



SCHOLÉ ACADEMY
CLASSICAL ACADEMIC PRESS

The Logic of Computer Programming

Fall Semester 2023-2024



ELIGIBLE STUDENTS:

Grades 9th-12th (open to 8th only under required prerequisites): *Designed for high school students, to gain understanding of the logic behind computers and computer programming to develop their own programs and more easily learn any programming language. Discussions will also involve issues surrounding the Christian's use of computer technology from a Biblical, Classical perspective. Students will write two papers to reinforce our class discussions surrounding technology, one reflecting on one's own use of technology and a final that will examine the positives and negatives of current or future technology through an essay or short story. It is recommended that the student has completed Formal Logic or has a sufficient Math background for this course.*

Please note: The student completing this course earns 1/2 high school course credit.

Class Dates: Fall Semester, Tuesday, September 5, 2022; running through Thursday, January 18, 2023.

Class Times: Tuesday & Thursdays: 3:30pm — 4:45pm (EST)

Instructor: Peter Belfry

E-mail: pbelfry.scholeacademy@gmail.com

SCHEDULE FOR *THE LOGIC OF COMPUTER PROGRAMMING*:

CLASS SESSIONS DATES:

Classes will take place on Tuesday & Thursdays: 3:30pm — 4:45am (EST) for 16 weeks and 32 classes on the following dates* --

September (8): 5, 7, 12, 14, 19, 21, 26, 28

October (9): 3, 5, 10, 12, 17, 19, 24, 26, 31

November (7): 3, 7, 9, 14, 16, 28, 30 [November 20-24 Thanksgiving Break]

December (4): 5, 7, 12, 14 [December 18-January 5 Christmas Break]

January (4): 9, 11, 16, 18

**Please note the above dates and times are the anticipated class sessions for this course. Should the instructor be forced to cancel class due to illness or family emergency, every effort will be made to schedule a makeup class at an alternate time.*

THE LOGIC OF COMPUTER PROGRAMMING COURSE MAP:

QUARTER 1

1. How Do Computers Think? How Should We Think About Computers?
2. Thinking in Computer Logic: Pseudocode and Flowcharts
3. Input/Output
4. Variables
5. Working with Variables in Decisions and Repetition
6. End of Quarter Writing Assignment: Reflection on one's own use of technology from a Christian perspective

QUARTER 2

1. Working with Variables in Decisions and Repetition Continued
2. Advanced Input/Output
3. Advanced Programming Techniques: Graphics, Sound, Images, Movement
4. Troubleshooting
5. Introduction to the Logic of Web Programming: HTML (time dependent)
6. End of Quarter Writing Assignment: Essay or Dystopian Short Story reflecting on the benefits and dangers of current and upcoming technological advances from a Christian perspective

*Throughout the course, students will be introduced to new technology, Bible verses, and quotes as the basis for discussion that considers technology from a Christian perspective.

OFFICE HOURS: Tuesdays and Thursdays 10:00-10:30 AM (EST) or upon request.

REQUIRED COURSE TEXTS:

The course text is *A Course in Programming with QBASIC*, which is available from Amazon [here](#).

OPTIONAL COURSE TEXTS: Papers and essays will be submitted using basic MLA formatting guides. The *MLA Handbook for Writers of Research Papers* — 7th Edition may be a helpful resource.

For a great introduction to coding logic in general: *How to Think Like a Coder: Without Even Trying* by Jim Christian, which is available from Amazon [here](#).

For an introduction to Web Programming: *Simple HTML, CSS and JavaScript lessons to get you started with Programming from Scratch* by Bob Mather, which is available from Amazon [here](#).

REQUIRED FREE SOFTWARE: Please download and prepare the following software ahead of the course:

Students should also install QBASIC 64 on their computer, found here:
<https://github.com/QB64Team/qb64/releases/tag/v2.0.2>

If students are unable to install QBASIC 64 on their computer for whatever reason, there are several online options they can use in this case.

<https://replit.com/> can be used for simple programs

<https://www.jdoodle.com/execute-freebasic-online/> is a second option that has strong functionality

<https://codeclubaki.github.io/qbasic/online/> this option has full functionality aside from not being able to save, so code must be saved in a separate file

Students should have Notepad++ for website development at the end of our course, found here: <https://notepad-plus-plus.org/downloads/>

THE LOGIC OF COMPUTER PROGRAMMING COURSE DESCRIPTION:

"Technology is a soaring exercise of the human imagination"

-Daniel Bell

From the beginning of man's creation, God has set it in the human heart to create through technology (Genesis 4:21-22). Technology has the power to be used for good and evil depending on the heart of the man that uses it. In the Scriptures we read of a city of sin being built called Babylon and how technology was used to build a tower for exaltation of self and ascension to heaven, but we also read of a city of God (Genesis 11:3-4, Hebrews 11:10).

