



Business & Computing Examinations (BCE) LONDON (UK)

Programming Qualification Analysis

The development of BCE qualifications include extensive market research from the following sources:

- Data from BCE Centre Annual Reports.
- Enquiries received from different stakeholders.
- Email survey from statutory consultees and stakeholder bodies.
- Questionnaire survey from BCE learners.
- Input received during Approved Centres and Corporate companies training seminar.
- BCE discussions and feedback from potential employers.

BCE learners are 18+, classified as follows:

- Holders of General Certificate of Secondary Education (GCSE) intending to obtain a qualification for employment or further education.
- Those already in employment furthering their knowledge for promotion or to venture into new fields.
- Corporate Companies approaching BCE directly or Approved Centres for in-house training.
- Those looking for career change.
- Mature adults with no formal qualifications.

Total Qualification Time (TQT) is the entire notional learning hours representing estimate of total amount of time reasonably required for learners to achieve necessary level of attainment for the award of a qualification. This comprises of **TQT** and **Guided Learning**.

Activities that contribute to TQT include:

- Guided Learning
- Independent and unsupervised research/learning
- Unsupervised compilation of a portfolio of work experience
- Unsupervised e-learning
- Unsupervised e-assessment
- Unsupervised coursework
- Watching a pre-recorded podcast or webinar
- Unsupervised work-based learning

Activities that contribute to Guided Learning include:

- Classroom-based learning supervised by a Tutor
- Work-based learning supervised by a Tutor
- Live webinar or telephone tutorial with a Tutor in real time
- E-learning supervised by a Tutor in real time
- Forms of assessment

Level 5 Diploma in Programming (157 Credits)

A computer programming qualification allow learners to work in a diverse industry within the computer technology environments. Computer programmers are in high demand in today's technologically advancing industries, and learners undertaking the computer programming qualification will receive formal, hands-on computer programming training that is necessary for today's growing market. Each programming language requires strong mathematical abilities and understanding technical concepts, with direct application for a variety of software and computer programs.

Why does the qualification exists – Everything to do with computers is based on programming. The qualification enable learners to explore a variety of computer languages, which helps immensely in understanding computer programming in general and also when looking for employment.

How it fits into the larger programme – Programming is at the *core* of computing. For a nation to be fully computerised, programmers have to play a centre role. Programmers write programs which enable computers to

switch on, process data and produce information. Without programmers, there would be no computers, mobile phones, planes, microwaves, digital television etc.

For whom it was designed – Learners who complete the Diploma in System Design interested in pursuing programming.

How it will benefit learners – Computer programming training lays the foundation for a variety of career options for prospective computer programmers, research analysts, and database administrators.

Units:

- Programming Principles and Paradigms
- C Programming
- VB .Net Programming
- C++ Programming
- Java Programming

Programming Principles and Paradigms - examines the principles and underlying programming grammatical analysis and syntax.

C Programming - a popular and widely used programming language for creating computer programs. Programmers around the world embrace C because it gives maximum control and efficiency to the programmer. C was initially used for system development work, in particular the programs that make-up the operating system. C is mainly used because it produces code that runs nearly as fast as code written in assembly language (a low level programming language). C Programming is used to create: operating systems; language compilers; assemblers; text editors; print spoolers; network drivers; modern programs; data bases; language interpreters and utilities.

VB .NET Programming - the .NET Framework is Microsoft's application development platform that enable developers to easily create Windows applications, web applications, and web services using a myriad of different programming languages, and without having to worry about low-level details like memory management and processor-specific instructions. Visual Basic is a computer programming system developed and owned by Microsoft. Visual Basic was originally created to make it easier to write programs for the Windows computer operating system. The basis of Visual Basic is an earlier programming language called BASIC that was invented by Dartmouth College professors John Kemeny and Thomas Kurtz. Visual Basic is often referred to using just the initials, VB. Visual Basic is easily the most widely used computer programming system in the history of software.

C++ Programming - created in 1983 by Bjarne Stroustrup, C++ was designed to serve as an enhanced version of the C programming language. C++ is object oriented and is considered a high level language. However, it features low level facilities. C++ is one of the most commonly used programming languages. The development of C++ actually began four years before its release, in 1979. It did not start out with the name C++; its first name was C with Classes. In the late part of 1983, C with Classes was first used for AT&T's internal programming needs. Its name was changed to C++ later in the same year. C++ was not released commercially until the late part of 1985. Developed at Bell Labs, C++ enhanced the C programming language in a variety of ways. Among the features of C++ are classes, virtual functions, templates, and operator overloading.

