



Centro de Investigación y de Estudios Avanzados  
Del Instituto Politécnico Nacional  
Secretaría Académica

Registro de Cursos o Asignaturas

<b>Nombre Completo del Programa de Posgrado</b>		<b>Maestría en Ciencias en Ingeniería Eléctrica</b>		
<b>Nombre Completo del Curso</b>		<b>Tópicos Selectos de Matemáticas I</b>		
<b>Tipo de Curso</b>		<b>Electivo</b>	<b>Créditos</b>	<b>8</b>
<b>Número de horas</b>	<b>Teóricas:</b>	<b>60</b>	<b>Prácticas:</b>	<b>0</b>
		Presenciales		No presenciales
<b>Profesores que impartirán el curso</b>				
<b>Raúl Ernesto González Torres</b>				
<b>Objetivos del curso:</b>	<b>General</b>			
	<b>Específicos</b>			
<b>Contenidos temáticos</b>				
<b>1. Lógica Temporal Proposicional CTL</b>				
1.1. Lógica Computacional Arbórea CTL. Sintaxis y Semántica				
1.2. El Algoritmo de Etiquetado. Aplicación a la Verificación Formal de Sistemas				
1.3. Un Procedimiento de Decisión Para la Lógica CTL				
<b>2. Lógica Temporal Proposicional Lineal LTLP y Autómatas de Büchi</b>				
2.1. Sintaxis y Semántica de la LTLP				
2.2. Autómatas de Büchi. ABEs y ABGEs				
2.3. Un Procedimiento de Decisión Para la LTLP usando ABGEs				
2.4. Recorte del ABGE de una Fórmula. El Algoritmo SAT				
<b>3. Comprobación de Modelos Explícita con LTLP</b>				
3.1. Qué es la Comprobación de Modelos (o Model Checking)				
3.2. Comprobación de Modelos Explícita con LTLP y ABGEs				
3.3. Clases Importantes de Propiedades a Comprobar. Seguridad y Vivacidad				
3.4. El Problema de Explosión de Estados. Técnicas de Reducción del Espacio de Estados				
3.5. Spin, una Herramienta Para Comprobación de Modelos Explícita de Software				
<b>4. Comprobación de Modelos Simbólica Usando BDDs</b>				
4.1. Comprobación de Modelo Simbólica Para CTL				
4.2. Estructuras de Kripke (EKs). Bisimulación e Isomorfismo y Producto de EKs				
4.3. Representación Simbólica con BDDs de una EK				
4.4. El Probador de una Fórmula de la LTLP				
4.5. Correctud de la Construcción				
4.6. SMV y NuSMV, Herramientas Para Comprobación de Modelos Simbólica				
<b>5. Introducción a la Prueba de Software (Software Testing)</b>				
5.1. Conceptos Básicos de la Prueba de Software				
5.2. Ejemplos Ilustrativos. El Problema del Triángulo, el Problema de la Comisión, etc.				

5.3. Repaso de Matemáticas Discretas Para Testers	
5.4. Pruebas Unitarias	
5.5. Pruebas Basadas en el Ciclo de Vida del Software	
5.6. Pruebas de Integración	
5.7. Pruebas de Sistema	
<b>6. Pruebas Basadas en Modelos</b>	
6.1. Panorama General de la Prueba Basada en Modelos (PBM)	
6.2. Diagramas de Flujo. Ventajas y Limitaciones	
6.3. Tablas de Decisión. Ventajas y Limitaciones	
6.4. Máquinas de Estados Finitas. Ventajas y Limitaciones	
6.5. Pruebas a Partir de Modelos de UML Basados en Transiciones	
6.6. Herramientas Para la PBM	
<b>7. Verificación Formal y Prueba de Software</b>	
7.1. Verificación vs. Pruebas de Software	
7.2. Importancia del Empleo de Ambas Técnicas Para la Validación de Sistemas	
7.3. Cómo Usar la Comprobación de Modelos en la Generación de Casos de Prueba	
<b>Bibliografía</b>	
1. Logic in Computer Science: Modelling and Reasoning About Systems, M. R. A. Huth y M. D. Ryan, Cambridge University Press, 2000	
2. Software Testing: A Craftsman's Approach (4th Edition), Paul C. Jorgensen, CRC Press, 2014	
3. Model Checking (2nd Edition), por E. Clarke, D. Peled et al., MIT Press, 2018	
4. Temporal Logic and State Systems, Fred Kröger y Stephan Merz, Springer-Verlag. 2008	
5. Software Testing and Quality Assurance: Theory and Practice, por Kshirasagar Naik y Priyadarshi Tripathy, John Wiley & Sons, Inc., 2008	
6. Model-Based Testing of Reactive Systems: Advanced Lectures, Manfred Broy, et al. (eds.), LNCS 3472, Springer-Verlag, 2005	
7. Practical Model-Based Testing: A Tools Approach, por Mark Utting y Bruno Leguard, Elsevier Inc., 2007	
8. Principles of the Spin Model Checker, Mordechai Ben-Ari, Springer-Verlag, 2008	
9. Temporal Verification of Reactive Systems, Z. Manna y A. Pnueli, Springer-Verlag, New York, 1995	
10. An Introduction to Binary Decision Diagrams: Tutorial por Henrik Reif Andersen, en: <a href="http://www.itu.dk/people/hra/notes-index.html">http://www.itu.dk/people/hra/notes-index.html</a>	
11. The Art of Software Testing (2nd Edition), Glenford J. Myers, John Wiley & Sons, 2004	
12. Infinite Words (Capítulo 1), D. Perrin y Jean-Eric Pin, versión preliminar, disponible en la web: <a href="http://www.liafa.jussieu.fr/~jep/Resumes/InfiniteWords.htm">www.liafa.jussieu.fr/~jep/Resumes/InfiniteWords.htm</a>	
<b>Criterios de evaluación</b>	
Examen 1	0%
Examen 2	0%
Examen 3	0%
Proyecto 1	0%
Proyecto 2	0%
Apreciación (puntualidad, participación, calidad en la entrega de trabajos, etc.)	0%
Total	100%

**Contribución del curso al perfil de egreso del programa**

**Conocimientos:**

**Habilidades:**

**Actitudes y valores:**