Course guide
804260 - PGA - Advanced Graphics Programming

Last modified: 20/07/2025

Unit in charge: Image Processing and Multimedia Technology Centre
Teaching unit: 804 - CITM - Image Processing and Multimedia Technology Centre.

Degree: BACHELOR'S DEGREE IN VIDEO GAME DESIGN AND DEVELOPMENT (Syllabus 2014). (Optional subject).

Academic year: 2025 **ECTS Credits:** 6.0 **Languages:** Catalan, English

LECTURER

Coordinating lecturer: Miquel Suau

Others: Miquel Suau
Albert García

TEACHING METHODOLOGY

Lectures are divided in 2h sessions. During lectures, the teacher presents the theoretical concepts and explains them by means of examples that are solved in class. Some time is also dedicated to the resolution of the proposed exercises/projects of the subject with the assistance of the teacher (solve the doubts that may appear).

LEARNING OBJECTIVES OF THE SUBJECT

- To describe the advanced theoretical concepts of computer graphics.

- To apply advanced theoretical concepts of computer graphics to video game development projects.

- To obtain the maximum performance of GPUs for the development of complex graphics algorithms in real time.

STUDY LOAD

Type	Hours	Percentage
Hours large group	18,0	12.00
Self study	90,0	60.00
Hours medium group	30,0	20.00
Guided activities	12,0	8.00

Total learning time: 150 h

CONTENTS

Shader programming with GLSL

Description:
1. Introduction to GLSL
2. Basic GLSL tools
3. Mathematical foundations for raycasting techniques

Related activities:
Delivery 1: Implementation of a procedural raycasting shader

Full-or-part-time: 50h
Theory classes: 10h
Practical classes: 10h
Self study : 30h

GPU programming with OpenGL

Description:
1. Global view of the render pipeline
2. Normalized device coordinates (NDC)
3. Rendering of triangle meshes
4. Passing "global" variables to shaders
5. State handling (visibility tests, blending, etc)
6. Debug and optimization tools
7. Geometric transformations
8. Render to texture

Full-or-part-time: 50h
Practical classes: 10h
Guided activities: 10h
Self study : 30h

Advanced computer graphics techniques

Description:
1. Bump, normal, and relief mapping
2. Full-screen effects
3. Screen Space Ambient Occlusion (SSAO)
4. Environment mapping
5. Physically based rendering
6. Water
7. Bloom

Related activities:
Final project: Implementation of a graphics engine

Full-or-part-time: 50h
Theory classes: 10h
Practical classes: 10h
Self study : 30h

GRADING SYSTEM

Two practical exercises (PE) related to specific contents of the subject will be developed and evaluated during the course. As well, a final project (FP), where the student has to implement a graphics engine with some of the techniques presented at class, will be evaluated. Participation and learning attitude (PART) is also considered. This part will be evaluated according to the participation of the student at class (solving exercises, proposing solutions/alternatives), the interest shown in learning the different techniques. As a project subject, there is no reevaluation exam.

Final mark = 0.30 * PE1 + 0.30 * PE2 + 0.30 * FP + 0.1 * PART

Irregular actions that may lead to a significant variation of the grade of one or more students constitute a fraudulent performance of an evaluation act. This action entails the descriptive grade of failure and a numerical grade of 0 for the ordinary global evaluation of the course, without the right to re-evaluation.

If the lecturers have indications of the use of AI tools not allowed in the evaluation tests, they may summon the students concerned to an oral test or a meeting to verify the authorship.

EXAMINATION RULES.

Part of the activities will be developed at class with the assistance of the lecturer. Students should also work autonomously to finish the activities and projects proposed during the course.

Projects can be done in groups of 2 people and will be submitted via the Campus Virtual following the guidelines provided by the instructions document of each one (name of the files, etc.). Projects submitted after midnight of the specified date will be considered as NP. Any issues that do not allow the student to submitt a project in time should have a reasonable cause and must be communicated with enough anticipation to the lecturer. The evaluation of the projects does not consist just on submitting the code, but also on oral presentations when required.

Projects have to be executed at CITM, so be sure that you work with the same version of the software as the one provided at the center and that your projects can be executed there without errors.

BIBLIOGRAPHY

Basic:
- Kessenich, J.M.; Sellers, G.; Shreiner, D. OpenGL programming guide: the official guide to learning OpenGL, version 4.5 with SPIR-V. 9th ed. Upper Saddle River, NJ: Addison-Wesley, cop. 2017. ISBN 9780134495491.
- Rost, Randi J; Licea-Kane, Bill; Ginsburg, Dan. OpenGL shading language. 3rd ed. Upper Saddle River, NJ: Addison Wesley, 2010. ISBN 9780321637635.
- Möller, Tomas; Haines, Eric; Hoffman, Naty. Real-time rendering [on line]. 4th ed. Boca Ratón: Chapman and Hall/CRC, 2018 [Consultation: 14/07/2025]. Available on: https://doi-org.recursos.biblioteca.upc.edu/10.1201/b22086. ISBN 9781351816144.

Complementary:
- Nguyen, Hubert. GPU gems 3. Upper Saddle River, NJ: Addison-Wesley, 2007. ISBN 9780321515261.

RESOURCES

Other resources:
ShaderToy: https://www.shadertoy.com />GLSL Sandbox: http://glslsandbox.com />OpenGL: https://www.opengl.org />OpenGL loader: https://glad.dav1d.de />GLM library: https://github.com/g-truc/glm />GLFW library: https://www.glfw.org />ImGui library: https://github.com/ocornut/imgui />STB library: https://github.com/nothings/stb />Assimp library: https://www.assimp.org