Java Programming – is a high-level programming language developed by Sun Microsystems. Java was originally called OAK, and was designed for handheld devices and set-top boxes. Oak was unsuccessful so in 1995 Sun changed the name to Java and modified the language to take advantage of the burgeoning World Wide Web. Java is an object-oriented language similar to C++, but simplified to eliminate language features that cause common programming errors. Java source code files (files with a .java extension) are compiled into a format called bytecode (files with a .class extension), which can then be executed by a Java interpreter. Compiled Java code can run on most computers because Java interpreters and runtime environments, known as Java Virtual Machines (VMs), exist for most operating systems, including UNIX, the Macintosh OS, and Windows. Bytecode can also be converted directly into machine language instructions by a just-in-time (JIT) compiler. Java is a general purpose programming language with a number of features that make the language well suited for use on the World Wide Web. Small Java applications are called Java applets and can be downloaded from a Web server and run on the PC by a Java-compatible Web browser, such as Firefox or Internet Explorer.

		Total	Number
--	--	-------	--------

Unit	Pre-requisite	Core-requisite	Qualification Time (TQT)	of Credits
Programming Principles & Paradigms	QBasic and Pascal programming knowledge	A pass or higher in Diploma in System Design or equivalence.	280	28
C Programming	Basic programming skills or basic knowledge of computer use.	A pass or higher in Diploma in System Design or equivalence.	220	22
VB .Net Programming	Knowledge of C Programming Language.	A pass or higher in Diploma in System Design or equivalence.	300	30
C++ Programming	Knowledge of C Programming Language.	A pass or higher in Diploma in System Design or equivalence.	220	22
Java Programming	Knowledge of C Programming Language.	A pass or higher in Diploma in System Design or equivalence.	240	24
Coursework (Project) for all units			310	31

Rules of combination:	All units are mandatory
Age Group:	18+
Qualification Type:	Vendor/Industry

Programming Principles Learning Hours Information Sheet

Unit Titles		Credits	Notional Learning Hours					Total
			Guided / Contact Learning	Independent Learning	Research Activities / Group Work	Assessment (self/class)	Coursework	
01	Programming paradigm	2.0	8	6	2	2	2	20
02	Programming language syntax	2.0	8	6	2	2	2	20
03	Formal syntax and semantics of programming languages	2.0	8	6	2	2	2	20
04	Program scope and characteristics of variables	2.0	8	6	2	2	2	20
05	Value-level programming	2.0	8	6	2	2	2	20
06	Semantics of expression	2.0	8	6	2	2	2	20
07	Programming language functions (procedures/methods)	2.0	8	6	2	2	2	20
08	Programming languages memory management	2.0	8	6	2	2	2	20
09	Functional vs procedural programming	2.0	8	6	2	2	2	20
10	Object oriented programming concepts	2.0	8	6	2	2	2	20
11	Functional programming	2.0	8	6	2	2	2	20
12	Semantics of programming languages	2.0	8	6	2	2	2	20
13	Event-based programming paradigm	2.0	8	6	2	2	2	20
14	Concurrency concepts	2.0	8	6	2	2	2	20
		28.0	112					280

C Programming Learning Hours Information Sheet

Unit Titles		Credits	Notional Learning Hours					Total
			Guided / Contact Learning	Independent Learning	Research Activities / Group Work	Assessment (self/class)	Coursework	
01	Creating simple C programs	2.0	8	6	2	2	2	20
02	Structured program development	2.0	8	6	2	2	2	20
03	C program control	2.0	8	6	2	2	2	20
04	Invoking and calling functions in C	2.0	8	6	2	2	2	20
05	Array structures in C	2.0	8	6	2	2	2	20
06	Using pointers in C	2.0	8	6	2	2	2	20
07	Fundamentals of strings and characters in C Programming	2.0	8	6	2	2	2	20
08	Formatting input/output to present results	2.0	8	6	2	2	2	20
09	C structures, enumerations, unions and bit manipulations	2.0	8	6	2	2	2	20
10	File processing in C	2.0	8	6	2	2	2	20
11	Linked lists, stacks queues and binary trees	2.0	8	6	2	2	2	20
		22	88					220

