**Title of Subject : Formal Methods in Software Engineering (SW-418)**

**Discipline :** Software Engineering (7th Semester)

**Effective :** F16 Batch & onwards

**Pre-requisite :** Discrete Structures

**Assessment :** Theory**:** 20% Sessional, 80% Written Semester Examination

## (20% Mid, 60% Final)

**Credit Hours :** 03 + 0 **Marks:** 100

**Minimum Contact Hours:** 45

# Specific Objectives of course:

* To teach knowledge and skills in, and judgment about, two important styles of formal methods for reasoning about software: model checking and deductive verification.

**COURSE LEARNING OUTCOMES:**

Upon successful completion of the course, the student will be able to:

|  |  |  |  |
| --- | --- | --- | --- |
| **CLOs** | **Description** | **Taxonomy level** | **PLO** |
| 1 | Identify what can and what cannot be expressed by certain specification/modeling formalisms | C1 | 1 |
| 2 | Write formal specifications of object-oriented system units, using the concepts of method contracts and class invariants. | C3 | 3 |
| 3 | Describe how the connection between programs and formal specifications can be represented in a program logic. | C4 | 5 |

**PROGRAM LEARNING OUTCOMES (PLOs):**

The course is designed so that students will achieve the following PLOs:

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| 1 | Engineering Knowledge: | ☑ | 7 | Environment and Sustainability: | ☐ |
| 2 | Problem Analysis: | ☐ | 8 | Ethics: | ☐ |
| 3 | Design/Development of Solutions: | ☑ | 9 | Individual and Team Work: | ☐ |
| 4 | Investigation: | ☐ | 10 | Communication: | ☐ |
| 5 | Modern Tool Usage: | ☑ | 11 | Project Management: | ☑ |
| 6 | The Engineer and Society: | ☐ | 12 | Lifelong Learning: | ☐ |

**Course outline:**

**INTRODUCTION TO FORMAL METHODS**

# Introduction to the use of mathematical models for specification and validation, Finite state machine models, models of concurrent systems, verification of models, and limitations.

# MODEL CHECKING

# Analyzing well-formedness (e.g. completeness, consistency, robustness, etc.), Analyzing correctness (e.g. static analysis, simulation, model checking, etc.).

# FORMAL LANGUAGES

# Formal analysis, An introduction to VDM-SL, Sets, Sequences, Composite objects, Maps, VDM-SL, Comparative Formal Methods, Proofs. Introduction to Object Constraint Language (OCL) and use of class variants, pre and post conditions on operations and model verification

# FORMAL VERIFICATION METHODS

# Unit level specification language for Java programs, a logic for verification of Java programs, verification of Java programs, in the sense that the implementation of a unit fulfils the specification.

# Recommended Books:

1. Modern Formal Methods and Applications, Hossam A. Gabbar, Springer-Verlag Latest Edition.
2. Formal Software Development: From VDM to Java, Charatan, Quentin, and Aaron Kans. Palgrave Macmillan, Latest Edition
3. Object Modeling with the OCL: The Rationale behind the Object Constraint Language, edited by Tony Clark, Jos Warmer, Latest Edition

|  |  |  |
| --- | --- | --- |
| **Approval:** |  | |
| **Board of Studies:** | **Resolution No. 02** | **Dated: 29-08-2019** |
| **Board of Faculty:** | **Resolution No. 01** | **Dated: 07-10-2019** |
| **Academic Council:** | **Resolution No. 96.10** | **Dated: 07-10-2019** |