



## Michael Trunk

---

### **Background:**

Computer programming is a foundational skill for many career fields and passion projects. From engineering to art, computer programs are important tools in everyday life for creating and sharing content. C++ is a human-to-computer language which can be used to write and customize these computer programs. Created in 1983, the language has grown and matured into a capable, modern language. Acquisition of programming skills are tied to real-world wage increases and job opportunities.

### **Objectives:**

As an introductory course, the group will learn the basics of C++ and Computer Science. No previous experience is required—all students from non-CSE majors are welcome! First, the group will focus on elementary syntax of the language. Then, the group will focus on higher-level conceptual design principles and techniques. Finally, each group member will be given time to develop their own creative project. After finishing the course, group members will have these essential skills and experiences to take with them into their future careers and passion projects.

### **Hardware and Software:**

Each participant is required to bring their own laptop. Both Windows and Mac computers are acceptable. For software, the group will use CodeLite, a free lightweight code editor. Additional free C++ compilers may be downloaded separately.

### **Assignments:**

Each week, participants will complete programming-related tasks, including readings, actual coding, and more. These tasks may range from simple to complex, depending on the group's progress through the course. The instructor will record the presence and quality of the completed tasks for each participant.

Completed assignments must be brought in during instruction time, unless one cannot attend in which case the assignment needs to be emailed to me beforehand.

Students who have concerns or run into problems completing the tasks can contact me for information and help! I am passionate about teaching this so I will tailor my responses and will even discuss techniques not taught during instruction time so that the problem can be solved.

**Attendance:**

Participation is necessary and serves to benefit the learner. I will record attendance each week for points. One must clear absences or tardiness with me in advance.

**Policies:**

Participants shall treat peers with respect and sustain a welcoming environment. One should not be a distraction to those trying to learn. For any concerns or questions, bring attention to the instructor or office coordinator—we will happily manage any problems.

**Schedule:**

This schedule is tentative and will depend on the group’s rate of progress through the material. Too, participants may have interests in other ideas or tasks in which case the group could divert time to learn about them.

<b>Week</b>	<b>Topic</b>	<b>Homework</b>
1	Syllabus – Setup CodeLite – Hello World	Draw A Flow Chart
2	Fundamentals – Main - Conditionals	Do Some Math!
3	Creating Functions – Scope – File Management	Calculate Integrals
4	Reference Types - Arrays	Statistics Calculator
5	Enumerators - Structs	Build A Bitboard
6	Classes	Build a Chess Board
7	Object-Oriented Programming Principles	Reading
8	Inlining – Const Correctness - Interfaces	Re-design Chess Board
9	The Preprocessor – Code Documentation	Document Chess Board
10	Catch-up-day – Discuss Final Project	Plan Final Project
11	Start Final Project	Work on Final Project
12	Work on Final Project	Work on Final Project
13	Work on Final Project	Finish Final Project
14	Share Projects – Real World	

**Contact:**

Michael Trunk      [trunk.6@osu.edu](mailto:trunk.6@osu.edu)      937-572-2243

Contact me with questions about anything and I will answer. I can even set up appointments to discuss assignments and projects.