VB .Net Programming Learning Hours Information Sheet

Unit Titles	Credits	Notional Learning Hours					
		Guided / Contact Learning	Independent Learning	Research Activities / Group Work	Assessment (self/class)	Coursework	Total
01 VB .Net menus and toolbars	2.0	8	6	2	2	2	20
02 Executing VB .Net applications	2.0	8	6	2	2	2	20
03 VB.Net objects	2.0	8	6	2	2	2	20
04 Using event handlers	2.0	8	6	2	2	2	20
05 Debugger and syntax errors	2.0	8	6	2	2	2	20
06 Control structures	2.0	8	6	2	2	2	20
07 Logical operators	2.0	8	6	2	2	2	20
08 Repetition statements	2.0	8	6	2	2	2	20
09 Counter-controlled repetition	2.0	8	6	2	2	2	20
10 Case multiple-selection statements	2.0	8	6	2	2	2	20
11 Classes, procedures and functions	2.0	8	6	2	2	2	20
12 Date and timer controls	2.0	8	6	2	2	2	20
13 Passing arguments by reference	2.0	8	6	2	2	2	20
14 Declaring and allocating arrays	2.0	8	6	2	2	2	20
15 Sequential access files	<u>2.0</u>	<u>8</u>	<u>6</u>	<u>2</u>	<u>2</u>	<u>2</u>	<u>20</u>
	30.0	120					300

C++ Programming Learning Hours Information Sheet

Unit Titles		Credits	Notional Learning Hours					Total
			Guided / Contact Learning	Independent Learning	Research Activities / Group Work	Assessment (self/class)	Coursework	
01	Creating simple C++ programs	2.0	8	6	2	2	2	20
02	C++ standard library functions and classes	2.0	8	6	2	2	2	20
03	Passing information between functions	2.0	8	6	2	2	2	20
04	Array structures in C++	2.0	8	6	2	2	2	20
05	Declaring pointers in C++	2.0	8	6	2	2	2	20
06	C++ Abstract Data Types	2.0	8	6	2	2	2	20
07	Objects, functions and classes	2.0	8	6	2	2	2	20
08	Operator overloading	2.0	8	6	2	2	2	20
09	Implementation of inheritance in OOP	2.0	8	6	2	2	2	20
10	Polymorphism in OOP	2.0	8	6	2	2	2	20
11	File processing in C++	2.0	8	6	2	2	2	20
		22.0	88					220

Java Programming Learning Hours Information Sheet

Unit Titles		Credits	Notional Learning Hours					Total
			Guided / Contact Learning	Independent Learning	Research Activities / Group Work	Assessment (self/class)	Coursework	
01	Basics of the Java environment	2.0	8	6	2	2	2	20
02	Java applications	2.0	8	6	2	2	2	20
03	Java applets	2.0	8	6	2	2	2	20
04	Java control-structures	2.0	8	6	2	2	2	20
05	Java counter control-structures	2.0	8	6	2	2	2	20
06	Using methods in Java	2.0	8	6	2	2	2	20
07	Using arrays in Java	2.0	8	6	2	2	2	20
08	Object-based programming	2.0	8	6	2	2	2	20
09	Object-oriented programming – inheritance	2.0	8	6	2	2	2	20
10	Object-oriented programming – polymorphism	2.0	8	6	2	2	2	20
11	Java character and string classes	2.0	8	6	2	2	2	20
12	Java files and streams	2.0	8	6	2	2	2	20
		24	96					240

Level 6 Advanced Diploma in Programming (163 Credits)

The Level 6 Advanced Diploma in Programming enhances programming skills acquired in the Level 5 Diploma in Programming.

Why does the qualification exist – Programming is one of the most complicated qualifications. To be competitive, learners need to be skilled and knowledgeable. By undertaking the Advanced level, learners can produce highly marketable programs.

How it fits into the larger programme – C, C++, VB .Net and Java programming are the core programming languages required in the industry. Advanced skills are essential to learners.

For whom it was designed – Learners who complete the Level 5 Diploma in Programming.

How it will benefit learners – Learners will be efficient in most sought out computer programs.

