

Guía docente de la asignatura

Fecha de aprobación por la Comisión
Académica: 19/07/2024**Programación del Cauce Gráfico
en Gpu para Aplicaciones de
Escritorio, Web y Móviles
(M52/56/5/10)****Máster**

Máster Universitario en Desarrollo del Software

MÓDULO

Fundamentos de la Informática Gráfica

RAMA

Ingeniería y Arquitectura

**CENTRO RESPONSABLE
DEL TÍTULO**

Escuela Internacional de Posgrado

Semestre

Segundo

Créditos

3

Tipo

Optativa

**Tipo de
enseñanza**

Presencial

PRERREQUISITOS Y/O RECOMENDACIONES**Prerrequisitos**

- Asignatura del MDS: Estructuras de datos y algoritmos geométricos de sistemas gráficos (EAGSG)

Recomendaciones:

- Conocimientos de programación orientada a objetos.

BREVE DESCRIPCIÓN DE CONTENIDOS (Según memoria de verificación del Máster)

1. Introducción al cauce gráfico en rasterización: etapas, lenguajes y herramientas.
2. Etapas del cauce: procesado de vértices, ensamblaje de primitivas, rasterización y procesado de fragmentos. Flujo de datos y programabilidad de las etapas.
3. Lenguajes y herramientas para programación del cauce gráfico.
4. Programación del cauce en aplicaciones de escritorio.
5. Programación gráfica y del cauce en sistemas Web.
6. Programación gráfica en dispositivos móviles.
7. Programación del cauce en motores gráficos.



1. Introduction the graphics pipeline in rasterization: stages, languages and tools.
2. Pipeline stages: vertex processing, primitive assembly, rasterization and fragment processing. Data flow and stages programability.
3. Languages and tools for graphics pipeline programming.
4. Pipeline programming in desktop applications.
5. Graphics and pipeline programming in Web systems.
6. Graphics programming in mobile devices.
7. Pipeline programming in graphics engines.

COMPETENCIAS

COMPETENCIAS BÁSICAS

- CB6 - Poseer y comprender conocimientos que aporten una base u oportunidad de ser originales en desarrollo y/o aplicación de ideas, a menudo en un contexto de investigación.
- CB7 - Que los estudiantes sepan aplicar los conocimientos adquiridos y su capacidad de resolución de problemas en entornos nuevos o poco conocidos dentro de contextos más amplios (o multidisciplinares) relacionados con su área de estudio.
- CB8 - Que los estudiantes sean capaces de integrar conocimientos y enfrentarse a la complejidad de formular juicios a partir de una información que, siendo incompleta o limitada, incluya reflexiones sobre las responsabilidades sociales y éticas vinculadas a la aplicación de sus conocimientos y juicios.
- CB9 - Que los estudiantes sepan comunicar sus conclusiones y los conocimientos y razones últimas que las sustentan a públicos especializados y no especializados de un modo claro y sin ambigüedades.
- CB10 - Que los estudiantes posean las habilidades de aprendizaje que les permitan continuar estudiando de un modo que habrá de ser en gran medida autodirigido o autónomo.

RESULTADOS DE APRENDIZAJE (Objetivos)

Conocimientos o contenidos

- **CO2** - Identifica y comprende los conceptos clave y las principales características de los sistemas software, en cuanto a sus requerimientos, diseño o implementación, pruebas, despliegue y explotación.
- **CO3** - Comprende y sabe utilizar los fundamentos y métodos matemáticos necesarios para abordar y resolver aplicaciones gráficas, de animación, de visualización, de simulación, de cuadros de mandos, y/o de realidad virtual y aumentada, entre otras.

Competencias

- **COM2** - Valorar las estrategias y modelos gráficos utilizados para el desarrollo de aplicaciones gráficas estableciendo compromisos entre el grado de realismo obtenido y la capacidad de interacción por parte de usuarios.

