programming Modules
File Operations
5.1 — Reading a File
5.2 — Searching a File
5.3 — Writing to a File
5.4 — Appending to a File

GUIs
5.5 — Creating a Tkinter GUI
5.6 — Using a GUI to Make an Interactive Program
5.7 — User Input in a GUI
5.8 — GUI Geometry Managers

Programming Project: Book Search
Students will write a program that reads in the entire text of a book, searches it for user-specified words or phrases, and then displays the results (including the number of times that word appears in the text). A simple GUI will be developed, which will allow the user to type in their desired search phrase, click a button to initiate the search, see the results on the screen, and click a button to initiate an export of the results to a file.

Unit 6: Final Project
Programming Project: Final Project
Students will each create a final project: a significant Python program that intersects with one of the student's passions or areas of interest and could make a difference for good by positively impacting the student, the student's family, or others. Each student will create his or her own project, but students will be paired with a collaboration partner; over the course of the project, collaboration partners will work together on identifying key steps in the algorithms of the two programs, generating a list of test cases, and solving problems that arise during the development process. Students will acknowledge the intellectual property of others by noting in their code (through comments) and in their final report which portions of code were developed by others. All projects will be demonstrated to the class at the end of the school year, as well as to family members and others during a public open house.

Readings and Class Discussions
The following topics will be discussed in class at different times during the year, interspersed with the Programming Units above.

Topic 1: The Broad Impact of Computing
Discussion Topics
- Reading and Class Discussion: *Blown to Bits*, Chapter 1: Digital Explosion
- Reading and Class Discussion: "'A century of living': In celebration of his 99th birthday, President Nelson reflects on innovation, growth, political breakthroughs" (Church News, 10 September 2023).

Research Project: Impact of Computing Interviews and Essays
Students will conduct two interviews with family members or neighbors who were born before 1971 and thus have witnessed firsthand the transformations brought about by the digital revolution. They will then use what they have learned from class, from their reading in *Blown to Bits*, and from their interviews to write essays responding to the following prompts.

- How has computing changed the world? You may consider the following areas: safety, security, privacy, productivity, convenience, and connectivity. Your essay should address both the good and the bad; state whether you believe the good outweighs the bad, or vice versa; and then support your claim.
- How has computing impacted life for individuals and families? Has it contributed to or detracted from "happy living"?
- What has not changed during the digital revolution? What is the significance of these unchanging things to those you interviewed? And what is the significance of these unchanging things to you?

Homework Activity and Class Discussion: Computing Innovations
At least three times during the year, students will read about a computing innovation as part of their weekly homework. After each reading, students will independently answer the following prompts. Students will then have a chance to share their thoughts and learn from each other in a class discussion.
1. What is the purpose of the computing innovation (what need does it fill or what problem does it solve)?
2. What is the function of the computing innovation (how does it work)?
3. Explain the beneficial and harmful effects of this computing innovation on society, the economy, or culture.
4. Identify the data used in the computing innovation and explain how that data is produced, consumed, and/or transformed by the computing innovation.
5. Identify data privacy, security, and/or storage concerns for this computing innovation.

The three computing innovations that will be discussed in this manner are as follows, along with the readings that students will complete.
- The "Internet of Things" — *Blown to Bits*, "The Internet of Things," pp. 42–46.
- Embedded microphones in Internet-connected devices — *Blown to Bits*, "You Pay for the Mic, We'll Just Listen In," pp. 62–63.
- ChatGPT — "I, Chatbot" by Sara Smith Atwood and ChatGPT, Y Magazine, Spring 2023.

Topic 2: Privacy and Safe Computing

Discussion Topics
- Reading and Class Discussion: *Blown to Bits*, Chapter 2: Naked in the Sunlight
- Class Discussion: What are the advantages and drawbacks of technology with regard to privacy and safety?

Class Activity: Legislative Roleplay
Students will work in groups to discuss potential policies, laws, or Constitutional amendments that could be enacted to solve some of the privacy and safety problems posed by digital technology. Each group will present their ideas to the class and the merits and drawbacks of these ideas will be discussed.

Topic 3: Legal and Ethical Concerns

Discussion Topics
- Reading and Class Discussion: *Blown to Bits*, Chapter 6: Balance Toppled
- Reading and Class Discussion: *Blown to Bits*, Chapter 7: You Can't Say That on the Internet

Research Project: Legal and Ethical Implications of an Innovation
Students will research and write a two-page essay on the legal and ethical implications of one of the innovations mentioned in the *Blown to Bits* readings they have completed.

Topic 4: Computer Systems and Networks

Discussion Topics
- Computer Systems
  - Parallel and Distributed Computing
- Computer Networks and the Internet
  - How the Internet Works

- ○ Fault Tolerance
- ○ Reading and Class Discussion: *Blown to Bits*, Chapter 4: Gatekeepers

<u>Classroom Simulation Activity</u>: Transmitting Messages through the Internet
Students will simulate how information is requested and passed through the Internet by handing index cards with "packets" of information from student to student. These "packets" will contain portions of pictures or texts that needs to be reassembled, so students will have to clearly label each "packet" with identifying information allowing it to get to the right place and be reassembled correctly once it arrives.

Topic 5: Artificial Intelligence and Machine Learning
<u>Discussion Topics</u>
- What is Artificial Intelligence?
- What is Machine Learning?
- Human versus Computer "Thinking"
- Strengths and Weaknesses of AI
- ChatGPT and Artificial Intelligence Today
- Reading and Class Discussion: *Blown to Bits*, Chapter 9: The Next Frontier

<u>Class Project</u>: Train a Machine Learning Model
In class, students will be coached through the process of training a machine learning model in Python.

Topic 6: Using Computer Science for Good
<u>Discussion Topics</u>
- Recognizing Truth in an Age of Artificial Intelligence and Virtual Reality
- Prophetic Counsel on Technology Use
- Using Computer Science to Build the Kingdom of God

Course Culmination
At the end of the year, the following will be done to culminate our efforts in this course.
1. Project Display and Demonstration — We will hold an after-school demonstration, in which you will display a couple of your best class projects, including your final project. This event will be held right after school for approximately half hour and will be open to the public, including your parents and anyone else you would like to invite.
2. Project Portfolio — Throughout the year, you will create reports on your projects. These reports will be assembled in your class notebook to create a portfolio of your work.