Glossary of programming terms

This glossary explains some of the words and phrases that we use in the course. It's a work in progress, so if there's anything you think is missing, let us know in the comments and we'll add them to the document.

Name	Description
Access level modifiers	Access level modifiers are used in object oriented programming to control access to different parts of the code. For example, we might use an access modifier to stop certain parts of our program changing the values of a variable. You will see the modifiers 'public', 'private' and 'protected' in our code, but this is an advanced concept that we will not be discussing in detail as part of this course. If you are interested in reading more about access modifiers you can find further information in the Java documentation.
Algorithm	An algorithm is a step-by-step solution to a given problem with a beginning and an end. Algorithms are written in 'human readable' language (e.g. Standard English) rather than a programming language.
Android Jelly Bean	Level 16 of the API with improved and additional features to the previous version of the API.
Argument	See parameter.
Array	An array is made up of a defined number of 'elements' or values. Each element is of the same data type. Elements within an array are identified by their position or 'index'. The index is numbered, starting with 0.
Assignment	The process of placing a new value into a variable.
Boolean Expression	A Boolean expression is a type of expression that always results in a Boolean value, either 'true' or 'false'. For example 0 < 5 is a Boolean expression that results in 'true'.
Boolean Type	A data type with two possible values: true and false.
Breakpoint	A breakpoint temporarily halts a program during execution. They are used

	by programmers to intentionally halt a program at a specific point in the code during the debugging process.
Bug	A programming error. Can be identified and resolved by 'debugging'.
Case sensitive	Distinguishing between upper and lower case characters. That is 'A' and 'a' are considered two different characters.
Character	A single letter, digit or symbol.
Class	A class is an advanced topic that we will not cover in full in this course. Classes are used by developers to define user-generated data types. For example, in one video, we mention that bitmaps can be stored in a variable. This is achieved in Java by writing a class that stores all the information about bitmaps and has all the functionalities that bitmaps require, allowing them to be used by other developers effectively.
Code	See source code.
Code block	A set of statements that are grouped together. In some languages, such as C++ and Java, a block of code is represented by placing them between curly brackets {}.
Comment	A description or explanation to help someone understand a section of a program; comments are ignored by compilers. Single line comments are prefaced by two forward slashes (`//`), while multi-line comments must be surrounded with `/* content of comment */`.
Compile time error	An error that is detected when the program is being compiled.
Compiler	A program that translates code in a 'high-level language' (such as Java or C++) to machine readable instructions for the computer.
Computer program	A sequence of instructions that is executed by a computer.
Data Type	All data are represented as binary values in the computer systems. Different data types are used to help the system understand what type of data your program is working with and how much space to be allocated to store the content of the value.
Emulator	An emulator is a piece of software that allows you to run programs for one system on another. In this course, we are using the Android emulator that allows you run software that has been developed for Android devices such as phones and tablets on your computer.
Function	Functions (also called 'procedures' in some programming languages and 'methods' in most Object Oriented Programming languages are a set of instructions bundled together to achieve a specific outcome.
IDE (Integrated Development Environment)	An Integrated Development Environment is a program that provides comprehensive facilities for software development. Most IDEs consists of at least a source code editor and a debugger.

Integer	A whole number - ie one that does not have a fractional part. 5 is an integer, 5.1 is not.
Integer type	Data type defined to hold an integer value
Java bytecode	The instruction for the Java Virtual Machine.
JDK (Java Development Kit)	The Java software development kit that contains the Java compiler and related development tools that are used by developers to create programs in Java. We will use this as part of the course.
JRE (Java Runtime Environment)	The Java Runtime Environment is the software used to run Java programs on your computer. Most people will already have this to allow them to use web-based content developed in Java. The JRE does not include the tools required to develop programs in Java, which is why we need to download the JDK as part of this course.
JVM	The Java Virtual Machine is the code execution component of the Java platform. It executes Java bytecode.
Loop	A sequence of instructions that is executed repeatedly
Method	A method is essentially a function. In object oriented programming the term 'method' is more commonly used.
Nested loop	A loop that is contained in another loop
Object	Is a single instance of the defined class. If we have defined a class for bitmaps, then each individual bitmap would be an object.
Object Oriented Programming (OOP)	OOP is a style of programming in which a program is designed by describing objects, their properties and their relationships to one another.
Object Oriented Programming languages	Computer languages that support the development of object oriented programs (eg Java, C++).
Operand	An operand is the variable that an operator affects. For example in 2 + 3 the operator (+) takes two operands (2 and 3) and adds them together.
Operator	Operators are special symbols that perform defined operations on one or more operand(s). For example the - operator subtracts one operand from another.
Operator precedence	Operator precedence is the rule that defines the order in which the operators are executed in a statement. This order is pre-defined and this should be adhered to in reading and writing programmes.
Parameter	Parameters (also called arguments) are values passed into a function.
Platform	A computing platform is hardware and software that allows a program to run.
Procedure	A procedure is also a function. In some programming languages when a function does not return a value it is called a procedure.
Reserved word	A word that has a special meaning in a programming language and therefore cannot be used as a name by the programmer.

Return value	The value returned by a method through a return statement.
SDK (Software Development Kit)	The collection of software tools that programmers use to develop executable code.
Source code	Instructions in a programming language that need to be translated (or 'compiled') before execution on a computer.
String	A sequence of characters.
Syntax error	An instruction that does not follow the programming language rules (also known as the 'syntax') and is rejected by the compiler.
Unary Operator	An operator that acts on one operand.
Variable	A variable is an allocated location in memory that is used to store a value. A name is used to identify a variable.