COURSE DESCRIPTION CARD - SYLLABUS

Course name
Game programming

Course

Field of study
computing science
Area of study (specialization)
Computer Games and Internet Technologies
Level of study
Second-cycle studies
Form of study
full-time

Year/Semester
1/2
Profile of study
general academic
Course offered in
polish
Requirements
compulsory

Number of hours

Lecture
15
Laboratory classes
45
Other (e.g. online)

Tutorials
Projects/seminars

Number of credit points

5

Lecturers

Responsible for the course/lecturer:
dr inż. Paweł Wojciechowski
email: Pawel.Wojciechowski@cs.put.poznan.pl
telephone: +(48 61) 6653031
Institute of Computing Science
ul Piotrowo 2, 60-965 Poznań

Responsible for the course/lecturer:

Prerequisites

The student starting this course should have basic knowledge of programming, three-dimensional computer graphics, computer networks, artificial intelligence and mathematics in the field of operations on vectors and matrices. They should also have the ability to solve basic problems in the field of

algorithmics, optimize the program code, work with external libraries for the C / C ++ language and the ability to obtain information from the indicated sources.

Course objective

Provide students with basic knowledge about the problems encountered when programming video games, in terms of the operation of graphics engines and physics, both two and three dimensional, the operation of artificial intelligence scripts and low-level programming of visual effects, animation techniques and three-dimensional modeling. Developing students' skills to prepare a three-dimensional model of a given object along with its animation.

Course-related learning outcomes

Knowledge

1. has an ordered, theoretically founded general knowledge in the field of: functionality of two and three-dimensional graphics engines, the way of describing the world for artificial intelligence scripts in games, three-dimensional graphics processing pipeline.

2. has detailed knowledge of the architecture and operation of 3D graphics engines knows the basic requirements for physics simulation and combining it with visualization.

3. knows the basic techniques of animation of 3D models, knows an example tool used to model three-dimensional objects.

4. has a basic understanding of trends in the evolution of three-dimensional computer graphics and graphics cards

Skills

1. is able to obtain information from literature, documentation and a discussion forum on the UnrealEngine 4 engine (in Polish and English), integrate them, interpret them and use alternative solutions to those given during classes.

2. is able to use solving tasks related to the development of artificial intelligence scripts, experimental and simulation methods offered by the UnrealEngine 4 and the evaLUAtion platform.

3. can solving engineering tasks such as programming a simple game, integrate knowledge from various areas of computer science (e.g. object-oriented programming, artificial intelligence methods, network programming, data processing of three-dimensional objects) and other scientific disciplines such as physics.

4. is able to assess the usefulness and the possibility of using individual versions of UnrealEngine 4 modules.

5. can assess the usefulness of 3D graphics engines supporting the programmer in the process of implementing the game, including the advisability of their use.

Social competences

understands that in the area of computer games, knowledge and skills very quickly become obsolete, hence there is a need for continuous learning

Methods for verifying learning outcomes and assessment criteria

Learning outcomes presented above are verified as follows:

The knowledge acquired during the course is verified by the implementation of laboratory tasks. Additionally, the lecture hosts an artificial intelligence scripts tournament.

The laboratories are divided into 5- 6 stages related to specific issues. Each of them requires an implementation of relatively short task which must be presented to the teacher. These tasks are not assessed. Additionally, each student carries out a final project, which is a computer game implemented in the selected technology.

Programme content

Lecture:

The issues of artificial intelligence in games are presented on the basis of the evaLUAtion project prepared during the classes in previous years. The lecture presents the basic assumptions of the environment, the problem of the script's interaction with the environment and the description of the game environment from the script level.

Introduction to the UnrealEngine. Building resources for two-dimensional games - preparing pictures and animating sprites as well as resource management. Interaction of game objects with the environment. Overview of the types of objects in the game and their properties. Introduction of physical properties of selected objects, including methods of limiting simulation to two dimensions.

Principles of 3D modeling on the example of the Blender3D application.Introduction to the tool, basic model properties, texturing methods, types of lighting, bump mapping. Types of animations in 3D graphics on the example of Blender. Keyframe concepts, types of interpolation, armature animation and Shape Keys deformations. Rules for exporting models and animations to the UnrealEngine engine.

Using the model in the process of building a 3D game and importing a standard character into the engine.

Properties of the 3D game engine, basic transformations of objects, import of models from Blender, animations, terrain creation system. The use of physics. Presentation of problems and possibilities of their solutions.

Overview of the 3D graphics pipeline, the evolution of graphics cards, their programming languages, introducing basic concepts: pixel shader, vertex shader, geometry shader, vertex and pixel shading, bump mapping.

Laboratories:

Laboratory classes are conducted in the form of fifteen 3-hour exercises, preceded by a 2-hour instructional session at the beginning of the semester. The laboratory program covers the use of external libraries for various programming languages used in game programming. Laboratory classes are divided into thematic blocks:

Artificial intelligence, during which students will learn about the evaLUAtion environment for learning how to write combat scripts in LUA. Each student prepares a team of three characters, who then participates in the final tournament. Then, at a later date, students carry out tasks related to the use of artificial intelligence mechanisms in the UE4 engine.

Introduction to the UnrealEngine - the use of basic tools, blueprints - the example of implementing a simple platform game using a model and textures prepared in a Blender3D.

Discussion of issues related to the physics module in the UE4 environment - basic mechanisms of collisions, motion and dependencies between objects. Handling of basic events generated during object collisions. Introduction to the concept of physical materials.

Overview of materials and introduction to the particle system. The use of user interface elements (GUI).

Introduction to the use of C ++ for game implementation.

Teaching methods

1st lecture: multimedia presentation, along with a demonstration of the following software: evaLUAtion, Blender, UnrealEnigne4

2. laboratory exercises: solving tasks involving the use of selected technologies, fighting scripts tournament

Bibliography

Basic

1. Perełki programowania gier t. 1, 2 i 3, DeLoura M., Helion, 2002

2. Fizyka dla programistów gier, Bourg, D.M., O'Reilly; Associates, 2003

3. Mathematics for 3D Game Programming and Computer Graphics, Third Edition, Lengyel, E., Course Technology PTR, 2011

4. Programming Game AI by Example, Buckland, M., Jones Bartlett Publishers, 2004

5. Game Coding Complete 4 edition, McShaffry, M., Graham, D., Course Technology PTR, 2012

6. Język Cg. Programowanie grafiki w czasie rzeczywistym, Fernando, R., Kilgard, M.J., Helion, 2003

7. Blender. Kompendium, Kuklo, K., Kolmaga, J., Helion, 2007

8. Animacja komputerowa Algorytmy i techniki, Parent, R., PWN, 2011

Additional

1. GPU Gems: Programming Techniques, Tips, and Tricks for Real-Time Graphics, Fernando, R. (Series Editor), Addison Wesley Professional, 2004

2. GPU Gems 2: Programming Techniques for High-Performance Graphics and General-Purpose Computation, Pharr, M., Fernando, R. (Series Editor), Addison Wesley Professional, 2005

Breakdown of average student's workload

	Hours	ECTS
Total workload	125	5,0
Classes requiring direct contact with the teacher	62	2,5
Student's own work (literature studies, preparation for laboratory classes, writing artificial intelligence scripts, making a 3D model and its animation, making a final project) [1]	63	2,5

[1] delete or add other activities as appropriate