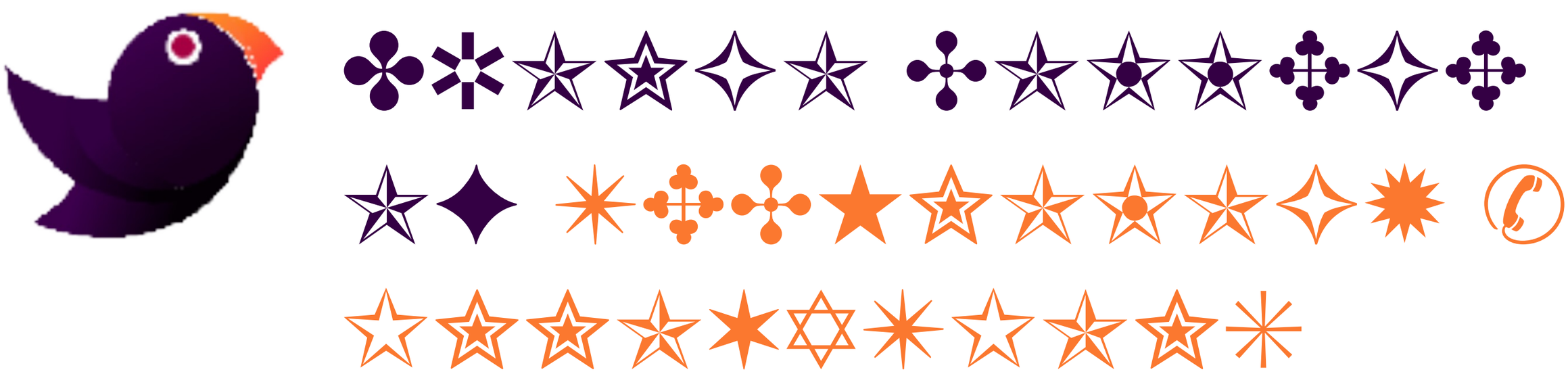


EXTENSIVE COMPUTER SCIENCE PROGRAM

What you'll learn

- A broad and robust understanding of computer science and programming

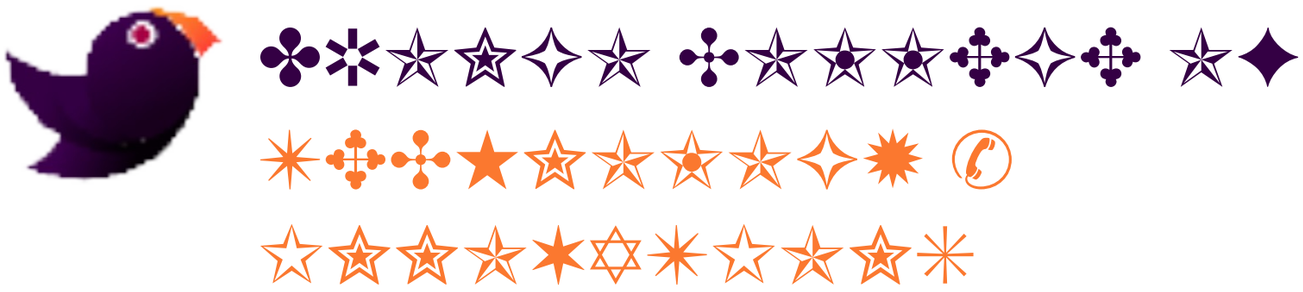
- How to think algorithmically and solve programming problems efficiently

- Concepts like abstraction, algorithms, data structures, encapsulation, resources management, security, software engineering and web development.

- Familiarity with a number of languages, including C, Python, SQL, and Javascript plus CSS and HTML

- How to engage with a vibrant community of like-minded learners from all levels of experience

- How to develop and present a final programming project to your peers



Course description

This is introduction to the intellectual enterprises of computer science and the art of programming for majors and non-majors alike, with or without prior programming experience. An entry-level course teaches students how to think algorithmically and solve problems efficiently. Topics include abstraction, algorithms, data structures, encapsulation, resource management, security, software engineering, and web development. Languages include C, Python, SQL, and JavaScript plus CSS and HTML. Problem sets inspired by real-world domains of biology, cryptography, finance, forensics, and gaming.