There is no doubt that the technology of computers along with a wide-ranging plethora of programs and applications have transformed the way our world communicates and solves everyday problems. Consider for instance our ability to take courses online through Scholé Academy. Like any technology, computer programming can be used for evil or for good, for the glory of man or of God. As Christians we ought to use technology to serve our creator and further His kingdom here on earth.

Most students, while being familiar with the technology of a computer, have not been taught the details of how a computer works or have the ability to use it to create a program of their own. They have not thought philosophically, classically or Biblically about the important cultural implications and decisions that must be made regarding these ever-changing technologies.

In this course, students will step back from their familiar use of technology and with a classical rather than modern approach, learn the logic of computer programming: how computers work and what logical processes they use. As they grasp these realities they will

come to a place where they will begin more easily to think like a computer or think in code, with the ability to create programs they can run for friends and family in a creative way. Along the way, they will be asked to consider important questions about the Christian's use of technology so they can make better, more informed decisions about what they believe and what they will do in the technologically obsessed culture in which we live.

As students learn to understand the logic of how a computer processes and thinks, which will be compared to that of a human (computers, after all, are created by humans), they will develop their own coding logic to solve problems in pseudocode (coding logic in an everyday, easy to understand language) and flowcharts (charts showing coding logic in an easy to understand, visual way). After they plan their code, they will create their program in the language of QBasic. Students will also be introduced to Web Design.

The result will be that students will become more aware of how the computer technology our world uses works and be able to create functional programs their family and friends can use as a result. They will be well equipped to more easily learn any new computer programming language and think in code. They will be able to make Biblical, ethical decisions about what technology they will participate in and how they will do it in a way that honors and serves God. Students will write two papers to reinforce our class discussions surrounding technology, one reflecting on one's own use of technology and a final that will examine the positives and negatives of current or future technology through an essay or short story.

This course will also well prepare students for a second semester course entitled "The Art of Computer Programming." Taking this course will ensure that they are able to move more quickly into developing creative, visually based programs in Visual Basic and Web Programming by more easily grasping the logic of a new, visually based language. These two courses will help students to excel in computers and computer programming should they continue on computer studies or to develop functional computer programs that will help in the career or business of their choice. It will help to develop their creativity and problem-solving abilities in the process.

At Scholé Academy, we have carefully considered how we should engage our contemporary culture as those who believe that Christ is the Truth (John 14:6), and that all truth has its source in him. We think it is important to provide our upper school students (in grades 7-12) with tools and opportunities for critically examining various cultural trends, issues and mores through the lens of orthodox, Christian beliefs. Being confident in the truth revealed to us in creation, the Scriptures, and the tradition of the church, we are not afraid to follow the truth and its implications nor to address error and falsehood. ... Read more about our [Faith & Culture](#).

STUDENT EXPECTATIONS: EXECUTIVE FUNCTION SKILLS

Students enrolling in Scholé Academy's Programming courses will be expected to show development of Executive Function Skills throughout the year. Executive Function Skills speaks to a set of qualities and skill sets students can develop and hone to better approach

the courses, lectures, readings and teachers they will face in their future academic coursework.

Each teacher will invariably have his own set of requirements and skills he requires students to bring to their studies. *Generally speaking*, there are five such qualities that are necessary for my students in various subjects; and I believe they would be accepted as “good” by many other teachers as well.

1. An Engaged Student: One who is willing to step into the arena of class discussion, ask questions, supply answers, generate the internal dialogue necessary to determine if what's being discussed is important and necessary to himself.

2. Note Taking: A student who during and after being engaged with the class has been trained to note important and relevant content in an organized fashion (Cornell Notes would be a great option). His notes would then be consulted, independently, for application in assignments and assessments.

3. Attention to Detail & Preparedness: These students are ones who consistently adhere to deadlines, submission requirements, adhering to style guides and codes, confirm technology is working prior to the start of class, be responsible to determine how to proceed after an absence, be responsible for consulting his course syllabus and adjusting as the class proceeds, etc.

