

Foundations of Computer Science and Augmented Reality & Virtual Reality

18-month Program



Become a Software Engineer - For real

Our intensive program will first introduce you to the foundations of computer science and software engineering, then specialize in what drives you.

The first three sprints of our program covers the foundations of computer science and software engineering, including Linux, data structures, algorithms, low-level programming languages, high-level modern languages, databases, APIs, and DevOps.

Then, the last sprint you choose the specialization that is right for you:

- Back-end Web Development
- Front-end Web Development
- Devops



What to Expect

1) No pre-course

Holberton does not expect students to come in with previous software engineering experience (although if you do have experience, that's awesome too).

There is no **pre-course work** (that's why you are attending a school after all), but we do recommend that you read through *The C Programming Language* book by Kernighan and Ritchie or *Programming in C* by Stephen Kochan.

The goal of reading through the book is not to deeply understand all the concepts, but to familiarize yourself with key terminology and content.

2) Coursework

We are training you to be a full-stack software engineers in 12 months. The program will be intense.

There are no formal teachers or formal lectures. Students are learning by creating and we rely on peer-learning, collaboration, and industry-relevant curriculum to guide the way.

There is no competition here at Holberton, rather students are helping each other towards their goals. Of course, there is also technical staff available to answer questions and extend support.

3) Professional development

We know that the skills to get a job are different from the skills to be good at a job. From week zero, we immerse students in professional growth and development via workshops, projects, meetups, and work simulations.

Whiteboarding, mock interviews, professional networking, and more begin as soon as students start the program so that they're confident and competent when the time comes to prove they're ready for the job.

4) Soft Skills

In today's tech world, it's not enough to be good at technical skills, you need to be a clear communicator as well.

We push our students to work on their public speaking skills, to publish blog posts to online tech communities and publications, and to speak at conferences and meetups.

This not only prepares students to be team players and clear communicators, but creates amazing networking opportunities.

5) Included in All Holberton Sprints



Technical writing

Technical writing is an invaluable skill and an excellent way to articulate and share your knowledge.



Collaboration

Collaboration is key to successful business. You will learn project management, interpersonal communication, and team collaboration skills.



The Framework

The Framework provides the structure, order, and balance necessary to maintain a productive peer learning environment and will help you succeed throughout your career.



Whiteboarding

Whiteboarding is an essential skill in the tech industry, both for effective planning and for excelling in tech interviews.



Mock Interviews

It is not enough for you to know the answers to the questions; you need to be able to clearly communicate your thought processes and understanding.

What You'll Learn

Foundation of Software and Engineering

This foundational knowledge of how computers and programming languages work will allow you to optimize and debug anything later on in your professional career. You will also begin working with algorithms and data structures which are essential foundations for great Software Engineers - the type that the best companies hire.

In the first sprint of foundations, you'll work in C and Unix programming, graphical programming, data structures, assembly language, and algorithms as well as reverse engineering and security protocols.

From there, you are introduced to higher-level languages, increasingly advanced algorithms, space and time complexity, database management, and front-end programming. Using the latest technologies, you will begin to create a complete web application project that will span the rest of the foundation sprints.

The final sprint of foundations emphasizes automation, scalability, and reliability, so that you are familiar with the infrastructure and best practices similar to those in tech powerhouses. Alongside a continuation in web development, you'll also advance in algorithmic understanding, technical writing, debugging, and project management.

Examples of Projects

- Write your own printf function
- Web stack debugging
- Build a video game
- Clone a marketplace
- Code your own shell

Curriculum Foundation of Computer Science & Software Engineering

01

1st Sprint

- Git and command line editors
- Introduction to Bash
- C - first statements
- C - pointers
- C - recursion
- C - static library
- C - memory allocation
- C - preprocessor
- C - variadic functions
- C - bit manipulation
- C - file I/O
- Singly linked lists
- Create your own printf
- Create your own basic Shell

03

3rd Sprint

- Python - Object-relational mapping
- Python - Web framework
- Python - RESTful API
- Python - web scraping
- Javascript - first statements
- Javascript - objects
- Javascript - scopes and closures
- Javascript - web scraping
- Search algorithms
- SSH
- SSL certificate
- Web server
- Load balancer
- Firewall
- MySQL primary-replica
- Server monitoring
- Code deployment
- Postmortem
- Webstak debugging
- Portfolio project

02

2nd Sprint

- Python - first statements
- Python - import and modules
- Python - data structures
- Python - exceptions
- Python - classes
- Python - inheritance
- Python - file I/O
- Python - JSON serialization/deserialization
- HTML/CSS introduction
- SQL - basic queries
- SQL - join queries
- C - dynamic libraries
- C - makefiles
- Doubly linked lists
- Stack and Queues
- Hash tables
- Sorting algorithms
- Binary trees
- Bash - scripting
- Unix processes and signals
- Regex
- Network introduction

Specialization



Back-end Web Development

Breathe life into the Web

Back-end web development is where the magic happens on websites. Finding the perfect rental, sharing a photo from the cloud, and keeping people secure while using the web are all driven by back-end web developers.