Students who earn a satisfactory score of 80% on all problem sets (i.e., programming assignments) and a final project are eligible for a certificate and a guaranteed internship or at best a full-time job.

FIRST SEMESTER

PART ONE - PROGRAMMING FUNDAMENTALS

1. Week one

    1.1 - Day one - Some fundamentals

        - Programming
        - Higher-Level Languages
        - Operating Systems
        - Compiling Programs
        - Integrated Development Environments
        - Language interpreters
        - Compiling & Running Your First Program
        - Exercise

    1.2 - Day two - Variables, Data Types and Arithmetic Expressions

        - Understanding Data Types and Constants
        - Working with Variables
        - Working with Arithmetic Expressions
        - The Assignment Operator
        - Triangular numbers
        - The for Statement
        - The while Statement
        - The do Statement
        - Exercise

    1.3 - Day three - Making decisions

        - The if Statement
        - Th switch Statement
        - Boolean Variables
        - The Conditional Operator
        - Exercise

2. Week two…continued

2.5 - Day five - Milestone project

- Project definition
- Project evaluation

3. Week three

3.1- Day one - Working with Structures

- The Basics of Structures
- A Structure for Storing the Date
- Functions and Structures
- Initializing Structures
- Arrays of Structures
- Exercise

3.2 - Day two - Working with Structures

- Structures Containing Structures
- Structures Containing Arrays
- Structure Variants
- Exercise

3.3 - Day three - Character Strings

- Revisiting the Basic s of Strings
- Arrays of Characters
- Variable-Length Character Strings
- Escape Characters
- More on Constant Strings
- Character Strings, Structures, and Arrays
- Character Operations
- Exercise

4. Week four…continued

4.3 - Day three - Data Types & Data Type Conversions

- Enumerated Data Types
- The typedef Statement
- Data Type Conversion
- Exercise

4.4 - Day four - Working with Larger Programs

- Dividing Your Program into Multiple Files
- Communication Between Modules
- Other Utilities for Working with Larger Programs
- Exercise

4.5 - Day five - Input and Output Operations in C

- Character I/O: getchar() and putchar()
- Formatted I/O: printf() and scanf()
- Input and Output Operations with Files
- Special Functions for Working with Files
- Exercise

5. Week five

5.1 - Day one - Miscellaneous and Advanced Features

- Miscellaneous Language Statement
- Working with Unions
- The Comma Operator
- Type Qualifiers
- Command-line Arguments
- Dynamic Memory Allocation
- Exercise

5. Week five…continued

5.2 - Day two - Debugging Programs

- Debugging with the Preprocessor
- Debugging Programs with gdb
- Exercise

5.3 - Day three - Project & milestone evaluation

- Project definition & preparation

5.4 - Day four - Project & milestone evaluation

- Project tasks
- Project assistance

5.5 - Day five - Project & milestone evaluation

- Project evaluation
- Student and Teacher feedback

PART TWO - DATA STRUCTURE & ALGORITHMS IN PYTHON

6. Week six

6.1 - Day one - Python Primer

- The Python Interpreter
- Preview of a Python Program
- Objects in python
- Expressions, Operators and Precedence
- Control Flow
- Functions
- Exercise

6.2 - Day two - Python Primer

- Simple Input and Output
- Exception Handling
- Iterators and Generators
- Additional Python Conveniences
- Scopes and Namespaces
- Modules and the Import Statement
- Exercise

6.3 - Day three - Object Oriented Programming

- Goals, Principles and Patterns
- Software Development
- Class Definations
- Inheritance
- Namespaces & Object Orientation
- Shallow and Deep Copying
- Exercise

6. Week six

6.4 Day four - Algorithm Analysis

- Experimental Studies
- Comparing Growth Rates - Seven Functions
- Asymptotic Analysis
- Simple Justification Technique
- Exercise

6.5 Day five - Recursion

- Illustrative Example
- Analyzing Recursive Algorithms
- Recursion Run Amok
- Further Examples of Recursion
- Designing Recursive Algorithms
- Eliminating Tail Recursion
- Exercise

