

BCS Higher Education Qualifications

Level 6 Software Engineering 2 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	4
6. Format and Duration of the Examination	4
7. Syllabus Detail	5
8. Recommended Reading List	6
9. Contact Points	7

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	March 2014	Released
Version 2.0	June 2016	Re-formatted with syllabus numbering – no change to content
Version 3.0	Dec 2016	Regulation Statement Added

2. Rationale

To develop, maintain in operation, and evolve software systems that are of high quality it is imperative the Computing professionals understand software development and its evolution as an engineering discipline. This understanding must be based on the theoretical foundations of software engineering and demonstrated through a critical application of software engineering theory to real world problems in practical applications.

3. Aims

- To gain a thorough understanding of the relationship between the processes used in the engineering of software systems, the software products produced, and of the theory, laws, and models, that provide a rational basis for the practice of software engineering.

4. Objectives

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

- Demonstrate a critical understanding of software systems engineering theory in the form of laws and models, and of associated methods, tools, and techniques;
- Critically apply the above to practical situations found throughout the Software Life Cycle from software requirements engineering, system specification, design, implementation, validation and verification, through to maintenance and evolution; and to recognize the potential for software reuse throughout the life cycle;
- Appreciate the importance of software project management, software economics, estimation and planning as well as the management of software project teams and their productivity;
- Appreciate the role of empirical studies in Software Engineering and the role of the Goal/Question/Metric in software engineering experiments, especially in the context of software quality improvement experiments with respect to both process and product improvements;
- Discuss critically recent advances in software engineering: component-based software engineering, model driven software development, the Agile paradigm including Extreme programming, software product lines engineering and community based software development such as Free and Open Source Software development;
- Appraise advanced software concepts and their applicability in practice: Design Patterns and Frameworks, Software Refactoring techniques, Software Architectural analysis, Software as a Service.

5. Prior Knowledge Expected

The learner must have achieved the Diploma in IT or have an appropriate exemption to be entered for the Professional Graduate Diploma in IT.

Candidates should be familiar with the material covered in the Certificate syllabus and the Software Engineering 1 Diploma syllabus and should have knowledge of Object Oriented software development, both design and implementation and practical experience of developing software systems.

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 three-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 ANALYSIS AND IMPROVEMENT OF SOFTWARE PROCESSES	1.1	Software Process Improvement (Such as: CMM, CMM-I, SPICE(ISO 15504); ISO and IEEE Software Engineering standards e.g. ISO 9001:2000/2008, ISO/IEC 12207, ISO/IEC 90003:2004, IEEE 1012:1998)
	1.2	Various Software Life Cycle Models (Waterfall, V-model, Prototyping, Spiral Model, Incremental Development, Evolutionary Development, Agile models including Extreme programming)
	1.3	Software Requirements Engineering including requirements management
	1.4	Software Management: project management, cost estimation, planning, personnel management, team building
	1.5	Software Evolution: Lehman's Laws of Software Evolution and related models and studies
2 ANALYSIS AND IMPROVEMENT OF SOFTWARE PRODUCTS	2.1	Software Maintenance and the related types of maintenance, Impact Analysis, Reverse and Re-engineering of software
	2.2	Software Architecture and software re-factoring, Architectural styles, examples, and applications, Architectural models, Model-driven development
	2.3	Software metrics, software complexity measures, measures of software coupling and cohesion, models and associated measures of software quality
	2.4	Empirical Software Engineering and Basili's Goal/Question/Metric
3 ADVANCED TOPICS IN SOFTWARE ENGINEERING	3.1	Software reuse, Component based software engineering, Software product lines, Design patterns
	3.2	Software as a service, including web services and dynamic reconfiguration of software systems
	3.3	Open Source Software Engineering
	3.4	Advanced use of UML including Object-Constraint Language and use of Assertions, pre- and post-conditions

8. Recommended Reading List

Software Engineering 2	ISBN 10	ISBN 13
Primary Texts		
• Ian Sommerville, Software Engineering: International Version Edition 9, Addison Wesley, 2010.	0137053460 0137035152	978-0137053469 978-0137035151
• Roger S. Pressman, Software Engineering: A Practitioner's Approach, McGraw-Hill, 7th Edition, 2009.	0071267824	978-0071267823
Other Texts		
• Fredrick P. Brooks, Jr., The Mythical Man-Month, Addison-Wesley, 1995 (paperback edition)	0201835959	978-0201835953
• Endres A. and Rombach, D., A Handbook of Software and Systems Engineering. Pearson (Addison Wesley), 2003.	0321154207	978-0321154200
• Norman Fenton and Shari Pfleeger, Software metrics (2nd ed.): a rigorous and practical approach, PWS Publishing, 1997.	0534956009 0534954251	978-0534956004 978-0534954253
• Erich Gamma, Richard Helm, Ralph Johnson and John Vlissides, DesignPatterns: Elements of Reusable Object-Oriented Software, 1994.	0201633612	978-0201633610
• Joshua Kerievsky, Refactoring to Patterns, Addison Wesley, 2004.	0321213351	978-0321213358
• Rini Van Solingen and Egon Berghout, The Goal/Question/Metric Method, McGraw-Hill, 1999.	0077095537	978-0077095536
• Jeffrey S.Poulin, Measuring Software Reuse: Principles, Practices, and Economic Models, Addison-Wesley, Reading, MA, 1997.	0201634139	978-0201634136
• Watts Humphrey, Managing the Software Process (SEI), Addison- Wesley, 1989.	0201180952	978-0201180954
• Diomidis Spinellis, Code Quality: the Open Source Perspective, Addison- Wesley, 2006.	0201546108 8131703800	978-0201546101 978-8131703809
• Kent Beck, Extreme Programming Explained: Embrace Change, Addison-Wesley, 2nd Edition, 2004.	0321166078	978-0321166074
• Alistair Cockburn, Agile Software Development, Addison-Wesley, 2nd Edition, 2006.	0321482751	978-0321482754
• Len Bass, Paul Clements, Rick Kazman, Software Architecture in Practice (SEI), Addison-Wesley, 3rd Edition, 2012.	0321815734	978-0321815736
• Hull M. E. C., Jackson K., Dick A. J. J., Requirements Engineering, 3rd edition, Springer, 2010.	1849964041	978-1849964043
• James Rumbaugh, Ivar Jacobson, Grady Booch, The Unified Modeling Language Reference Manual, Addison-Wesley, 2nd Edition, 2004.	032171895X	978-0321718952
• Mens, Tom; Demeyer, Serge (Eds.), Software Evolution, Springer- Verlag, 2008.	3540764399	978-3540764397

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