If you like building frameworks, working on complex projects, and the idea of making solutions that can help millions of people, Back-End Web Development might be for you.

The first three sprints of our on-site intensive education covers the foundations of computer science and software engineering. Students will explore and learn practical low-level programming, high-level programming, algorithms, databases, system engineering, and networking technologies.

For the last sprint, students will then focus on key back-end concepts and technologies. This includes languages like Python and Javascript, and concepts like API pagination, caching algorithms, testing, authentication mechanisms, and background jobs.

Students will learn how to architect and develop for platforms that are secure, optimized, stable and scalable.

Examples of Projects

- MySQL performance debugging
- Cache from scratch
- Authentication service
- Background jobs system
- Yellow pages in GraphQL
- Thumbnails on-demand

Curriculum Back-end Web Development

04

4th Sprint

- ES6 introduction / promise
- ES6 classes / data manipulation
- TypeScript
- Python
- async
- MySQL advanced
- NoSQL introduction
- Redis introduction
- API Pagination
- Caching algorithms
- Unit tests and integration tests
- i18n
- Personal data
- User
- authentications
- Node JS introduction
- Queuing system
- GraphQL API
- Async file API

Foundation

Graduate



Front-end Web Development

Make Amazing Web Experiences

Front-end web development defines how we use our computers every day. The work of front-end developers is what helps technology be usable to the average person, and companies with the best, most stable, and most usable sites are the most popular on the web.

If you want to make the web more usable, more accessible, and more fun to use, Front-End Web Development is the path for you.

The first three sprints of our on-site intensive education covers the foundations of computer science and software engineering. Students will explore and learn practical low-level programming, high-level programming, algorithms, databases, system engineering, and networking technologies.

The final sprint of the curriculum will build upon this knowledge and focus on the skills that make successful front-end web developers. Students will study and master technologies like HTML, CSS, JavaScript, and developer tools and apply these technologies with React, one of the most popular front-end frameworks.

For Holberton, a front-end developer is not only someone who can code with a framework, but someone that understands why these frameworks exist, work and how they interact with the browser (like Chrome, Edge, Safari, etc.)

Examples of Projects

- Desktop and mobile version of a product website
- Student dashboard in React
- Countries portal with React and GraphQL
- CRM dashboard in React

Curriculum Front-end Web Development

04

4th Sprint

- ES6 introduction / promise
- ES6 classes / data manipulation
- TypeScript
- HTML / CSS advanced
- Developer tools
- Responsive design
- Webpack
- React introduction / props
- React component
- React inline-styling
- React state / immutable
- React Redux - action creator/normalizr
- React Redux - reducer/selector
- React Redux - connector/provider

Foundation

Graduate



Devops

Reliability at Scale

The first three sprints of our on-site intensive education covers the foundations of computer science and software engineering. Students will explore and learn practical low-level programming, high-level programming, algorithms, databases, system engineering, and networking technologies.

Throughout this last sprint, students will extend their knowledge of system engineering and DevOps, digging deeper into the intricacies of the cloud platform such as AWS, monitoring softwares, CI/CD tools, and more.

Students will deepen their understanding of how to turn code into an agile, safe, templaterize and reliable live, production-ready product.

Examples of Projects

- Migrate an application to the AWS cloud
- Set up monitoring on a server
- Creating a CI/CD pipeline using Travis CI
- Break an application into microservices
- Dockerize an entire web application

Curriculum DevOps

04

4th Sprint

- Cloud Migration with Terraform and Packer
- AWS
 - EC2
 - Elastic Load Balancers
 - Aurora/Relational Database Service
- TLS/SSL Certificates
- Configuration Management with Puppet
- CI/CD with Travis CI
- Monitoring
 - Graphite, Statsd, Grafana
 - Service-Level Objectives
 - Service-Level Agreements
 - Service-Level Indicators
- Docker
- Alerts
- Logging
- Database Drivers

Foundation

Graduate

The application Process

Our selection process is based only on talent and motivation. We don't care what degrees you may or may not have, if you have any previous programming experience, or your ability to pay. If you possess curiosity, determination, and drive to succeed, then we want you as a Holberton student.

Our automated admissions process aims to remove human biases. It was created specifically to identify smart, motivated people and doesn't take into account previous education, work experience, gender, ethnicity, or age. There's also no cost to apply. — the only requirements are you must be 18 years old and have a GED or high school diploma.

Start Today With \$0 Upfront

At Holberton, we believe that people from every community and background should have the opportunity to become a software engineer. That is why we offer an Income Share Agreement option to help make your new career more accessible.

Learn more



Contact us

Connect with your local campus. Check out our [Locations](#) page to find contact information and explore events, workshops, and networking opportunities in your city.