

NOMBRE DE LA ASIGNATURA:			
MODELACIÓN Y SIMULACIÓN DE SISTEMAS			
OBJETIVO GENERAL Y PARTICULARES DE LA ASIGNATURA:			
General: Este curso complementa los cursos de sistemas dinámicos y control, al capacitar a los estudiantes en el uso de lenguajes de programación orientado a objetos basado en ecuaciones con el alcance de desarrollar de manera sencilla modelos de sistemas dinámicos eventualmente muy complejos. Al finalizar el curso el alumno conocerá los conceptos y técnicas generales de entornos de programación orientado a objetos y lenguaje de modelado multi-dominio para modelado de sistemas complejos orientado a componentes. Así mismo, el alumno será capaz de utilizar los entornos de programación para los lenguajes C++, Modelica, SciLab, Prolog, MatLab y Simulink con el nivel suficiente, para realizar simulaciones de sistemas ciber-físicos, mecatrónicos u otros sistemas multi-dominio. Particulares: Modelación de sistemas físicos, y el lenguaje de programación en lenguaje C con el nivel suficiente para realizar programas de simulación y/o aplicación a dispositivos mecatrónicos.			
Duración del ciclo: 24 SESIONES DE 2.5 HORAS	Horas totales con docente: 60 HORAS	Horas totales independientes: 40 HORAS	Instalaciones: AULA, EQUIPO AUDIOVISUAL
CICLO, ÁREA O MÓDULO: OPCIONAL		CRÉDITOS: 4	CLAVE: COC-8
TEMAS Y SUBTEMAS:			
Capítulo 1: Introducción a la Modelación y Simulación 1.1 Sistemas, experimentos y modelo • Niveles de conocimiento de los sistemas • Sistemas naturales y artificiales • Experimentos • Para qué es un modelo 1.2 Tipos de modelos matemáticos y simuladores • Clasificaciones de los modelos matemáticos • Modelos de tiempo discreto, eventos discretos e híbridos • Ejemplos de modelos: Sistemas físicos, económicos y ecológicos • Simuladores y paradigmas de programación • Pros y contras de la simulación 1.3 Modelos para sistemas dinámicos y señales • Ecuaciones diferenciales • Forma estándar del modelo en el espacio de estados • Sistemas lineales y no-lineales • Modelos dinámicos y señales de entrada y salida • Señales de entrada: Escalón, Rampa, pulso y sinusoidal • Perturbación de sistemas de EDO de primer orden • Perturbación de sistemas de EDO de segundo orden 1.6 Modelos de sistemas • Sistemas eléctricos • Sistemas mecánicos traslacionales y rotacionales • Sistemas hidráulicos • Sistemas térmicos • Bloques constructores Capítulo 2: Introducción a la programación orientada a objetos			

2.1 Modelado matemático orientado a objetos

- Qué es la programación orientada a objetos
- Pilares de la programación orientada a objetos

2.2 Estructura básica de un programa

- Variables, tipos predefinidos
- Variabilidad de las variables y sus atributos
- Comentarios
- Constantes
- Valores iniciales
- Arrays y cadenas.

2.3 Construcciones Algorítmicas

- Secciones de Algoritmos y Sentencias de Asignamiento
- Sentencias
- Funciones
- Sobrecarga de Operadores y Números Complejos
- Funciones Externas
- Algoritmos Vistos como Funciones

2.4 Definición de clases e Instancias

- Creación de Instancias
- Inicialización
- Clases Especializadas
- Reutilización de Clases a través de Modificaciones
- Clases Predefinidas y Atributos
- Polimorfismo

2.5 Clases Genéricas

- Parámetros de Clase que son Instancias
- Parámetros de Clase que son Tipos

2.6 Clases Parciales

- Reutilización de las Clases Parciales

2.7 Herencia

- Herencia de Ecuaciones
- Herencia Múltiple
- Procesado de los Elementos de la Declaración y Uso Previo a la Declaración
- Orden de la Declaración de las Cláusulas extends
- El Ejemplo del Alunizaje Usando Herencia

2.8 Modelación por Componentes de Software

- Componentes
- Diagramas de Conexiones
- Conectores y Clases connector
- Conexiones
- Conexiones Implícitas con las palabras clave Inner/Outer
- Conectores Expandibles para Buses de Información
- Conectores tipo Stream

Capítulo 3: Simulación de Modelos en Tiempo Continuo

3.1 Modelado físico no causal

- Causalidad computacional
- Principio de flujo de datos síncrono
- Variables de Across Through
- Principio de conservación de potencia
- Ecuaciones triviales y variables alias

3.2 Programación de modelos físico atómico y conocimiento de la GUI

- Librerías estándares
- Péndulo amortiguado

- Nivel de tanques
- Ecuación de Lotka-Volterra
- Sistemas masas acopladas-resorte-fricción-fuerza externa
- Circuito rectificador con diodo ideal y no ideal
- Transferencia de calor radial
- Vibración longitudinal y transferencia de calor en una barra

3.3 Diseño y uso de librerías de componentes

- Péndulo amortiguado
 - Componentes mecánicos
 - Comparación con modelo atómico
- Sistemas masa acoplada-resorte-fuerza externa
 - Componentes mecánicas
 - Comparación con modelo atómico
- Nivel de tanques
 - Componentes hidráulicas
 - Comparación con modelo atómico
- Circuito rectificador
 - Componentes eléctricas
 - Comparación con modelo atómico

3.4 Modelo de componentes de software

- Componentes, diagramas de conexiones, conectores y clases connector
- Class conector
- Componente Interface
- Partial Class
- Diseño de componentes
- Arquitectura de la librería
- Ejemplo de desarrollo de librería: Circuito eléctrico rectificador
 - Conjunto de ecuaciones
 - Conectores
 - Clases parciales y componentes
 - Paquetes