Habilidades o destrezas



- **HD03** - Aplica los modelos, métodos, técnicas, paradigmas, algoritmos, lenguajes y herramientas más apropiados para la creación, desarrollo o mantenimiento de sistemas software que cumplan con criterios de calidad, usabilidad, robustez, fiabilidad, seguridad, facilidad de implementación y despliegue en las plataformas más actuales.
- **HD04** - Sabe diseñar, implementar y evaluar algoritmos gráficos, tanto 2D como 3D, para la representación, digitalización, modelado, visualización tanto realista como expresiva, animación e interacción de modelos con su entorno y su aplicación en interfaces de usuario gráficas

PROGRAMA DE CONTENIDOS TEÓRICOS Y PRÁCTICOS

TEÓRICO

1. Visualización 2D y 3D.

1. Rasterización versus Ray-tracing. Comparativa
2. Historia de las tecnologías y herramientas para gráficos por ordenador.
3. Aplicaciones gráficas interactivas en escritorio, web y móviles. Estructura.

2. Lenguajes y herramientas para programación del cauce gráfico en:

1. Aplicaciones de escritorio.
2. Aplicaciones Web.
3. Dispositivos móviles.
4. Soporte para programación del cauce en motores gráficos.

3. Introducción al cauce gráfico en rasterización:

1. Codificación de primitivas como secuencias de vértices. Secuencias indexadas y no indexadas.
2. Matrices de transformación 2D y 3D. Grafos de escena.
3. Sistemas de coordenadas. Matrices de vista y proyección 2D y 3D. Cámaras 3D.
4. Visualización asíncrona en modo diferido.
5. Carga y visualización de texturas.
6. Un modelo de iluminación básico. Fuentes de luz y materiales.
7. Etapas del cauce y tipos de shaders: procesado de vértices, post-procesado de vértices (ensamblaje de primitivas), rasterización, procesado de fragmentos, procesado de muestras (test de profundidad, mezclado).

4. Programación del cauce en aplicaciones de escritorio.

1. Aplicaciones interactivas de escritorio. Eventos de entrada. Librería de ventanas.
2. Librerías para vectores y matrices.
3. Tipos de primitivas y representación en memoria.
4. Envío de primitivas en modo diferido en aplicaciones de escritorio.
5. El cauce gráfico para visualización 2D y 3D. Definición de vistas 2D y cámaras 3D.
6. Carga y visualización de texturas.
7. El lenguaje de programación para shaders.
8. Compilación y activación de shaders. Parámetros.
9. Implementación de grafos de escena.
10. Implementación de animaciones sencillas.



5. Programación gráfica y del cauce en sistemas Web.

1. Interfaz de usuario y gestión de eventos en Web. Elementos de una página.
2. Inserción de elementos gráficos en páginas Web.
3. Representación de vectores y matrices.
4. Representación y visualización de secuencias de vértices.
5. Compilación y activación de shaders. Parámetros.
6. Definición de vistas 2D y de cámaras 3D.
7. Carga y visualización de texturas.
8. Implementación de grafos de escena.
9. Implementación de animaciones sencillas.

6. Programación gráfica en dispositivos móviles.

1. Interfaz de usuario y gestión de eventos en dispositivos móviles.
2. Inserción de elementos gráficos en aplicaciones móviles.
3. Representación de vectores y matrices.
4. Representación y visualización de secuencias de vértices.
5. Compilación y activación de shaders. Parámetros.
6. Definición de vistas 2D y de cámaras 3D.
7. Carga y visualización de texturas.
8. Implementación de grafos de escena.
9. Implementación de animaciones sencillas.

7. Programación del cauce en motores gráficos.

1. Introducción a un motor gráfico.
2. Programabilidad del cauce en motores gráficos.
3. Grafos de nodos.
4. Ejemplos de shaders en motores gráficos.

1. Visualization in 2D and 3D.

1. Rasterization versus Ray-tracing: Comparison.
2. History of technologies and tools for computer graphics.
3. Interactive graphic applications on desktop, web, and mobile. Structure.

2. Languages and tools for graphics pipeline programming in:

1. Desktop applications.
2. Web applications.
3. Mobile devices.
4. Support for graphics pipeline programming in graphics engines.

3. Introduction to the graphics pipeline in rasterization:

1. Encoding primitives as sequences of vertices. Indexed and non-indexed sequences.
2. 2D and 3D transformation matrices. Scene graphs.



