

Contents

```
json {  "course_title": "Simulación de Sistemas",  "modules": [    {      "module_title":  
"Introducción a la Simulación y Simulación de Monte Carlo",      "module_num": "1",  
"objectives": [        "Entender el proceso de modelado y análisis de sistemas.",        "Aplicar  
técnicas de muestreo de distribuciones de probabilidad para resolver problemas de decisión.",  
"Comprender los pasos de la simulación y el concepto de aleatoriedad."      ],      "content_outline":  
"Introducción a la Simulación. Pasos de la Simulación. Aleatoriedad. Simulación de Monte  
Carlo.",      "class_notes": {        "class_num": "1",        "introduction": "Esta  
clase introduce la simulación como herramienta para modelar y analizar sistemas complejos,  
especialmente aquellos con incertidumbre. Se discutirán los pasos generales involucrados  
en un estudio de simulación y la importancia de la aleatoriedad. Se introduce la Simulación  
de Monte Carlo como una técnica fundamental.",        "theory": "La simulación es una  
técnica para imitar el comportamiento de un sistema real a través de un modelo. La Simulación  
de Monte Carlo utiliza muestreo aleatorio para obtener resultados numéricos. Los pasos de  
la simulación incluyen: (1) Definición del problema, (2) Formulación del modelo, (3)  
Recolección de datos, (4) Implementación del modelo, (5) Verificación y validación,  
(6) Experimentación, (7) Análisis de resultados, (8) Documentación. La aleatoriedad se  
introduce mediante la generación de números aleatorios.",        "challenges": "Problema  
1: Estimar el valor de usando Simulación de Monte Carlo. Solución: Generar puntos  
aleatorios dentro de un cuadrado de lado 2. Contar cuántos puntos caen dentro de un  
círculo inscrito en el cuadrado. La proporción de puntos dentro del círculo es aproximadamente  
igual a /4. Problema 2: Simular el lanzamiento de un dado 1000 veces y calcular la  
frecuencia de cada número. Solución: Usar un generador de números aleatorios para simular  
el lanzamiento del dado y registrar la frecuencia de cada resultado. Preguntas para  
discusión: ¿Cuáles son las ventajas y desventajas de la simulación en comparación con  
otros métodos de análisis? ¿Cómo afecta la calidad de los números aleatorios a la precisión  
de la simulación de Monte Carlo?"      }    },    {      "module_title": "Generación  
de Números Aleatorios y Variables Aleatorias",      "module_num": "2",      "objectives":  
[        "Comprender los métodos para generar números aleatorios.",        "Aplicar  
pruebas de aleatoriedad para verificar la calidad de los generadores.",        "Aplicar  
la técnica de la transformada inversa para generar variables aleatorias.",        "Aplicar  
las técnicas de aceptación y rechazo para generar variables aleatorias"      ],      "content_outline":  
"Generación de números aleatorios. Pruebas de aleatoriedad. Transformada inversa. Aceptación  
y rechazo.",      "class_notes": {        "class_num": "2",        "introduction":  
"Esta clase profundiza en la generación de números aleatorios y variables aleatorias,  
elementos cruciales para la simulación. Se estudian diferentes métodos de generación,  
pruebas de aleatoriedad y técnicas para generar variables aleatorias a partir de distribuciones  
específicas.",        "theory": "Los generadores de números aleatorios (GNA) son algoritmos  
que producen secuencias de números que aparentan ser aleatorios. Un método común es  
el generador congruencial lineal (GCL):  $X_{i+1} = (aX_i + c) \bmod m$ . Las pruebas de  
aleatoriedad evalúan si una secuencia de números generados por un GNA cumple con las  
propiedades estadísticas de la aleatoriedad (uniformidad, independencia). La transformada  
inversa permite generar variables aleatorias a partir de su función de distribución  
acumulada. La técnica de aceptación y rechazo es útil para generar variables aleatorias  
de distribuciones complejas.",        "challenges": "Problema 1: Implementar un GCL y  
realizar una prueba de Chi-cuadrado para evaluar su aleatoriedad. Solución: Implementar  
el algoritmo GCL con parámetros apropiados. Generar una secuencia de números. Dividir el  
rango [0, 1) en intervalos. Contar cuántos números caen en cada intervalo. Comparar las  
frecuencias observadas con las frecuencias esperadas usando la prueba de Chi-cuadrado.  
