COURESE DESCRIPTION

<table>
<thead>
<tr>
<th>Dept. Number</th>
<th>Course Title</th>
<th>Credit Hours</th>
</tr>
</thead>
<tbody>
<tr>
<td>CS 5371</td>
<td>Software Safety and Risk Analysis (Prescribed Elective)</td>
<td>3</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Instruction Type</th>
<th>Course Coordinator</th>
<th>Semester Hours</th>
</tr>
</thead>
<tbody>
<tr>
<td>Lecture</td>
<td>Salamah Salamah</td>
<td>45</td>
</tr>
</tbody>
</table>

CATALOG DESCRIPTION

Principles of software development for safety and mission critical systems. Topics include safety-related analysis, specification, design, implementation, and maintenance techniques; survey of programming language and operating system issues for implementing safety-related software; safety requirements, hazard and risk analyses, fault tolerance, basics of software reliability, and issues of verification, validation, and certification; models for safety in a distributed system; safety standards and guidelines across application domain and selected tools supporting safety assurance of software components.

TEXT BOOK

Available online at [http://mitpress.mit.edu/books/engineering-safer-world](http://mitpress.mit.edu/books/engineering-safer-world)

SUPPLEMENTAL MATERIAL


COURSE OUTCOMES

Upon completion of the course, students will be able to do the following:
1. Describe concepts of integrity, dependability and fail-safe operations, as well as basic techniques of hazard analysis;
2. Analyze a system with respect to safety, including cause of software failures, as a part of the high integrity software lifecycle;
3. Identify the methods of testing for safety and related verification and validation issues;
4. Determine types of faults and determine best approach to fault tolerance;
5. Apply software quality safety standards and guidelines;
6. Describe issues related to software safety certification;
7. Apply models for controlling redundant computations in a distributed system;
8. Work on a team to research topics in the development of safe and reliable systems, participate in laboratory experiments with state-of-the-art tools, and produce a safety risk analysis report.

PREREQUISITE BY TOPIC

None

MAJOR TOPICS COVERED IN THE COURSE

- Software Safety Concepts and Criteria
- Hazard and Risk Analysis
- System and Software Reliability
- Safety Critical Systems Development
- Software Fault Tolerance
- Validation and Verification, and Formal Methods for Software Safety
- Safety Critical Hardware and Software
- Quality, Safety Standards and Certification
- Issues of Programming Implementation for Software Safety
- Software Safety Case Studies
- Safety Engineering Ethics