Proposed Plan for Computer Science INSET at Tollbar May, 2013

Programming Skill	Preparation before the INSET	Covered during the INSET	Potentially covered
Problem Solving		✔	
Opening IDLE	✔		
Running an existing program	✔		
Printing to the screen	✔	✔	
Storing an input	✔	✔	
Simple arithmetic operations	✔	✔	
String handling techniques		✔	
Variables and assignment		✔	
Data types		✔	
Importing modules		✔	
EasyGUI module			✔
Random module		✔	
Time module		✔	
Selection (if statements)		✔	
If - Else if - Else		✔	
Iteration (loops - while)		✔	
Iteration (loops - for)		✔	
Functions and Procedures		✔	
Turtle module		✔	
Lists (1D arrays)			✔
Regular expression module			✔
File handling			✔

Preparation prior to the INSET

Completing the following tasks before the INSET will allow us to start a good couple of hours ahead of most introduction to programming courses.

It also means that complete novices and those with previous programming experience can start from a more level playing field.

Task 1

Install Python 3 from http://www.python.org/getit/ (the first bullet pointed link should do).

Once installed, launch IDLE, the Python editor.

Test by typing a couple of simple arithmetic statements:

```
Python 3.3.0 (v3.3.0:bd8
[GCC 4.2.1 (Apple Inc. b
Type "copyright", "credi
>>> 2+2
4
>>> 3*3
9
>>> 9/4
2.25
>>> 16-18
-2
>>> |
```
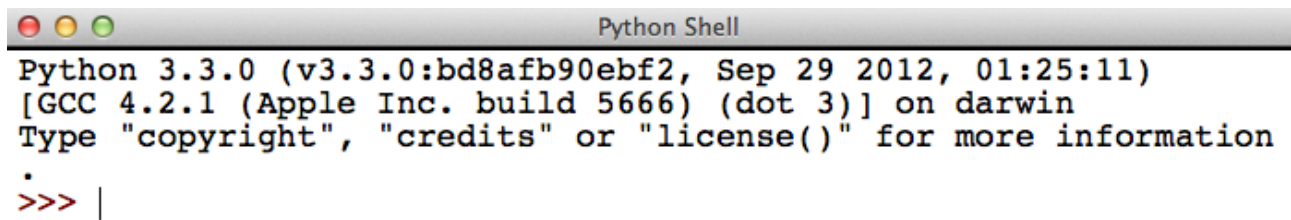
Task 2

In Python, click **File >> Open** and choose the file **quiz.py** (attached by email).

A code window will open. Press **F5** or click **Run >> Run Module** to run the program.

Task 3

Close any open IDLE windows so that only the **shell** window is open (the one that says **Python 3.3.0** or similar at the top.

Press **CTRL-N** or click **File >> New** to create a new program window. Keep these two windows side-by-side.

In the empty code window write this line (colour coding should happen automatically):

```python
print("Hello world!")
```

Save the file as **hello.py** and run it.

The shell window should display the phrase **Hello world!**

You can change the text inside the speech marks to say anything you want.

Task 4

Create a new file and write the following lines of code:

```python
name = input("What is your name? ")
print("Hello " + name)
```

Run the program and it should first ask for your name and then say **Hello** followed by whatever you typed in. Try experimenting by changing the text in speech marks and adding more text to the **print** command, e.g.

```python
name = input("What is your name? ")
print("Hello " + name + ", how are you today?")
```

Also try changing the spacing inside the speech marks and outside the speech marks to see what difference it makes.

Task 5

Open the file **numbers.py** and look at the code.

There are some bits here we don't want to get into just yet, but:

• The red lines starting with # are comments. These are for you, Python will ignore them

• The first two lines of code will ask for two numbers

• The next line of code will add them and store the answer

• The last line of code will print out the answer

Run the program and then follow the instructions in red to try different arithmetic functions.

That's it!

We'll recap and explain a couple of those bits so you can see what's actually happening, but if you've got this far then you can write and execute simple computer programs.

Proposed Plan

Tuesday Morning

• Recap basic input and output from preparation materials

• String manipulation (concatenation)

• Concept of a variable

• Data types and casting

• Importing the time module and using time.wait()

• Importing the random module and using random.randint()

Tuesday Afternoon

• Subject to technical requirements, importing and using the EasyGUI module

• Simple *if* statements, building up to *If then / else if / else* statements

• Problem solving and planning solutions to tasks

• Repeating code (while loops)

Wednesday Morning

• Repeating code (for loops)

• Turtle module

• Functions and procedures (making use of for loops and the turtle module)

• More problem solving

Wednesday Afternoon

TBD. Potentially:

• more programming (e.g. lists, file handling & regular expressions)

• looking at GCSE controlled assessment

• theory topics (e.g. binary conversion and boolean logic)