

## **TEMARIO: Simulación Multi-Agente en Unity y software de multitudes**

Titular: Dr. José Alberto Hernández Aguilar  
Adjunto: Ing, Peter Savier Oropeza Martínez  
Martes de 9:00 a 11:00 am  
Viernes de 9:30 a 11:30 am

1. Sistemas Multiagentes
  - 1.1 Definición
  - 1.2 Agentes y sus componentes
  - 1.3 Arquitectura multiagente
  - 1.4 Plataformas de Agentes
  - 1.5 Modelos de comportamiento predefinidos y en entornos dinámicos
  - 1.6 Máquinas de estado finito (FSM)
  - 1.7 Herramientas de Desarrollo
  
2. Mundos Virtuales
  - 2.1 Definición de mundo virtual
  - 2.2 Descripción de los mundos virtuales (características, clasificación, ejemplos y aplicaciones, elementos, propiedades)
  - 2.3 Herramientas de diseño para mundos virtuales
  - 2.4 Importación del mundo virtual a la herramienta de desarrollo multiagente
  - 2.5 Interacción entre objetos
  - 2.6 Colisiones entre objetos
  
3. Plataforma Unity 3D
  - 3.1 Teoría del videojuego
  - 3.2 Situación actual del sector
  - 3.3 Historia breve del videojuego
  - 3.4 Teoría y conceptos
  - 3.5 Interfaz
  - 3.6 Terrain y Skyboxes
  - 3.7 Componentes
  - 3.8 Uso de la física
  - 3.9 Controlador de primera y tercera persona
  - 3.10 Scripting
  - 3.11 Funciones reservadas
  - 3.12 Triggers & Colliders
  - 3.13 Comunicación entre scripts
  - 3.14 Inteligencia artificial integrada (IA)
  - 3.15 Animation
  - 3.16 Pathfinding
  - 3.17 Co-routines
  
4. FrameWork Menge
  - 4.1 Definición de Framework

- 4.2 Definición de máquinas de estado finito
  - 4.3 CUDA
  - 4.4 GPU vs CPU
  - 4.5 Entorno Menge
  - 4.6 Entradas con XML
  - 4.7 Definición de agentes en el espacio
  - 4.8 Definición de obstáculos
  - 4.9 Definición de METAS
  - 4.10 NavMesh
  - 4.11 Scripting y alteraciones del código fuente en C++
  - 4.12 Ejemplos de simulación
- 
- 5. FlameGPU
    - 5.1 Definición
    - 5.2 Arquitectura
    - 5.3 Entorno
    - 5.4 Grafo en el espacio
    - 5.5 Definición de un agente X-Machine
    - 5.6 Protocolos de comunicación
    - 5.7 Definición de agentes
    - 5.8 Definición de obstáculos
    - 5.9 Entradas con XML
    - 5.10 Scripting y alteraciones del código fuente en C++
    - 5.11 Ejemplos de simulación

## Referencias

- Russell, S. J., & Norvig, P. (2004). *Inteligencia Artificial: un enfoque moderno*.
- Jones, M. T. (2008). *Artificial intelligence: a systems approach*. Laxmi Publications, Ltd.
- Simonov, A., Lebin, A., Shcherbak, B., Zagarskikh, A., & Karsakov, A. (2018). Multi-agent crowd simulation on large areas with utility-based behavior models: Sochi Olympic Park Station use case. *Procedia Computer Science*, 136, 453-462.
- Pérez-Ramírez, M., Zayas-Pérez, B., Beltran-Labra, R., Márquez-Salazar, D., & Zezzatti, C. A. O. O. (2016). Implementation of a Reactive Model for Responding to a Trembling Earthquake: A Perspective from Virtual Reality and Multiagent Systems. *Research in Computing Science*, 122, 25-35.
- Narang, S., Randhavane, T., Best, A., Shapiro, A., & Manocha, D. (2016). *Fbcrowd: Interactive multi-agent simulation with coupled collision avoidance and human motion synthesis*. Technical report, Department of Computer Science, UNC Chapel Hill.
- Bussmann, S. (1992). Simulation Environment for Multi-Agent Worlds. *Techn. Rep. D-92-01, DFKI*, 10, 11-12.
- De Paiva Oliveira, A., & Richmond, P. (2016, March). Feasibility Study of Multi-Agent Simulation at the Cellular Level with FLAME GPU. In *FLAIRS Conference* (pp. 398-403).

Petreska, I., & Stamatopoulou, I. (2013, September). A comparative study of tools for visualisation of state-based spatial multi-agent models. In *Proceedings of the 6th Balkan Conference in Informatics* (pp. 53-60). ACM.

Richmond, P., & Chimeh, M. K. (2017, July). Flame GPU: Complex system simulation framework. In *High Performance Computing & Simulation (HPCS), 2017 International Conference on* (pp. 11-17). IEEE.

ACTIVIDADES DE ENSEÑANZA-APRENDIZAJE: -

Exposición de temas con sesión de preguntas y respuestas. - Resolución de ejercicios. - Investigación documental (en artículos recientes y/o clásicos); y discusión de resultados. - Implementación computacional del contenido temático del curso. - Realización de un proyecto final.

CRITERIOS DE EVALUACIÓN

- Exámenes 20%
- Tareas 10%
- Participación 10%
- Exposiciones 20%