Units:

- Advanced Operating System Principles
- Advanced C Programming
- Advanced VB .Net Programming
- Advanced C++ Programming
- Advanced Java Programming

Advanced Operating System Principles - programming languages are controlled by operating systems. Knowledge of the behind the scenes of an operating system and how it controls the programming being written is equally as important programming itself.

Advanced C Programming - This unit broadens the skills of a C language programmer by introducing sophisticated problem solving techniques including the advanced use of pointers, abstract data types, data structures, portability, and optimization techniques. Skills are reinforced by hands-on laboratory exercises.

Advanced VB .NET Programming - This unit covers Microsoft .Net architecture, concentrating on more advanced features provided by the Visual Basic.NET programming environment. It addresses ways to use Visual Basic.NET features and the .NET framework to build real world distributed applications, demonstrating techniques and implementation steps to build each tier of a modern multi-tier application. Concepts covered include the Microsoft .NET architecture; object-oriented features of Visual Basic.NET; distributed application architecture; database connectivity features of .NET; implementing Windows based forms and web-based graphical user interfaces; web services; and performance and scalability issues.

Advanced C++ Programming - Expanding on several topics in C++, this unit includes object-oriented analysis, design, and programming. Advanced memory management, stream and file I/O, persistence, multiple inheritance, advanced polymorphic programming, templates, Standard Template Library (STL) libraries, C++ style, and efficiency are also discussed.

Advanced Java Programming – This unit introduce learners to advanced features of the Java programming language. Learners will learn how to use inheritance, interfaces, exception handling, file input and output, and generic types, and how to incorporate graphical user interfaces (GUIs) into their programming applications. Learners will also learn how to apply object-oriented design and programming principles to their programs. Typical assignments and projects include using built-in and programmer-defined classes to develop full-featured, easy-to-use programs.

Unit	Pre-requisite	Core-requisite	Total Qualification Time (TQT)	Number of Credits
Advanced Operating System Principles	Programming skills and Operating System knowledge.	A pass or higher in Diploma in Programming or equivalence.	300	30
Advanced C Programming	Programming experience in C for at least six months.	A pass or higher in Diploma in Programming or equivalence.	240	24
Advanced VB .Net	Programming experience in VB .Net for at least six months.	A pass or higher in Diploma in Programming or equivalence.	260	26
Advanced C++ Programming	Programming experience in C++ for at least six months.	A pass or higher in Diploma in Programming or equivalence.	260	26
Advanced Java Programming	Programming experience in Java for at least six months	A pass or higher in Diploma in Programming or equivalence.	260	26
Coursework (Project) for all units			310	31

Rules of combination:	All units are mandatory
Age Group:	19+
Qualification Type:	Vendor/Industry

Advanced Operating System Principles Learning Hours Information Sheet

Unit Titles	Credits	Notional Learning Hours					
		Guided / Contact Learning	Independent Learning	Research Activities / Group Work	Assessment (self/class)	Coursework	Total
01 Operating system architecture	2.0	8	6	2	2	2	20
02 Networked and Distributed Operating Systems	2.0	8	6	2	2	2	20
03 Caching and buffering	2.0	8	6	2	2	2	20
04 Computer hardware components	2.0	8	6	2	2	2	20
05 Interrupt Processing	2.0	8	6	2	2	2	20
06 Threads and processes implementation	2.0	8	6	2	2	2	20
07 Semaphores	2.0	8	6	2	2	2	20
08 Operating system deadlocks	2.0	8	6	2	2	2	20
09 Processor scheduling	2.0	8	6	2	2	2	20
10 Scheduling Algorithms	2.0	8	6	2	2	2	20
11 Virtual memory management	2.0	8	6	2	2	2	20
12 Disk space management tools	2.0	8	6	2	2	2	20
13 File system technologies	2.0	8	6	2	2	2	20
14 Operating system performance monitoring	2.0	8	6	2	2	2	20
15 Distributed file system	<u>2.0</u>	<u>8</u>	<u>6</u>	<u>2</u>	<u>2</u>	<u>2</u>	<u>20</u>
	30.0	120					300