3. Coordinate systems. 2D and 3D view and projection matrices. 3D cameras.
4. Asynchronous visualization in deferred mode.
5. Loading and displaying textures.
6. A basic lighting model. Light sources and materials.
7. Pipeline stages and shader types: vertex processing, vertex post-processing (primitive assembly), rasterization, fragment processing, sample processing (depth test, blending).

4. Graphics pipeline programming in desktop applications.

1. Interactive desktop applications. Input events. Window libraries.
2. Libraries for vectors and matrices.
3. Types of primitives and representation in memory.
4. Sending primitives in deferred mode in desktop applications.
5. The graphics pipeline for 2D and 3D visualization. Definition of 2D views and 3D cameras.
6. Loading and displaying textures.
7. The programming language for shaders.
8. Compilation and activation of shaders. Parameters.
9. Implementation of scene graphs.
10. Implementation of simple animations.

5. Graphics and pipeline programming for Web systems.

1. User interface and event management on the Web. Elements of a page.
2. Insertion of graphic elements on web pages.
3. Representation of vectors and matrices.
4. Representation and visualization of sequences of vertices.
5. Compilation and activation of shaders. Parameters.
6. Definition of 2D views and 3D cameras.
7. Loading and displaying textures.
8. Implementation of scene graphs.
9. Implementation of simple animations.

6. Graphics programming for mobile devices.

1. User interface and event management on mobile devices.
2. Insertion of graphic elements in mobile applications.
3. Representation of vectors and matrices.
4. Representation and visualization of sequences of vertices.
5. Compilation and activation of shaders. Parameters.
6. Definition of 2D views and 3D cameras.
7. Loading and displaying textures.
8. Implementation of scene graphs.
9. Implementation of simple animations.

7. Pipeline programming for graphics engines.

1. Introduction to a graphics engine.
2. Programmability of the pipeline in graphics engines.
3. Node graphs.
4. Examples of shaders in graphics engines.

PRÁCTICO



- Casos prácticos de programación del cauce en aplicaciones de escritorio.
- Casos prácticos de gráficos y programación del cauce en Web.
- Casos prácticos de programación de gráficos y del cauce en dispositivos móviles.
- Casos prácticos de programación del cauce en un motor gráfico.

- Practical cases of pipeline programming in desktop applications.
- Practical cases of graphics and pipeline programming on the Web.
- Practical cases of graphics and pipeline programming in mobile devices.
- Practical cases of pipeline programming in a graphics engine.

BIBLIOGRAFÍA

BIBLIOGRAFÍA FUNDAMENTAL

1. J.F. Hughes, A. van Dam, M. McGuire, D.F. Sklar, J.D. Foley, S.K. Feiner, K. Akeley. **Computer Graphics: Principles and Practice** (3ed Edition). Ed. Pearson, 2014. ISBN: 978-0-321-39952-6. Web autores: <http://cgpp.net> - Web editorial: <https://www.pearson.com/en-us/subject-catalog/p/computer-graphics-principles-and-practice/P200000000336/9780321399526>
2. Steven J. Gortler. **Foundations of 3D Computer Graphics** (1st edition). Ed. The MIT Press, 2012. ISBN: 9780262017350. Web autores: <http://www.3dgraphicsfoundations.com/> - Web editorial: <https://mitpress.mit.edu/9780262017350/foundations-of-3d-computer-graphics/>
3. Dave Shreiner, Graham Sellers, John Kessenich, Bill Licea-Kane. **OpenGL Programming Guide: The official guide to learning OpenGL** (9th edition, GL 4.5). Ed. Addison Wesley, 2016. ISBN 13: 9780134495538. Web editorial: <https://www.pearson.com/en-us/subject-catalog/p/opengl-programming-guide-the-official-guide-to-learning-opengl-version-45-with-spir-v/P200000000369/9780134495538>
4. Edward Angel, Dave Shreiner. **Interactive Computer Graphics: A Top-Down Approach with WebGL**, 7th edition. Ed. Pearson, 2015. ISBN-13: 9780133574845. Web editorial: <https://www.pearson.com/en-us/subject-catalog/p/interactive-computer-graphics-a-top-down-approach-with-webgl/P2000000003538/9780133574845>

