

Course outline

Course name	FORMAL SOFTWARE SPECIFICATION METHODS
Course code	BCT 313
Semester	Year 3 Bsc. Computer Technology
Lecturer/email address	Mwangi karanja jkuatnotes@gmail.com

1. Course objectives

Formal methods is a term that refers to a diverse collection of techniques, with strong mathematical foundations, that are used to provide assurance about the correctness of systems. The course aims to offer a sound background to students in software engineering with languages and methods for formal specification, development and verification.

Prerequisites: knowledge in mathematical logic, discrete structures and in software engineering

2. Course Content

Software engineering concepts. Propositional and predicate calculus . Higher order logic. Advanced data structures and algorithms. Higher order theoretical tools. Abstraction, design. Formal methods such as VDM. Use of formal development tools.

3. Detailed Course content

Introduction	General discussion of specification and formality in the software development process; Introduction to Formal methods, Software Correctness, limitations of testing, formal methods (relation with testing), Difficulties in Formal methods, Proof Automation.
Propositional and first order logic	Logic in Computer Science, formalization, (Syntax and semantics), calculus, syntax and semantics of propositional logic, first order atomic formulas, first order semantics and first order models, temporal logic, semantics of temporal logic, safety and liveness properties,
Formal software specification methods	Introduction, Algebraic Specification (Abstract Data Types) Specification and requirements, Specification languages and Applications, effectiveness of specifications.
Quality Assurance with FM	Correctness of properties, theorem proving and property driven approaches.

4. Teaching methodology

The activities will involve lectures, research assignments, discussions, reflections and presentations. They will also be involved in facilitating discussions. Group presentations are emphasized to enhance team work where each group is expected to present to class its work .

5. Course text and recommended reading

Recommended Reading material: Author(s), Title, Edition, Publisher, Year of Publication etc. (Text books, Reference Books, Journals, Reports, Websites etc.)	
1.	A. Diller, <i>Z An Introduction to Formal Methods</i> (2nd ed.), Wiley, 1994.
2.	Marc Frappier and Henri Habrias (Eds) <i>Software Specification Methods: An Overview Using a Case Study</i> . Publisher: Springer. (Edition 2006)
3.	J.M. Spivey, <i>The Z Notation</i> , Prentice Hall, 1992 available online at[http://spivey.oriel.ox.ac.uk/7Emike/zrm]
4.	J.P. Bowen and M.G. Hinchey, "Ten Commandments Revisited: A Ten-Year Perspective on the Industrial Application of Formal Methods," Proc. 10th Workshop on Formal Methods for Industrial Critical Systems (FMICS 2005), ACM Press, 2005, pp. 8-16.
5.	B. Potter, J. Sinclair, D. Till, <i>An Introduction to Formal Specification and Z</i> , 2nd ed., Prentice Hall 1996.

6. Course evaluation

- a. Continuous assessment tests 30% and Final semester exams 70%.

7. Ground rules

- a. Late assignments will not be accepted.
- b. Group work must be completed in time as required.
- c. Mobile phones must be switched off in class.