COURSE DESCRIPTION CARD - SYLLABUS

Course name
AI and games

Course

Field of study
Artificial Intelligence
Area of study (specialization)

Level of study
First-cycle studies
Form of study
full-time

Year/Semester
4/7
Profile of study
general academic
Course offered in
English
Requirements
elective

Number of hours

Lecture
22
Tutorials

Laboratory classes
22
Projects/seminars

Other (e.g. online)

Number of credit points

4

Lecturers

Responsible for the course/lecturer:
Paweł Wojciechowski, Ph.D.

e-mail: pawel.wojciechowski@cs.put.poznan.pl

tel: +48 61 665 3031

Faculty of Computing and Telecommunications
Piotrowo 2, 60-965

Responsible for the course/lecturer:

Prerequisites

Programming skills. Basic mathematical knowledge from secondary school. Knowledge of basic algorithms and mechanisms of artificial intelligence.

Course objective

The aim of the course is to familiarize students with the problem of artificial intelligence in computer games, with particular emphasis on games taking place in real time. Students will learn solutions allowing to simulate intelligent behaviour of actors in games on the basis of professional solutions available e.g. in the Unreal Engine game engine.

Course-related learning outcomes

Knowledge

K1st_W2: has a well-grouded knowledge of the problem of creating artificial intelligence in video games

K1st_W3: has well-structured, detailed theoretical knowledge of the key issues related to the creation of artificial intelligence in games, including the use of modern game engines and the use of components prepared there for the requirements of the created level for the needs of bots, and the very mechanisms of building intelligent behaviour and a way of presenting the game environment and character movement

K1st_W4: knows basic techniques, methods and algorithms and tools for building intelligent behaviour of the character in the Unreal Engine environment

K1st_W7: has basic knowledge about the life cycle and processes occurring in software and hardware information systems with particular emphasis on artificial intelligence in games

Skills

K1st_U3: is able to formulate and solve complex computer science problems with particular emphasis on programming character behaviour in a video game using appropriately selected engine components, scene queries, pathfinding mechanisms

K1st_U7: is able to critically analyse and evaluate the functioning of behaviour of an actor controlled by artificial intelligence scripts in a game

K1st_U8: is able to design - according to given specifications - and implement an algorithm simulating intelligent behaviour of various types of actors in the Unreal Engine

K1st_U9: is able to formulate and implement new algorithms controlling the behaviour of bots in different types of video games

K1st_U10: is able to acquire, analyse and process data of various types, synthesise it into knowledge and conclusions useful for improving the strategy of actors controlled by artificial intelligence scripts

K1st_U11: is able to use and adapt models of intelligent behaviour and is familiar with the components of the Unreal Engine used for this purpose

Social competences

K1st_K1: understands that with regard to the subject matter of artificial intelligence in games,

knowledge and skills quickly become out-of-date, recognizing the need for continuous education and improvement of own competences

K1st_K2: is aware of the significance of knowledge and scientific research connected with computer science and artificial intelligence in solving practical problems of key importance for the functioning of individuals and companies in such application areas as entertainment

K1st_K5: can think and act in an enterprising way

Translated with www.DeepL.com/Translator (free version)

Methods for verifying learning outcomes and assessment criteria

Learning outcomes presented above are verified as follows:
Lectures: Assessment test is conducted at the last lecture.

Labs: Part of the class on a selected topic students need to develop programs that control actors in games. These actors take part in tournaments organised for all students at once. The strategy of each solution should be documented in a report. Winners of the tournaments may be exempted from the lecture credit colloquium.

Programme content

Introduction to the issue of artificial intelligence in computer games. Discussion of the basic types of mechanisms taking into account the game category. Characteristics and requirements for artificial intelligence in games.

Discussion of the evaLUAtion environment - a platform for learning to write scripts for artificial intelligence (bots) in games. Presentation of the board on which the game is played, basic control instructions. Presentation of LUA language characteristics. Presentation of how to implement such a solution.

The problem of actors movement in the given environment and used algorithms of path finding.

Modern game engines - what is a game engine anyway? Introduction to Unreal Engine. Discussing basic engine modules and class types. Programming with blueprints.

Presentation of artificial intelligence components in the engine. Presentation of how to create a character controlled by artificial intelligence. Initialising input data and reacting to changing environments. Discussion of the Blackboard and BehaviourTree components.

Discussing the AI Racer platform - which is an environment based on the Unreal Engine enabling the creation of solutions controlling cars. Presentation of the basic elements of the platform and car control. Building a racetrack in the environment.

Discussion of the Environemt Query System mechanisms offered by the Unreal Engine. Presentation of system components. How to integrate the query system with the behaviour tree.

Tournament of fighting bots evaLUAtion - each student prepares scripts which constitute his team. These scripts then take part in a tournament, competing against the other teams.

AI Racer bot tournament - students prepare their scripts to control a car, which then take part in races on previously unknown race tracks.

Procedural generation of content e.g. maps.

Discussion of problems in creating artificial intelligence in games depending on the type of game.

Teaching methods

Lectures: multimedia presentations on selected topics combined with the presentation of selected modules of the discussed software.

Workshops: solving tasks presenting selected aspects of the discussed topics. Work on computers in dedicated environments.

Bibliography

Basic

Mark DeLoura, Game Programming Gems (Game Programming Series), Charles River Media, 2000
or
Mark DeLoura, tł. Rafał Jońca, Perełki programowania gier : vademecum profesjonalisty, T. 1, Helion, 2002

Mark DeLoura, Game Programming Gems 2 (Game Programming Series), Charles River Media, 2001
or
Mark DeLoura, tł. Rafał Jońca, Perełki programowania gier: vademecum profesjonalisty, T. 2, Helion 2002

Dante Treglia, Game Programming Gems 3 (Game Programming Series), Charles River Media, 2002
or
Dante Treglia, tł. Rafał Jońca, Perełki programowania gier: vademecum profesjonalisty, T. 3, Helion, 2003

Mike Dickheiser, Game Programming Gems 6 (Game Programming Series), Charles River Media, 2006
or
Mike Dickheiser, tł. Mikołaj Szczepaniak, Perełki programowania gier: vademecum profesjonalisty, T. 6, Helion, 2008

Mat Buckland, Programming Game AI by Example, Jones & Bartlett Learning, 2004

Joanna Lee, Unreal Engine: nauka pisania gier dla kreatywnych, Helion 2017

LUA reference manual LUA https://www.lua.org/docs.html

Unreal Engine documentation: https://docs.unrealengine.com/5.0/en-US/

Additional

Andrew Kirmse, Game Programming Gems 4 (Game Programming Series), Charles River Media, 2004

Kim Pallister, Game Programming Gems 5 (Game Programming Series), Charles River Media, 2005

Ian Millington, AI for Games, 3rd Edition, CRC Press, 2020

Breakdown of average student's workload

	Hours	ECTS
Total workload	98	4,0
Classes requiring direct contact with the teacher	45	2,0
Student's own work (literature studies, preparation for laboratory classes/tutorials, preparation for tests/exam, project preparation) [1]	43	2,0

[1] delete or add other activities as appropriate