

Session 1

Introduction to Programming

- ✓ Define a program
- ✓ Programming languages
- ✓ Goals of programming

What is a Program?

Computers are tools for running instructions on data. A program is a series of instructions that tell a computer what to do to the data provided to the program.

As an example think of a computer program as a recipe, the input to a recipe are the raw ingredients that the recipe requires, the recipe describes, in order, what to do with the raw ingredients and the final output of a recipe is the completed meal.

Everything a computer does is built around programs and all programs require some form of data to process. For example moving the mouse or typing on a keyboard provides input to a special program on your computer that then handles that data and produces an output, the output is moving the cursor on the screen or displaying text on the screen.



Exercise

Write down the steps for making a sandwich. Could another person follow these instructions to create your sandwich (output)? Does it contain all the instructions necessary as well as the ingredients (input)?

How do we Make Computer Programs?

Recipes are written using a human language which wouldn't be very useful for a computer as it's far too vague; we have special languages for creating computer

programs called programming languages. Just as there are hundreds of different human languages there are hundreds of different programming languages. Some languages are more specific to a certain type of problem for example SQL (Structured Querying Language) is designed for databases, or MATLAB which is designed for doing complex mathematical computations. Most languages are quite generic or general purpose, by generic we mean that they can be used for all sorts of problems and programs.

Let's talk about some of the most popular programming languages and see some examples of each. Take note of how each example is so different between the languages yet each program produces the same output.

C++

C++ is a relatively old, generic programming language, it can be functional or object orientated and also allows low level memory manipulation.

When used correctly it can be fast, efficient and flexible. It is frequently used in systems that require performance such as operating system modules, embedded systems or computer video games.

Here's an example of a C++ hello world application.

```
# include <iostream>

int main()
{
    std::cout << "Hello, world!\n";
}
```

Java

Java is a more recent generic programming language which is class based and object orientated. It has always been very popular amongst developers due to the ability to 'write once, run anywhere', it's currently in use on a wide variety of platforms such as smart televisions and mobile phones.

Java derives a lot of syntax from C and C++ but without the low level memory

manipulation.

Let's have a look at hello world in Java.

```
class HelloWorldApp {
    public static void main(String[] args) {
        System.out.println("Hello World!"); // Display the
string.
    }
}
```

PHP

PHP has evolved from a basic web scripting hack into one of the most widely used web languages in the world. PHP is a generic language with a large amount of built in support for web applications, it can be embedded into HTML. The major users of PHP are Wordpress and Facebook.

Most websites run some form of PHP but despite it's popularity there is no formal specification.

Here's an example of hello world in PHP.

```
<!DOCTYPE html>
<html>
  <head>
    <title>PHP Test</title>
  </head>
  <body>
    <?php echo '<p>Hello World</p>'; ?>
  </body>
</html>
```

Python

Python is an extremely popular language which is part of a new breed of languages that don't follow the stricter C style of syntax and allow more freedom to the programmer. Python promotes code readability, liberal syntax, has a dynamic typing and has a large

amount of functionality available by standard. Due to these benefits it is the language that will be used during this course. Let's have a look at the hello world application in Python.

```
print "Hello, world!"
```



Exercise

In the previous section we introduced a lot of new words; look up the definitions of the following words -

- *Generic*
- *Syntax*
- *Function*

Optional exercise - Find the home pages or Wikipedia entries of the previously mentioned languages and familiarise yourself with each of them.

Goals of Programming

So far we've seen what a program is and we've seen some of the languages we can use to write our own programs, but what are the actual goals of programming?

Firstly your program should do one thing and do it well, users will not react well to a huge program that attempts to do everything but fails to do half of what it promises. Focus on the problem you're trying to solve.

Secondly ensure that your application is easy to use; it doesn't matter how fast or efficient an application is if it's a pain to use. Keep it simple.

Thirdly, ensure that the code you write is clean and understandable by other programmers. In the event that you have to take over code written by another person you will be extremely happy when you see a well documented and structured piece of code instead of code spaghetti.