4. Employ Critiques: These students are ones who receive feedback to one of their submissions, and then are sure to apply that feedback to future assignments rather than repeating mistakes. These students also glean information from the live class critiques of fellow students and note mistakes to avoid by learning from others.

5. Initiative/Maturity: This student would hear the teacher comments and be able to assess whether or not the teacher was describing his work, and then take the initiative to schedule office hours with his teacher if necessary.

STUDENT EXPECTATIONS IN ACTION

Students will interact with the textbook to begin to learn the logic of programming and how to use it in code. Live classes will help to explain coding logic in everyday, easy to understand, simple terms. In class, we will build on the textbook by ensuring understanding, looking at examples and sharing some of our own. Students will be challenged to make their own unique creations. Students will also be challenged in live classes to consider relevant issues surrounding technology, engaging in class discussion and debate. Relevant news articles and current events may be discussed and provided to students outside of class for review and in-class discussion. As we move through the course, students will be provided with ideas, concepts, and larger tasks to seek to accomplish with the teacher's guidance and suggestions. Students will be challenged to solve problems

provided by the teacher in a creative way and produce programs that can be used by their friends and family.

The following forms of assessment and learning will be used:

Coding Logic. Students will demonstrate understanding of programming concepts through the use of pseudocode, flowcharts, and solving problems that are provided.

Coding Practice. Students will be asked to solve problems using the logic and language they have learned.

Coding Creations. Students will be asked to solve a larger problem in a creative way, having to pull together the knowledge they have learned. Emphasis will be on visually appealing and creative solutions. Students will be asked to keep comments in their code to help them and others understand what each component does within their program.

Socratic Discussion. Students will be challenged to discuss and debate relevant issues surrounding technology and consider how we can approach these issues Classically and Biblically in a way that honors Christ and furthers His kingdom.

Program Presentations. Students will be asked to either live or with a recording, demonstrate the major programs they develop through screen capture and their own commentary to explain how it works and what it is for.

Writing. Students will be asked to reflect Classically and Biblically about technology and relevant issues surrounding technology in society and how we as Christians ought to respond.

In this class, students will be expected to listen attentively, participate actively in class discussions and practices. Students are expected to arrive to class on time and with all assigned material completed. The instructor will facilitate learning for the student, but the responsibility for staying up-to-date with classwork and assignments ultimately falls to the student.

All assignments will be due into the appropriate Canvas Assignment folder prior to the start of class each day. Students turning in late work will earn a 10% penalty for each day the assignment is late. Students will submit any written work by scanning their homework pages and uploading it into the Canvas assignment window. Photographs of completed assignments will not be accepted as they are incredibly difficult to read.

When developing coding logic in pseudocode, a flowchart, and QBasic or another language, students will be expected to provide comments in their code. Comments are not part of the program, but help to explain what the code does to someone reading it. This is important to show that the student understands what they are doing and it is their own work.

For programs the students create, they can be submitted with either a functional code file

or with the code in a text document along with screenshots of the student running the program. For bigger projects, the student will be asked to record a video or do a live demonstration.

STUDENT EVALUATION: GRADING

While pursuing *The Logic of Computer Programming* through Scholé Academy will be “restful” (I’d also like to say it’s going to be a lot of fun), we also recognize the need to provide grades for students who will be using this course as part of their prepared college transcript. It’s a delicate balance to achieve both restful learning and excellent academic performance. Earning a specific grade should not overshadow achievement goals for mastery of this discipline which will well prepare you for the study of other programming languages in the future or to be able to develop programs that are useful for your future career or business. As the teacher I can assign the following grades to your student’s level of achievement: *magna cum laude* (with great praise); *cum laude* (with praise); *satis* (sufficient, satisfactory) and *non satis* (not sufficient).

Ideally, every average student working diligently should do praiseworthy work (*cum laude*). Those who excel beyond this expectation will be the *magna cum laude* students. Students who do adequate but not praiseworthy work are designated *satis*. *Non satis* means lacking sufficiency or adequacy.

Inasmuch as you might be fully on board with this grading method in theory, there will undoubtedly be the need to complete a college transcript with either a numeric or traditional letter grade. Traditional percentage grades will be provided and will be readily accessed on *The Logic of Computer Programming* Canvas page. Additionally, Mr. Belfry will provide a transcript of that grade to the requesting parent at the end of the year.

