

BCS Higher Education Qualifications

Diploma in IT

Software Engineering Syllabus

Version 3.0

December 2016

This is a United Kingdom government regulated qualification which is administered and approved 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	3
5. Prior Knowledge Expected	3
6. Format and Duration of the Examination	4
7. Syllabus Detail	4
8. Recommended Reading List	5
9. Contact Points	5

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		Released
Version 2.0	June 2016	Re-formatted with syllabus numbering – no change to content
Version 3.0	Dec 2016	Regulated statement added.

2. Rationale

This module is for those who wish to understand an introduction to Software Engineering and the skills necessary to create software products and applications that are effective, maintainable and value-for-money.

3. Aims

- To apply the knowledge of a disciplined approach to the development of software and to the management of the software product lifecycle

4. Objectives

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

- Explain the background of the software crisis and the need for an engineering approach
- Appreciate the distinction between software programming and an engineering approach to the development of a software product
- Create models of software data and processes using object oriented modelling approaches such as the UML
- Describe and evaluate software tools and technology to enhance productivity and quality of software development
- Demonstrate skills of software documentation, quality assurance and evaluation, and testing as part of software development
- Describe development contexts and can apply estimation methods for planning these contexts

5. Prior Knowledge Expected

Candidates must have achieved the Certificate in IT or have an appropriate exemption to be entered for the Diploma in IT. Candidates are expected to have used a variety of programming languages and to be familiar with a range of software development technologies.

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 SOFTWARE ENGINEERING	1.1	The nature of software
	1.2	Theoretical models
	1.3	The software crisis
	1.4	The cost of maintenance
	1.5	The cost of quality
2 SOFTWARE ENGINEERING KEY PRACTICES	2.1	The multidisciplinary nature of software design
	2.2	Team work
	2.3	Productivity
	2.4	Testing
	2.5	Product maintenance
	2.6	Software product life cycle
3 SOFTWARE DEVELOPMENT MODELS AND METHODS	3.1	Design principles (transparency, separation of concerns, abstraction, modularity and development by incremental methods)
	3.2	OO notation for describing software components and architecture
	3.3	OO approaches such as the UML modelling of use cases for a logical/end-user view, system components and architecture for the development view, behavior and deployment for process and physical implementation views
4 VALIDATION, VERIFICATION, AND TESTING	4.1	Product and process visibility
	4.2	Traceability in software systems and processes
5. SOFTWARE ENGINEERING TOOLS AND ENVIRONMENTS	5.1	Upper and lower CASE tools
	5.2	Role of the repository for supporting incremental development
	5.3	Software reuse and evolution
6. PROJECT MANAGEMENT	6.1	Project estimating and project planning
	6.2	Management and maintenance of software products in the consumer marketplace
	6.3	Total cost of system ownership
	6.4	Software life-cycle cost modelling
	6.5	Project development cost modelling
	6.6	Project and product risk management

8. Recommended Reading List

Software Engineering 1	ISBN 10	ISBN 13
Primary Texts		
• Pressman R.S. and Ince D., Software Engineering: A Practitioner's Approach, McGraw Hill (7th Ed), 2009/10	0073375977 0071267824	978-0073375977 978-0071267823
• Ian Sommerville, Software Engineering: International Version Edition 9, Addison Wesley, 2010	0137053460 0137035152	978-0137053469 978-0137035151
Other Texts		
• Fredrick P. Brooks, Jr., The Mythical Man-Month, Addison-Wesley, 1995 (paperback edition)	0201835959	978-0201835953
• Humphrey W., A Discipline for Software Engineering. Addison Wesley, 1995	0201546108	978-0201546101
• Endres A. and Rombach, D., A Handbook of Software and Systems Engineering. Pearson (Addison Wesley), 2003	0321154207	978-0321154200
• Conallen J., Building Web Applications With UML, Addison-Wesley, 2nd Ed., 2002. (paperback)	0201730383	978-0201730388
Other Reading		
• Gilb T., Evolutionary Project Manager's Handbook at http://www.ida.liu.se/~TDDDB02/pkval01vt/EvoBook.pdf		
• Kruchten P., 1995, Architectural Blueprints - the '4 +1 ' View Model of Software Architecture, IEEE Software 12,6 (Nov 1995) pp42- 50 and http://www.rational.com/media/whitepapers/Pbk4p1.pdf		
• Kruchten P., 2002, A software development process for a team of one at http://www.nada.kth.se/~karlm/light_sw_process.pdf		
Cross Reference Guide to Primary Texts		
• Section 1. Sommerville Chs 1-2, Pressman Ch 1		
• Section 2. Sommerville Chs 1-2		
• Section 3. Sommerville Ch12, Pressman Chs 8 and 12		
• Section 4. Sommerville Chs 19-20, Pressman Chs 17-21		
• Section 5. Sommerville Ch3, Pressman Chs 22-23		
• Section 6. Sommerville Ch3, Chs 22-24, Pressman Chs 2-4.		
In addition, the Other Texts illuminate models, methods, and management of diverse systems and show how they are 'powered' by engineering-quality software		

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/heg

Post:

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