BIBLIOGRAFÍA COMPLEMENTARIA

1. Graham Sellers, Richard S. Wright, Nicholas Haemel. **OpenGL Superbible: Comprehensive Tutorial and Reference** (7th edition). Ed. Addison-Wesley, 2015. ISBN-13: 9780672337475. Web editorial: <https://www.pearson.ch/Informatik/Addison-Wesley/EAN/9780672337475/OpenGL-Superbible-Comprehensive-Tutorial-and-Reference>
2. Eric Lengyel. **Mathematics for 3D Game Programming and Computer Graphics** (3rd edition). Ed. Cengage Learning, 2011. Web autores: <http://mathfor3dgameprogramming.com/>



ENLACES RECOMENDADOS

Como apoyo a la enseñanza y aprendizaje de esta asignatura, se usará la Plataforma de Recursos de Apoyo a la Docencia (PRADO) de la Universidad de Granada: <https://prado.ugr.es>.

Aprende OpenGL:

- <https://learnopengl.com/>

Páginas del manual de OpenGL y GLSL, todas las versiones:

- <https://docs.gl/>

Esquema del cauce gráfico (consorcio Khronos)

- https://www.khronos.org/opengl/wiki/Rendering_Pipeline_Overview

Guía y tutorial de WebGL, en Mozilla Developer Network

- https://developer.mozilla.org/en-US/docs/Web/API/WebGL_API

Librería GLFW (documentación, código fuente, binarios).

- Sitio web: <https://www.glfw.org>
- Documentación: <https://www.glfw.org/documentation.htm>

Páginas del manual de referencia de OpenGL (consorcio Khronos)

- OpenGL, ver.2.1: <https://registry.khronos.org/OpenGL-Refpages/gl2.1/>
- OpenGL+GLSL, ver.4.5: <https://registry.khronos.org/OpenGL-Refpages/gl4/>

Especificación oficiales de OpenGL (consorcio Khronos):

- Actuales (ver 4.6): https://registry.khronos.org/OpenGL/index_gl.php#apispecs
- Versiones anteriores: https://registry.khronos.org/OpenGL/index_gl.php#oldspecs

Documentación y referencia de WebGL (consorcio Khronos)

- Sitio Web: <https://www.khronos.org/webgl/>
- Especificación (WebGL 2.0): <https://registry.khronos.org/webgl/specs/latest/2.0/>

Documentación y páginas de referencia de OpenGL ES (consorcio Khronos)

- Sitio Web: <https://www.khronos.org/opengles/>
- Páginas del manual de referencia (OpenGL ES 2.0): <https://registry.khronos.org/OpenGL-Refpages/es2.0/>

Documentación del soporte para OpenGL ES en Android

- <https://developer.android.com/develop/ui/views/graphics/opengl/about-opengl>

Manual de Unity: conceptos esenciales de shaders (Unity 3D):

- <https://docs.unity3d.com/Manual/ShaderOverview.html>



EVALUACIÓN (instrumentos de evaluación, criterios de evaluación y porcentaje sobre la calificación final)**EVALUACIÓN ORDINARIA**

El artículo 18 de la Normativa de Evaluación y Calificación de los Estudiantes de la Universidad de Granada establece que la convocatoria ordinaria estará basada preferentemente en la evaluación continua del estudiante, excepto para quienes se les haya reconocido el derecho de evaluación única final.

Se realizará una evaluación continua del trabajo del estudiante, valorando tanto los conocimientos adquiridos como las competencias alcanzadas.

Modalidad presencial:

Para la evaluación en modalidad presencial se tendrán en cuenta los siguientes sistemas de evaluación, indicándose entre paréntesis el rango del porcentaje con respecto a la calificación final del estudiante.

- **SE1** - Actividades realizadas durante el desarrollo del curso mediante la entrega de ejercicios, trabajos, informes, a través de la plataforma docente (50%)
- **SE2** - Actividades realizadas después de finalizar el curso mediante la entrega de ejercicios, trabajos, informes, a través de la plataforma docente (30%)
- **SE5** - Asistencia y participación activa (20%)

Modalidad virtual:

Para la evaluación en modalidad virtual se tendrán en cuenta los siguientes sistemas de evaluación, indicándose entre paréntesis el rango del porcentaje con respecto a la calificación final del estudiante.

