

Master of Science in Computer Science

a.a. 2017-18

year	code	course name	ECTS	type	semester	educational activity type	ECTS	hours	faculty
2	F1801Q116	Software Evolution and Reverse Engineering	6	elective	First semester	lecture	5	40	Arcelli Fontana
						practice exercise	1	10	

Professors' CV: <http://www.unimib.it/go/176181440>

Contents Introduction to the principal problematics of reverse engineering, software evolution and program comprehension. Deep study of some topics with different tools experimentations.

Textbooks Slides, papers, online books, survey and tutorial, Master and PhD thesis

Course objectives The student will learn all the principal techniques used to support software evolution and reverse engineering. The student will be able to use different tools useful for reverse engineering, program comprehension and software maintainability.

Prerequisites Knowledge of Java language.

Teaching methods Lessons, exercitations, students experimentatons of tools with oral presentation.

Learning assessments Final exam with a project or experimentation of some tools of reverse engineering.

Extended Syllabus

1 Introduton to Software evolution and Reverse Engineering. Legacy systems. Software comprehension and maintainability: principal problems.

2 Techniques and tools for Reverse Engineering and Systems Integration. Introduction and application to model-driven reverse engineering.

3 Object-oriented patterns for reverse engineering;

4 Software quality metrics and software quality assessment. Application Portfolio Management: problems, tools, techniques and metrics.

5 Modernization of legacy systems: Migration of legacy systems towards SOA architecture. Unified approach for reverse engineering of data and software.

Master of Science in Computer Science a.a. 2017-18

6 Tools and techniques for design pattern detection in the code. Data mining techniques for design pattern detection. Tools for software architecture reconstruction.

7 Antipattern and code smell detection in the code, their refactoring. Impact of refactoring techniques on the code quality. Correlation analysis among code smells and metrics.

8 Techniques of hacking, decompiling and code obfuscation. Static and dynamic analysis for reverse engineering.