Advanced C Programming Learning Hours Information Sheet

Unit Titles	Credits	Notional Learning Hours						
		Guided / Contact Learning	Independent Learning	Research Activities / Group Work	Assessment (self/class)	Coursework	Total	
01	String functions	2.0	8	6	2	2	2	20
02	String Handling Library	2.0	8	6	2	2	2	20
03	Standard input/output stream	2.0	8	6	2	2	2	20
04	Passing structures to functions	2.0	8	6	2	2	2	20
05	Enumeration Constants	2.0	8	6	2	2	2	20
06	Implementing complex data structures	2.0	8	6	2	2	2	20
07	Pre-process directives	2.0	8	6	2	2	2	20
08	C Programming advanced concepts	2.0	8	6	2	2	2	20
09	Command-Line Arguments	2.0	8	6	2	2	2	20
10	Strings and pointers	2.0	8	6	2	2	2	20
11	Derived data types	2.0	8	6	2	2	2	20
12	Binary files	2.0	8	6	2	2	2	20
		24	96					240

Advanced VB .Net Programming Learning Hours Information Sheet

Unit Titles	Credits	Notional Learning Hours						
		Guided / Contact Learning	Independent Learning	Research Activities / Group Work	Assessment (self/class)	Coursework	Total	
01	.NET Framework	2.0	8	6	2	2	2	20
02	Classes in VB .Net	2.0	8	6	2	2	2	20
03	Objects in VB .Net	2.0	8	6	2	2	2	20
04	Formulating Algorithms	2.0	8	6	2	2	2	20
05	Using collections in VB .Net	2.0	8	6	2	2	2	20
06	Handling mouse events in VB .Net	2.0	8	6	2	2	2	20
07	Working with keyboard events in VB .Net	2.0	8	6	2	2	2	20
08	Graphical User Interfaces	2.0	8	6	2	2	2	20
09	VB .Net OO implementation	2.0	8	6	2	2	2	20
10	Database connection in VB .Net	2.0	8	6	2	2	2	20
11	Creating a VB .Net Web service	2.0	8	6	2	2	2	20
12	Web based applications	2.0	8	6	2	2	2	20
13	Implementing web controls	2.0	8	6	2	2	2	20
		26	104					260

Advanced C++ Programming Learning Hours Information Sheet

Unit Titles	Credits	Notional Learning Hours					
		Guided / Contact Learning	Independent Learning	Research Activities / Group Work	Assessment (self/class)	Coursework	Total
01 C++ classes and objects	2.0	8	6	2	2	2	20
02 Container Classes and Iterators	2.0	8	6	2	2	2	20
03 Function Overloading	2.0	8	6	2	2	2	20
04 C++ operator overloading	2.0	8	6	2	2	2	20
05 C++ inheritance	2.0	8	6	2	2	2	20
06 Composition vs. Inheritance	2.0	8	6	2	2	2	20
07 C++ polymorphism	2.0	8	6	2	2	2	20
08 C++ class templates	2.0	8	6	2	2	2	20
09 C++ input/output streams	2.0	8	6	2	2	2	20
10 Exception handling in C++	2.0	8	6	2	2	2	20
11 Web programming with Common Gateway Interface	2.0	8	6	2	2	2	20
12 Linked lists, stacks, queue and tree data structures	2.0	8	6	2	2	2	20
13 Bits, characters, strings and structures	<u>2.0</u>	<u>8</u>	<u>6</u>	<u>2</u>	<u>2</u>	<u>2</u>	<u>20</u>
	26	104					260

Advanced Java Programming Learning Hours Information Sheet

Unit Titles	Credits	Notional Learning Hours					
		Guided / Contact Learning	Independent Learning	Research Activities / Group Work	Assessment (self/class)	Coursework	Total
01 Java graphic capabilities	2.0	8	6	2	2	2	20
02 Java graphical user interface controls	2.0	8	6	2	2	2	20
03 Advanced graphical user interface components	2.0	8	6	2	2	2	20
04 Java exception and error handling	2.0	8	6	2	2	2	20
05 Multithreading in Java	2.0	8	6	2	2	2	20
06 Java network programming	2.0	8	6	2	2	2	20
07 Java multimedia programming	2.0	8	6	2	2	2	20
08 Java dynamic data structures	2.0	8	6	2	2	2	20
09 Java utilities package and bit manipulation	2.0	8	6	2	2	2	20
10 Java collections framework	2.0	8	6	2	2	2	20
11 Java database connectivity	2.0	8	6	2	2	2	20
12 Java servlets technology	2.0	8	6	2	2	2	20
13 Java server pages technology	2.0	8	6	2	2	2	20
	26.0	104					260