7. Week seven

7.1 Day one - Array Based Sequences

- Python’s Sequence Type
- Low-Level Arrays
- Dynamic Arrays and Amortization
- Efficiency of Python’s Sequence Types
- Using Array-Based Sequences
- Multidimensional Data Sets
- Exercise

7. Week Seven

7.2 Day two - Stacks, Queues and Deques

- Stacks
- Queues
- Double-Ended Queues
- Exercise

7.3 Day three - Linked Lists

- Singly Linked Lists
- Circularly Linked Lists
- Doubly Linked Lists
- The Positional List ADT
- Sorting a Positional List
- Case Study - Maintaining Access Frequencies
- Link-Based vs Array-Based Sequences
- Exerice

7.4 Day four - Trees

- General Trees
- Binary Trees
- Implementing Trees
- Case Study: An Expression Tree
- Exercise

7.5 Day five - Priority Queues

- The Priority Queue Abstract Data Type
- Implementing a Priority Queue
- Heaps
- Sorting with a Priority Queue
- Adaptable Priority Queue
- Exercise

8. Week Eight

8.1 Day one - Maps, Hash Tables, and Skip Lists

- Maps and Dictionaries
- Hash Tables
- Sorted Maps
- Skip Lists
- Sets, Multisets, and Multimaps
- Exercise

8.2 Day two - Search Trees

- Binary Search Trees
- Balanced Search Trees
- AVL Trees
- Exercise

8.3 Day three - Search Trees

- Splay Trees
- (2,4) Trees
- Red-Black Trees
- Exercise

8.4 Day four - Sorting and Selection

- Merge Sort
- Python’s built in Sorting Algorithms
- Studying Sorting Through an Algorithmic Lens
- Exercise

8.5 Day five - Sorting and Selection

- Quick Sort
- Selection
- Comparing Sorting Algorithms
- Exercise

10. Week Ten

10.1 - Day one - Project

10.2 - Day two - Project

10.3 - Day three - project

10.4 - Day four - project

10.5 - Day five - project

11. Week Eleven

11.1 - Day one - Project

11.2 - Day two - Project

11.3 - Day three - project

11.4 - Day four - project

11.5 - Day five - project

PART THREE - WEB DEVELOPMENT

12. Week Twelve

    12.1 - Day one - Introduction to HTML

        - Anatomy of an HTML Tag

        - The HTML boilerplate

        - Text structure in HTML

        - HTML Image Elements

        - HTML Links and Anchor

        - HTML Tables

        - Challenge

    12.2 - Day two - Intermediate HTML & CSS

        - HTML forms

        - Inline CSS

        - Internal & External CSS

        - CSS Selectors, Classes and Ids

        - CSS Selectors, Classes and Ids

        - Favicons, Divs, CSS display

        - Static, Relative and Absolute positioning

        - Centering Elements, Font Styling and CSS sizing

        - Challenge

    12.3 - Day three - Introduction to Bootstrap

        - Introduction and Installing Bootstrap

        - Wireframing

        - Navigation Bar

        - Grid Layout

        - Bootstrap Containers

        - Bootstrap Buttons & Front Awesome

        - Bootstrap Buttons & Front Awesome

        - Challenge

12. Week Twelve … Cont…

12.4 Day four - Intermediate Bootstrap

- Bootstrap Carousel
- Bootstrap Cards
- The CSS Z-Index and Stacking Order
- Media Query Breakpoints
- Selector Priority
- Combining Selectors
- Challenge

12.5 Day five - Web Design

- Introduction to Web Design
- Understanding Color Theory
- Manage Attention with effective User Interface (UI) Design
- Project

13. Week thirteen

13.1 Day one - Introduction to Javascript

- Introduction to JavaScript
- Javascript Alerts
- Data Types, Variables & Naming Conventions
- String Manipulations
- JS Arithmetics
- Functions
- Challenge

13.2 Day two -Intermediate Javascript

- Random number Generation
- Control Statements
- Comparators
- Javascript Arrays
- Challenge

13.3 Day three - Document Object Model (DOM)

- Introduction to DOM
- Selecting Elements with Javascript
- DOM manipulation
- Separation of Concerns: Structure vs Style vs Behavior
- Challenge