STUDENT EVALUATION: MASTERY PORTRAIT

Mastery portrait: Students who are prepared to take this class are typically early to late teens, adolescents approaching young-adulthood. This developmental stage is an interesting one, brimming with lots of new characteristics. It’s imperative, then, that this course not only provide the academic components necessary to achieve mastery of the content of the class (knowledge) and skills associated with analytical thought; but to also help engage the student in development of their moral virtues. These three aspects of the course would comprise the “learning target”.

- At the completion of this course *cum laude* students will be able to name, define and use the programming logic studied in the course with pseudocode, flowcharts, and functional programs.
- Additionally, they will have attained the skills necessary to use the programming logic they learn to develop their own creative programs that solve problems and interest the user.
- Students will also be guided in development of the virtues of Truth, Goodness and Wisdom. They will learn to discern how technology can be used in a way that glorifies God. They will learn to consider their own beliefs about what technologies are able to be used in a moral way and which should be rejected by the Christian.

STUDENT EVALUATION: ASSIGNMENTS, TYPES & WEIGHTS

Mr. Belfry will communicate with students regarding assignment feedback and grading through the free online grading system, Canvas. The teacher will provide students with more detailed information and access to the Course Title course page.

Student's grades will be comprised of:

Coding Logic, Art, and Practice (Homework) 15%

Coding Creations (Assignments) 35%

Program Presentations (Projects) 30%

Writing (Based on Class Discussions) 20%

STUDENT EVALUATION: ACADEMIC DISHONESTY

Students will often take assessment tests and/or quizzes privately at home. Students are on their honor to abide by Scholé Academy's Learning Philosophy which assumes the personal cultivation of Student-Virtues described in the Student-Parent Handbook.

Additionally, plagiarism is a serious and punishable offense. Proper citation of all sources is essential to the academic endeavor. Remember to cite any source if the information is not common knowledge or is an opinion obtained through any source. A plagiarized assignment will result in a failing grade. Students should consult their chosen style manual (see Student Expectations above) for specific direction on obtaining, quoting and paraphrasing sources. In computer programming, comments are essential to show it is your own work by explaining what each part of the code does.

THE VIRTUAL CLASSROOM:

We will be using the free online "virtual classroom" software provided by Zoom, one of the leading companies that provides such software. The virtual classroom will provide students with interactive audio, text chat and an interactive whiteboard in which texts, diagrams, video and other media can be displayed and analyzed. We will provide students with a link (via email) that will enable students to join the virtual classroom.

Specific information regarding the technology used by Scholé Academy (including required technology) can be found by visiting the Technology in the Classroom section of the Student Parent Handbook.

Students will submit documents by scanning and uploading them to their personal computer, then attaching those files as .pdfs to an email. They will submit their work to the *Art of Argument* Canvas assignment page (access granted after enrollment is secured).

ABOUT THE INSTRUCTOR:

Peter Belfry has a range of teaching and tutoring experience in a variety of subjects and age levels from kindergarten through to adult education at the college level and has taught at several classical, Christian and public schools. Peter serves as a professor of Computer Science with Canadore College, teaching courses on Operating Systems and programming languages such as Windows, Linux, HTML, CSS, C++, C#, and Visual Basic. Peter holds an Honors BA from Trent University in History as well as a BA in Education, specializing in History and Computer Science. He has taught many computer courses to grades 7-12 students and has developed an interactive, digital programming textbook for students. He loves how programming allows creativity to flourish by using both the left and right sides of the brain, being both creative and logical in nature. He holds an MA from Knox Theological Seminary in Classical and Christian studies, which provides him a background for teaching from a classical perspective. For his MA program, he read and reflected on many of the Great Books as well as studied Scripture and church history. Peter has completed a week-long teacher training with the Association of Classical Christian Schools and Rockbridge Academy. His favorite piece of classical literature is Dante's *The Divine Comedy*.

In addition to teaching, Peter also has experience serving in a pastoral role and enjoys volunteering to serve in his local church and community. He helps in evangelistic outreach as well as teaching lessons from the Bible. Peter has experience and training as an English as a Second Language instructor as well. He has experience teaching both online and in person. He believes in Scholé's approach in seeking "restful learning" and believes that education should be life-giving and freeing for the soul as it should acknowledge the Lord Jesus as the source of all that is true, good and beautiful. Peter lives in the North Bay, Ontario area with his wife and twin boys.

Peter provides tutoring services with Scholé Academy and teaches the following classes: *The Art of Argument: An Introduction to the Informal Fallacies*, *Formal Logic: The Discovery of Deduction*, *The Logic of Computer Programming*, and *the Art of Computer Programming*.