Problema 2: Generar variables aleatorias exponenciales usando la transformada inversa.  
Solución: La función de distribución acumulada de la exponencial es  $F(x) = 1 - \exp(-x)$ .  
Resolver para x en términos de u (un número aleatorio uniforme entre 0 y 1):  $x = -\ln(1-u)$ .  
Implementar este algoritmo. Preguntas para discusión: ¿Qué propiedades debe tener un buen
```

```

generador de números aleatorios? ¿Cómo se elige un GNA para una aplicación específica?
¿Cuáles son las limitaciones de la técnica de la transformada inversa?"      }      },
{      "module_title": "Modelos de Datos de Entrada",      "module_num": "3",      "objectives":
[      "Proponer modelos estadísticos para representar datos de entrada a partir
de información del mundo real.",      "Ajustar distribuciones a datos de entrada
utilizando R."      ],      "content_outline": "Ajuste de distribuciones a datos de
entrada en R.",      "class_notes": {      "class_num": "3",      "introduction":
"Esta clase se enfoca en la importancia de representar datos de entrada de manera adecuada
para la simulación. Se exploran técnicas para ajustar distribuciones de probabilidad a
datos reales usando el software R.",      "theory": "El modelado de datos de entrada
implica identificar y ajustar una distribución de probabilidad a los datos observados. Se
pueden usar histogramas, diagramas de dispersión y pruebas de bondad de ajuste (Kolmogorov-Smirnov,
Chi-cuadrado) para seleccionar la distribución adecuada. R proporciona funciones y paquetes
para el ajuste de distribuciones (fitdistrplus).",      "challenges": "Problema:
Dados un conjunto de datos de tiempo de servicio en una cola, ajustar una distribución
exponencial y una distribución gamma a los datos usando R. Solución: Importar los datos
a R. Usar la función `fitdist` del paquete `fitdistrplus` para ajustar ambas distribuciones.
Comparar los resultados (parámetros estimados, errores estándar, pruebas de bondad de
ajuste) para determinar la distribución más adecuada. Preguntas para discusión: ¿Cómo
se decide qué tipo de distribución usar para modelar un conjunto de datos? ¿Qué son las
pruebas de bondad de ajuste y cómo se interpretan?"      }      },      {      "module_title":
"Simulación de Eventos Discretos (SED)",      "module_num": "4",      "objectives":
[      "Entender e identificar la lógica, estructura y componentes de la simulación
de eventos discretos.",      "Entender el paso del tiempo en simulación (Calendario
de eventos).",      "Seleccionar y usar métricas de desempeño adecuadas para modelar
un sistema."      ],      "content_outline": "Paso del tiempo en simulación. Calendario
de eventos. Componentes de la simulación de eventos discretos.",      "class_notes": {
"class_num": "4",      "introduction": "Esta clase introduce la Simulación de Eventos
Discretos (SED), un paradigma fundamental para modelar sistemas donde el estado cambia
solo en puntos discretos en el tiempo. Se explora el concepto de calendario de eventos y
los componentes básicos de un modelo SED.",      "theory": "En SED, el tiempo avanza
de evento en evento. El calendario de eventos es una lista ordenada de eventos futuros,
programados para ocurrir en momentos específicos. Los componentes básicos de un modelo
SED incluyen: entidades (objetos que fluyen a través del sistema), recursos (elementos
que las entidades necesitan), colas (lugares donde las entidades esperan) y eventos
(acciones que cambian el estado del sistema). Las métricas de desempeño (tiempo de espera,
utilización de recursos, throughput) se usan para evaluar el rendimiento del sistema.",
"challenges": "Problema: Simular una cola simple (un servidor, una cola) usando SED.