- **SE1** - Actividades realizadas durante el desarrollo del curso mediante la entrega de ejercicios, trabajos, informes, a través de la plataforma docente (30%)
- **SE2** - Actividades realizadas después de finalizar el curso mediante la entrega de ejercicios, trabajos, informes, a través de la plataforma docente (30%)
- **SE6** - Participación activa en foros de debate o de recogida de información (20%)
- **SE9** - Cuestionarios on-line (20%)

EVALUACIÓN EXTRAORDINARIA

El artículo 19 de la Normativa de Evaluación y Calificación de los Estudiantes de la Universidad de Granada establece que los estudiantes que no hayan superado la asignatura en la convocatoria ordinaria dispondrán de una convocatoria extraordinaria. A ella podrán concurrir todos los estudiantes, con independencia de haber seguido o no un proceso de evaluación continua. De esta forma, el estudiante que no haya realizado la evaluación continua tendrá la posibilidad de obtener el 100% de la calificación final.

Para la evaluación en este caso se tendrán en cuenta los mismos sistemas de evaluación y criterios que para la evaluación ordinaria.

EVALUACIÓN ÚNICA FINAL

El artículo 8 de la Normativa de Evaluación y Calificación de los Estudiantes de la Universidad de Granada establece que podrá acogerse a la evaluación única final cualquier estudiante que no pueda cumplir con el método de evaluación continua por causas justificadas.

Para acogerse a la evaluación única final, el estudiante, en las dos primeras semanas de impartición de la asignatura o en las dos semanas siguientes a su matriculación, si ésta se ha producido con posterioridad al inicio de las clases o por causas sobrevenidas, lo solicitará, a través del procedimiento electrónico, a la Coordinación del Máster, quien dará traslado al profesorado correspondiente, alegando y acreditando las razones que le asisten para no poder seguir el sistema de evaluación continua.

Para la evaluación en este caso se tendrán en cuenta los siguientes sistemas de evaluación:

- **SE1** - Actividades realizadas durante el desarrollo del curso mediante la entrega de ejercicios, trabajos, informes, a través de la plataforma docente (40%)
- **SE2** - Actividades realizadas después de finalizar el curso mediante la entrega de ejercicios, trabajos, informes, a través de la plataforma docente (40%)
- **SE9** - Cuestionarios on-line (20%)

INFORMACIÓN ADICIONAL

Siguiendo las indicaciones recogidas en el artículo 15 de la Normativa de Evaluación y de Calificación de la Universidad de Granada sobre la originalidad de los trabajos presentados por los estudiantes, se informa de lo siguiente:

1. La Universidad de Granada fomentará el respeto a la propiedad intelectual y transmitirá a los estudiantes que el plagio es una práctica contraria a los principios que rigen la formación universitaria. Para ello, procederá a reconocer la autoría de los trabajos y su protección, de acuerdo con la propiedad intelectual, según establezca la legislación vigente.
2. El plagio, entendido como la presentación de un trabajo u obra hecho por otra persona como propio o la copia de textos sin citar su procedencia y dándolos como de elaboración propia, conllevará automáticamente la calificación numérica de cero en la asignatura en la que se hubiera detectado, independientemente del resto de las calificaciones que el estudiante hubiera obtenido. Esta consecuencia debe entenderse sin perjuicio de las responsabilidades disciplinarias en las que pudieran incurrir los estudiantes que plagien.
3. Los trabajos y materiales entregados por parte de los estudiantes tendrán que ir firmados con una declaración explícita en la que se asume la originalidad del trabajo, entendida en el sentido de que no ha utilizado fuentes sin citarlas debidamente.

Información de interés para estudiantado con discapacidad y/o Necesidades Específicas de Apoyo Educativo (NEAE): [Gestión de servicios y apoyos](https://ve.ugr.es/servicios/atencion-social/estudiantes-con-discapacidad) (<https://ve.ugr.es/servicios/atencion-social/estudiantes-con-discapacidad>).

