

BCS Higher Education Qualifications

Certificate in IT Software Development Syllabus

Version 3.0

December 2016

This is a United Kingdom government regulated qualification which is administered by one or more of the following: Ofqual, Qualification in Wales or SQA

Contents

1. Change History	3
2. Rationale	3
3. Aims	3
4. Objectives	4
5. Prior Knowledge Expected	4
6. Format and Duration of the Examination	4
7. Syllabus Detail	5
8. Recommended Reading List	6
9. Contact Points	6

1. Change History

Any changes made to the syllabus shall be clearly documented with a change history log. This shall include the latest version number, date of the amendment and the changes made. The purpose is to identify quickly what changes have been made.

Version Number	Date	Changes Made
Version 1.0	May 2014	Released
Version 2.0	February 2016	Re-formatted with syllabus numbering and insertion of reading list – minor change to content
Version 2.0	Dec 2016	Regulation statement added

2. Rationale

Programming, using many styles and languages, provides solutions to a wide variety of scientific, engineering and business problems. Programming is a core skill that will be used throughout a computer practitioner's career. It is a skill acquired largely by practice and experience. Learning how to program requires a disciplined and structured approach to encourage good practice and to assist in the development of easily maintained systems. This module introduces candidates to the fundamental concepts of programming with the emphasis being laid on the whole of the software development process.

Candidates are free to use the programming language of their choice but it is recommended that a C type language is capable of being understood, as any examples of code given in the examination paper will be based on C.

3. Aims

- To stress the importance of good design, documentation and usability
- To emphasise skills in problem solving and algorithm specification rather than just writing syntactically correct code
- To introduce a systematic approach to algorithm development which will assist in subsequent programming and system design modules
- To introduce candidates to the environment in which software is developed and to the tools that assist in this process

4. Objectives

Upon successful completion of this module, candidates will be able to demonstrate their competence in, and their ability to:

- Distinguish between systems software and application software
- Understand the phases of software development
- Be able to develop and understand algorithms
- Be able to develop code from algorithms in a 3rd generation high level programming language
- Be able to follow 3rd generation high level code and apply modifications to it
- Develop competence in the techniques of systematic problem analysis, program construction and documentation
- Gain an understanding of the basic concepts of good user-interface design
- Understand and develop test strategies
- Understand the need for quality assurance/security in software development and its operation
- Gain an understanding of the principles of multiple module program construction
- Understand the need for compilers, interpreters, code generators
- Develop a knowledge and understanding of a range of fundamental algorithms

5. Prior Knowledge Expected

There are no specific entrance requirements for the Certificate in IT, however it is strongly recommended that all candidates register with an approved centre. Studying with an approved centre will deliver significant benefits.

Candidates are required to become a member of BCS, The Chartered Institute for IT to sit and be awarded the qualifications. Candidates may apply for a four-year student membership that will support them throughout their studies.

6. Format and Duration of the Examination

The examination is a two-hour closed book examination (no materials can be taken into the examination room) based on the syllabus in this document.

Examinations are held twice a year and are undertaken in normal examination conditions with one or more duly appointed invigilators.

The pass mark is 40%.

7. Syllabus Detail

Category	Ref	Content
1 FUNDAMENTAL CONCEPTS OF THE PROGRAMMING PROCESS	1.1	Concept of an algorithm
	1.2	Control structures: selection and iteration
	1.3	Development and semi-formal specification of algorithms, based on a simplified computer model Development of code from an algorithm
2 PHASE-SPECIFIC ISSUES OF SOFTWARE DEVELOPMENT	2.1	Development techniques such as modular programming, defensive programming and recursion
	2.2	Approaches to software build, such as evolutionary prototyping or 4GL development
	2.3	Objectives and principles of testing and test-case specification Testing and debugging strategies including dry-running, white-box and black-box
	2.4	Styles of software documentation, such as for users or support personnel Content of software documentation such as GUI descriptions and maintenance details
	2.5	Role of quality assurance and security
	2.6	Impact of the cloud and emerging technologies
3 INTRODUCTION TO PROGRAMMING CONCEPTS	3.1	Types: numeric and non-numeric, elementary and derived, subtypes, and expressions such as assignments, input/output
	3.2	Subprograms: procedures and functions
	3.3	Data structures: Arrays (1- and 2-dimensions), implementation of queues, stacks and lists. Concept of data abstraction
	3.4	Sorting and searching algorithms: comparative effectiveness with respect to computation and storage of scanning versus indexing methods
4 FILES: SEQUENTIAL, INDEX-SEQUENTIAL AND RANDOM ACCESS	4.1	Comparative effectiveness of storage and retrieval for applications such as batch processing or on-line query or both
5 INTRODUCTION TO CONCEPT OF USER-INTERFACE DESIGN	5.1	User requirements and characteristics of user interfaces; principles and techniques of dialogue control, navigation and selection
6 ROLE AND NEED FOR SYSTEM SOFTWARE	6.1	System software and its relation to application software
7 CASE STUDIES IN PROBLEM SOLVING/ALGORITHM ANALYSIS	4.1	Understand case studies and develop algorithms to solve the problems

8. Recommended Reading List

Module Name	ISBN 10	ISBN 13
Primary Texts		
• Lesley Anne Robertson, Simple Program Design: A Step-By-Step Approach, Course Technology Inc (5 th Ed), 2006.	1423901320	978-1423901327
• Bell, D. Software Engineering for Students: A Programming Approach, Addison Wesley (4th Ed), 2005.	0321261275	978-321261274
Indicative Programming Texts		
Java Texts		
• Goodrich, Michael, T, Tamassia, R., Data Structures & Algorithms in Java, John Wiley and Sons, 6th Edition, 2014.	1118771338	978-118771334
• Horstmann, C.S., Java Concepts 6/E for Java 7 and 8 International Student Version, Wiley, 2010.	0470561599	978-470561591
• Deitel, H., Deitel, P Java How to Program, Pearson (8th Ed), 2010	0131364839	978-0131364837
C/C++ Texts		
• Savitch, W., Problem solving with C++, Pearson (7th Ed), 2009. 0321549406 978-0321549402	0321549406	978-0321549402
• C How to program, 7/E Paul Deitel & Harvey Deitel (Prentice Hall 2012)	013299044X	978-0132990448
Other Reading		
Other textbooks that describe introductory programming will be appropriate. The computer trade press and the computing/IT supplements of newspapers will help to give candidates both an understanding of the scope of the discipline and introduce new developments in the field.		

9. Contact Points

Email:

Customer Service team via www.bcs.org/contact

Phone:

UK: 01793 417424 or 0845 300 4417 (lo-call rate)

Overseas: +44 (0)1793 417424

Lines are open Monday to Friday, 08.15 a.m. to 5.45 p.m. UK time.

Website:

www.bcs.org/heq

Post:

BCS, The Chartered Institute for IT
First Floor, Block D, North Star House, North Star Avenue,
Swindon SN2 1FA, United Kingdom