3.5 Operaciones adicionales usando las librerías

- Acceder a elementos de una librería
- Clases reemplazables, instancias y herencias
- Conectores, tipos, bloques, funciones y paquetes reemplazables
- Ejemplo: Vibración longitudinal de una barra
 - Clase record
 - Instancias de modelos de sistemas físicos
- Ejemplo: Transferencia de calor longitudinal en una barra
 - Uso de arrays de componentes
- Ejemplo: Control del nivel y temperatura en un tanque
 - Eventos discretos, nivel variable en el tiempo
- Ejemplo: Disipación de calor en un circuito
 - Constructores inner/outer

Capítulo 4: Causalidad computacional

4.1 Variables de desconocidas (unknown) y variables ficticias (dummy)

4.2 Estructura computacional del modelo

4.3 Singularidad estructural

4.4 Algoritmo de particionado

4.5 Sistemas sobredeterminados y subdeterminados

4.6 Ejemplo: Simulación de un circuito eléctrico

4.7 Sistemas DAE de alto índice

4.8 Dificultades asociadas a las soluciones numéricas de alto índice

4.9 Discusión de los principios de el algoritmo de integración DASSL

Capítulo 5: Modelado y Simulación de Sistemas Híbridos

5.1 Especificación de sistemas híbridos y el formalismo OHM

- Tipos de eventos
 - Continuos en el tiempo
 - Discretos en el tiempo
 - Detección y manejo de eventos
 - Determinación de eventos y eventos simultáneos
 - Ejecución de eventos

5.2 Especificaciones de los modelos híbridos en la plataforma de simulación

- Cláusula When
- Sentencia If
- Función reinit
- Ejemplo: Pelota rebotando
- Ejemplo: Tanque de almacenamiento con válvula de drene
- Ejemplo: Dos tanques y sistemas de válvulas

5.3 Modelos con estructura variable

- Switch ideal
- Ejemplo: Sistema hidráulico con tubo de alivio
- Ejemplo: Tubo de desagüe
- Ejemplo: Tanques conectados en paralelo
- Ejemplo: interruptor resistivo

5.5 Inicialización y experimentación con modelos híbridos

- Planteando el problema de inicialización
- Variables continuas en el tiempo
- Ejemplo: El péndulo plano simple
- Variables discretas en el tiempo
- Bucle de control
- Modelo inicial

RESULTADOS DEL APRENDIZAJE:

1. El estudiante aprenderá modelado matemático de sistemas multidominio
2. El estudiante aprenderá a modelar sistemas físicos con lenguajes de programación orientados a objetos basados en ecuaciones.
3. El estudiante aprenderá a desarrollar librerías basadas en componentes y conexiones
4. El estudiante aprenderá las bases de la programación no causal y fundamentos de los algoritmos de las plataformas de simulación
5. El estudiante aprenderá simulación de eventos discretos en el tiempo

ACTIVIDADES DE APRENDIZAJE BAJO CONDUCCIÓN DEL DOCENTE:

1. Exposición oral de la materia
2. Presentación audiovisual
3. Simulaciones en clase
4. Prácticas de simulación

ACTIVIDADES DE APRENDIZAJE INDEPENDIENTE:
1. Estudio independiente 2. Desarrollo de prácticas 3. Trabajos y tareas fuera de clase.
MEDIOS Y CRITERIOS DE EVALUACIÓN:
Escala de evaluación 0 – 10. Calificación mínima aprobatoria: 8.0 • Tareas y ejercicios propuestos durante el curso 20 % • Evaluaciones parciales (2 evaluaciones) 40% • Proyectos parciales 40%
RECURSOS Y MATERIALES
Computadora personal, Software de simulación OpenModelica, SciLab-Python, Simulink-MatLab, otros Compiladores.
BIBLIOGRAFÍA
1. Fritzson, P., Pop, A., Aronsson, P., Akhvlediani, D., Bachmann, B., Baluta, V., ... & Zackrisson, B. (2023). OpenModelica Users Guide. Release v1.20.0-1-g2faf7aa0ea. 2. Fritzson, P. (2014). Principles of object-oriented modeling and simulation with Modelica 3.3: a cyber-physical approach. John Wiley & Sons. 3. Fritzson, P., & Engelson, V. (1998). Modelica—A unified object-oriented language for system modeling and simulation. In ECOOP'98—Object-Oriented Programming: 12th European Conference Brussels, Belgium, July 20–24, 1998 Proceedings 12 (pp. 67-90). Springer Berlin Heidelberg. 4. Urquía, A., Martín-Villalba, C., & del Rosal, J. (2018). Modeling and simulation in Engineering using Modelica. Madrid, Spain, UNED Editorial. 5. Moraleda, A. U., & Villalba, C. M. (2016). Métodos de simulación y modelado. UNED-Universidad Nacional de Educación a Distancia. 6. Ljung, L., & Glad, T. (1994). Modeling of dynamic systems. Prentice-Hall, Inc. 7. Wellstead, P. E. (1979). Introduction to physical system modelling (Vol. 4). London: Academic Press. 8. Karnopp, D. C., Margolis, D. L., & Rosenberg, R. C. (1997). System dynamics: modeling and simulation of mechatronic systems, 2000. ed: John Wiley & Sons, NY.
REQUISITOS ACADÉMICOS DEL PERSONAL DOCENTE
Se requiere de profesores con estudios de posgrado y experiencia en campo, investigadores dentro del área de afín, interesados en la transmisión de sus experiencias.