13.4 Day four - Advanced JS and jQuery

- Higher Order Function and Passing Functions as Arguments
- Javascript Objects - Deeper understanding
- Introduction to jQuery
- Selecting Elements with jQuery
- Manipulating styles, Text and Attributes with jQuery
- Event Listeners with jQuery
- Animations with jQuery

13. Week thirteen … cont…

13.5 Day five - Project

- The Dicee Game
- The Simon Game

14. Week fourteen

14.1 Day one - Vue Javascript

- Introduction to Vue.js
- Creating and Connecting Vue App Instances
- Interpolation & Data Binding
- Understanding “Methods” in Vue Apps
- Working with Data inside of a Vue App
- Raw HTML Content with v-html
- Understanding Event Binding
- Working with Event Arguments
- Using the Native Event Object
- Classes & Computed Properties
- Challenge

14.2 Day two - Rending Conditional Contents & Lists

- v-if, v-else, v-else-if
- Using v-show instead of v-if
- Rending Lists of Data
- Challenge - The Monster Slayer Game

14.3 Day three - Introducing Components

- Introducing Components
- Component Communication
- Global vs Local Components
- Challenge

14. Week fourteen…cont…

14.4 Day four - Forms

- v-model & Inputs
- v-model & Dropdowns
- Using v-model with Checkboxes and Radiobuttons
- Basic Form Validation
- Custom Control Components
- Raw HTML Content with v-html
- Understanding Event Binding
- Working with Event Arguments
- Challenge

14.5 Day five - Sending Http Requests

- Sending a POST Request
- Getting Data (GET Request)
- Handling Error Responses
- Challenge

15. Week fifteen

15.1 Day one - Database

- CREATE and INSERT Data
- SQL Commands: READ, SELECT and WHERE
- Updating Single Values and Adding Columns in SQL
- SQL Commands: DELETE
- Challenge

15.2 Day two - Database

- Indexing Tables
- Metadata with SHOW and INFORMATION_SCHEMA
- Challenge

15.3 Day three - Database

- The Inner Join
- Left and Right (Outer) Joins
- Challenge

15.4 Day four - Database

- Subqueries with Relative Comparison Operators
- IN and NOT IN Subqueries
- ALL, ANY, and SOME subqueries

15.5 Day five - Database

- Project & Challenges

PART FOUR: API DEVELOPMENT WITH FASTAPI

16. Week sixteen

16.1 Day one - RESTful API with FastAPI

- First endpoint and running it
- Path parameters
- Query parameters
- The request body
- Form data and file uploads
- Headers and Cookies
- The request object
- Challenge

16.2 Day two - Customizing the response

- Path operation parameters
- The response parameter
- Raising HTTP errors
- Building a custom response
- Structuring a bigger project with multiple routers
- Challenge

16.3 Day three - Managing Pydantic Data Models in FastAPI

- Standard field types
- Optional fields and default values
- Validating email addresses and URLs with Pydantic types
- Creating model variations with class inheritance
- Challenge

17. Week seventeen…cont…

17.4 Day four - Authentication and Security in FastAPI

- Security dependencies in FastAPI
- Storing a user and their password securely in a database
- Retrieve a user and generating an access token
- Securing endpoints with access tokens
- Configure CORS and Protecting against CSRF attacks
- Challenge

17.5 Day five - Project

- Project

PART FIVE: BUILD DATA SCIENCE API WITH FASTAPI

19. Week nineteen

    19.1 Day one - Introduction to NumPy

        - Getting started with NumPy

        - Manipulating array with NumPy

        - Challenge

    19.2 Day two - Introduction to Pandas

        - Using pandas Series for one-dimensional data

        - Using pandas DataFrames for multi-dimensional data

        - Importing and exporting CSV data

        - Challenge

    19.3 Day three - Introduction to Machine Learning

        - Supervised versus unsupervised learning

        - Model validation

        - basic of scikit-learn

        - Challenge

    19.4 Day four - Classifying data with Naive Bayes models

        - Intuition

        - Classifying data with Gaussian Naive Bayes

        - Classifying data with Multinomial Naive Bayes

        - Challenge

    19.5 Day five - Classifying data with support vector machines

        - Intuition

        - Using SVM in scikit-learn

        - Finding the best parameters

        - Challenge