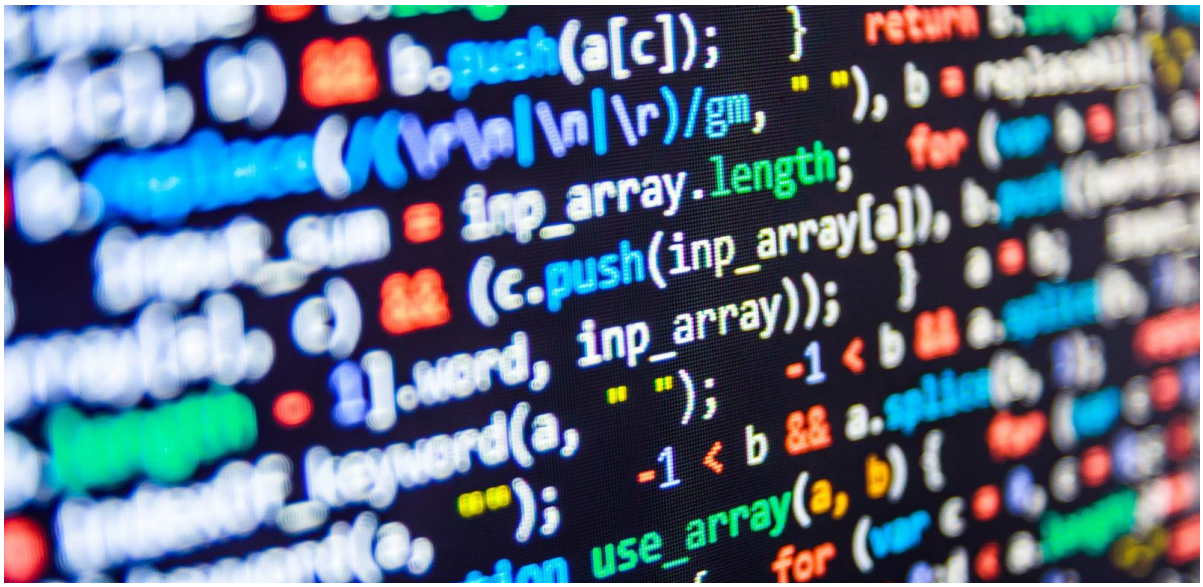
#CodeYork

Specification

For Summer 2017



Summary

This course is a **four week** variant of the five week **Python 3** course originally taught at Millthorpe School. The focus has been slightly shifted away from recursion and games to focus more on the basics of functions, and also to allow the presentation of some additional, simpler applications of programming. The problem sheets have been rewritten to match the new structure, and the slides improved and simplified. This includes the total removal of the original problem sheet and slides for connect four.

The Schedule

1. Introduction
    a. Types, Operators, and Variables
    b. If Statements and Lists
2. Functions and Control
    a. Introduction to Functions
    b. Further Control Structures
3. Recursion and Examples
    a. Introduction to Recursion
    b. Interesting Examples
4. Two Player Games
    a. Implementing Tic-Tac-Toe

Course Prerequisites

In order to take this course pupils must have been programming in **any language** for at least **16 hours** of contact time or equivalent. Pupils must have at least **some experience** with Python, and have used IDLE before, however, this need not be the language they've spend the most time programming in.

This best targets pupils in **year 10 or 11**. An aptitude for **mathematics** is helpful. In particular, familiarity with **order of operators**, **fractions**, **elementary algebra** and **functions** would be of use. Year 9 pupils would likely struggle with the later content unless they are particularly advanced in both programming and mathematics.

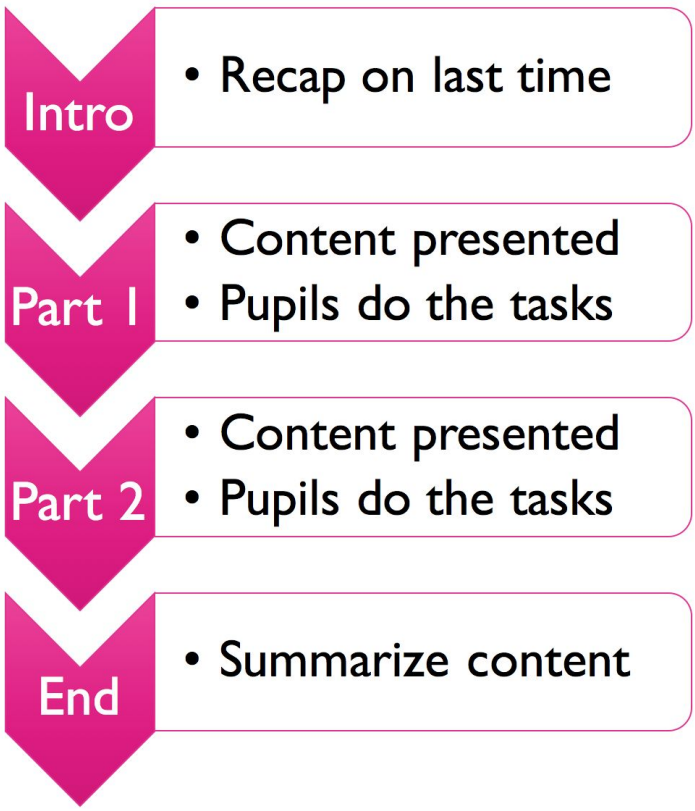
Lab Requirements

This course requires a software lab equipped with computers with **Python 3**, IDLE, and internet access. A projector and a screen will also be an essential resource as slides and interactive examples make up an important part of the process.

New Session Structure

Our first three sessions will implement a two part structure, where at the start of each part, we present some content on the board, and then the pupils have some tasks to do. This will ensure that the pupils remain engaged throughout the entirety of the session. Each session is designed to last **60 minutes**, so a time slot of around 70 minutes is ideal.

There will also feature a recap of the previous session's content at the beginning of each session, and a summary of the current session at the end. Our final session session will not implement this structure so tightly as the programming task is much larger.



Summary of Content

Our first session remains similar to the original taught at Millthorpe. We have, however, attempted to shorten the slides, and have adopted our session format, splitting the content into two pieces. We have also moved some of the more advanced control structures into session two.

Our second session covers functions and control structures. This has been constructed from various parts from the old Millthorpe sessions, with a small amount of new material. There will be no mention of recursion in this session unlike at Millthorpe.

Our third session will be looking first at recursion, and then at some other interesting applications of recursion and programming. We will not be setting questions on the second part of our content, so we will allow part one to run for more time than usual, and then, part two will present some content purely for interest's sake. *(It is possible to run this course by omitting this session and proceeding straight to the final session.)*

Our final session will be similar to our final session at Millthorpe. The pupils will be tasked with completing the code for a Python implementation of tic-tac-toe. This will allow them to put all the skills they've learnt into practice. *(Similarly, it is possible to run this course by omitting this session, and finishing at session three.)*

Extension Task

For students finding the content straight forward, or looking for something extra to do at home, we will be providing the same sudoku extension task designed for Millthorpe. We will not, however, be carrying forward the second extension task on sorting, as it proved to be too difficult, and not as interesting as the sudoku extension for the pupils.