

Response 200 OK



```
def func() {}
```

# E.O.S

## <INITIATIVE>

```
let data = [];
```

Inspiring girls to code the future

```
return True;
```

# MENTORED PROJECTS BOOKLET





# BINARY EXPLOITATION

MENTOR: YU FANG

## DESCRIPTION:

- This project will introduce the fundamentals of binary exploitation (“pwn”), focusing on how programs work at a low level and how vulnerabilities can be identified and exploited. You will be first guided through simple C programming, the structure of memory, and common vulnerabilities like buffer overflows.

The deliverable by the end of the project will be flexible, but likely a small portfolio including learning notes and challenge write-ups, and a self-made simple CTF challenge (with source code and compiled binary), along with a clear explanation and final presentation.

## SOFTWARE NEEDED

- Have admin rights to their laptop, install Kali VM

## WHO SHOULD TAKE THIS?

- Curious about how programs run on a lower level
- Interested in cybersecurity, or ethical hacking
- Enjoy problem-solving, debugging, and puzzle-like challenges



# COMPUTER VISION

MENTOR: YU FANG

## DESCRIPTION:

- This project introduces the fundamentals of artificial intelligence and deep learning using Python. You will learn how neural networks work, including how layers, weights, and activations allow computers to “learn” from data.
- It includes Convolutional Neural Networks (CNNs), understanding convolutions, filters, pooling, and how models extract features from images, as well as Vision Transformers (ViT), and how attention mechanisms and self-attention enable models to focus on important parts of the data. It also covers data handling and preprocessing, model training and evaluation, techniques to prevent overfitting, and practical skills for building and testing AI models.

Final deliverable is implementing these architectures and training them on datasets of your choice.

## SOFTWARE NEEDED

- Kaggle account

## WHO SHOULD TAKE THIS?

- Curious about how AI and neural networks
- Interested in computer vision, deep learning



# COMPETITIVE PROGRAMMING

MENTOR: KAI QIN

## DESCRIPTION:

- This project provides an introduction to competitive programming techniques, including data structures and algorithms as well as problem solving strategies.
- You will first learn the syntax of C++, the most widely used programming language in competitive programming, before learning the applications of data structures in the C++ Standard Library such as sets, maps and deques, problem solving concepts such as Big O Notation and greedy, as well as classic programming techniques like BSTA, precomputation with prefix sums, and sliding window.

Final deliverable will be a collection of library code for classic techniques, as well as solutions to a few problems and possibly a self proposed problem with a solution explanation.

## SOFTWARE NEEDED

- nil

## WHO SHOULD TAKE THIS?

- Enjoy puzzles, problem solving and logical reasoning
- Want to learn ways of thinking applicable in problem solving beyond a pure programming context



# BASIC APP DEVELOPMENT

MENTOR: CHRISTABEL

## **DESCRIPTION:**

- This project introduces basic front-end and back-end development for apps through simple block coding.
- You'll be introduced to MIT App Inventor where you'll learn how to design UI/UX for users and code the back-end while implementing basic coding concepts and the modules available in the App Inventor.

The final deliverable will be to create an app of your choice that tackles a real world issue or an issue that you are currently facing in your life, with an intuitive design and easy to understand code.

## **SOFTWARE NEEDED**

- nil

## **WHO SHOULD TAKE THIS?**

- Anyone who's wanted to try creating your own apps
- (preferably) Students with a general understanding of basic coding concepts(e.g. Variables, Loops, Conditionals)



# IOS APP DEVELOPMENT

MENTOR: CHRISTINE

## **DESCRIPTION:**

- The project introduces Apple's iOS development framework SwiftUI through Swift Playgrounds, a beginner-friendly yet powerful platform for building iOS apps.
- Through this project, you'll learn the fundamentals to iOS-specific UI/UX design, how the concept of accessibility is behind every iOS App and a new computing language, Swift.
- You can further extend your learning by exploring Apple technologies in Machine Learning (Vision, CoreML etc), Augmented Reality (ARKit, RealityKit), Game Development (SpriteKit) or other areas (Widgets, Maps and Location, Audio etc) too.

•  
Your final iOS app deliverable will be either a thoughtfully enhanced version of a provided app concept, or an original app developed from your own initiated idea.

## **SOFTWARE NEEDED**

- Students with an Apple iPad or Macbook (either works!)

## **WHO SHOULD TAKE THIS?**

- Students interested in learning how iOS apps are coded



# PYTHON TEXT ADVENTURE GAME

MENTOR: MEIHSIN

## **DESCRIPTION:**

- This project builds upon pre-existing knowledge on basic python, alongside databases, to construct a long text-adventure game with Object Oriented Programming (OOP) and functions. Minimal game design theory will also be taught to make the game more enjoyable to play.
- Although newly taught content is minimal, coding the game serves as a way to learn how to organize large volumes of code in a clear and concise way without confusing yourself.

The final deliverable will be a text-adventure game roughly an hour long, playable as an .exe file.

## **SOFTWARE NEEDED**

- nil

## **WHO SHOULD TAKE THIS?**

- Interested in game development/storywriting
- Enjoy more creative freedom when constructing your project
- Low commitment – Minimal new content taught
- Creating your own game!



# PYGAME GAME DEV

MENTOR: SHUANGYU

## DESCRIPTION:

- This project builds upon pre-existing knowledge on basic python, and implements event driven programming as well as Object orientated programming to create a 2d interactive game.
- It involves working with basic Python, event handling (e.g. key event, collision, etc) as well as math for creating the games' physics. There are countless possibilities for one to explore.

The final deliverable is a 2d game that minimally should include movement, functional UI and way to win/scoring function.

## SOFTWARE NEEDED

- code editor
- pygame python module

## WHO SHOULD TAKE THIS?

- Interested in game development
  - (preferable) Students who are able to design basic sprites
- 