Calcular el tiempo de espera promedio y la utilización del servidor. Solución: Definir
los eventos: llegada, inicio de servicio, fin de servicio. Implementar la lógica de la
cola: cuando llega una entidad, si el servidor está libre, inicia el servicio; si no, se
une a la cola. Mantener estadísticas del tiempo de espera y el tiempo que el servidor
está ocupado. Preguntas para discusión: ¿Cómo se modela la variabilidad en SED? ¿Qué
tipos de sistemas son adecuados para ser modelados usando SED?"      }      },      {
"module_title": "Software de Simulación (Simul8 y Netlogo)",      "module_num": "5",
"objectives": [      "Introducción al software de simulación Simul8 y Netlogo.",
"Formulación, planeación y definición del sistema en Simul8.",      "Verificación y
validación de modelos en Simul8.",      "Componentes básicos en Netlogo.",      "Simulación
Basada en Agentes"      ],      "content_outline": "Introducción software simulación.
Operaciones básicas e intermedias en Simul8. Componentes básicos en Netlogo. Simulación
basada en agentes",      "class_notes": {      "class_num": "5",      "introduction":
"Esta clase introduce herramientas de software para la simulación, incluyendo Simul8
(para SED) y Netlogo (para simulación basada en agentes). Se exploran las operaciones
básicas e intermedias en Simul8 y los componentes básicos en Netlogo.",      "theory":

```

"Simul8 es un software comercial para SED con una interfaz gráfica amigable. Permite la construcción de modelos visuales y la ejecución de simulaciones. La verificación implica asegurar que el modelo se implementa correctamente. La validación implica asegurar que el modelo representa con precisión el sistema real. Netlogo es un entorno de modelado basado en agentes. Los agentes son entidades autónomas que interactúan entre sí y con el entorno. ", "challenges": "Problema: Construir un modelo simple de una cola en Simul8. Simular el modelo y obtener estadísticas de desempeño. Problema: Construir un modelo simple de propagación de una enfermedad en Netlogo. Preguntas para discusión: ¿Cuáles son las ventajas de usar software de simulación en comparación con la implementación manual? ¿Qué tipo de problemas son más adecuados para ser modelados con Simul8 y cuáles con Netlogo?" } }, { "module_title": "Análisis de Resultados de Simulación", "module_num": "6", "objectives": ["Analizar los resultados de simulación y dar recomendaciones para mejorar un sistema."], "content_outline": "Análisis de resultados.", "class_notes": { "class_num": "6", "introduction": "Esta clase se centra en el análisis de los resultados de la simulación para obtener información útil y tomar decisiones informadas sobre el sistema modelado. Se discuten técnicas para la interpretación y el análisis estadístico de los datos de salida.", "theory": "El análisis de los resultados de la simulación implica la recopilación de datos de salida, el análisis estadístico y la interpretación de los resultados para responder a las preguntas de investigación. Se utilizan técnicas estadísticas como intervalos de confianza, pruebas de hipótesis y análisis de varianza para evaluar la significancia de los resultados. Es importante considerar la variabilidad inherente a la simulación y realizar múltiples réplicas para obtener resultados robustos.", "challenges": "Problema: Dada una simulación de una línea de producción, analizar los resultados para identificar cuellos de botella y proponer mejoras. Solución: Recopilar datos de utilización de recursos, tiempos de espera y throughput en diferentes etapas de la línea de producción. Identificar las áreas con mayor utilización y tiempos de espera. Proponer cambios en la configuración de la línea de producción (por ejemplo, aumentar la capacidad del cuello de botella) y simular nuevamente para evaluar el impacto de los cambios. Preguntas para discusión: ¿Cómo se determina la duración de la simulación y el número de réplicas necesarias? ¿Cómo se valida que los resultados de la simulación son representativos del sistema real?" } }] }