COURSE DESCRIPTION CARD - SYLLABUS

Course name
AI in Games [S2SI1E>SIG]

Course

Field of study
Artificial Intelligence

Year/Semester
2/3

Area of study (specialization)
–

Profile of study
general academic

Level of study
second-cycle

Course offered in
english

Form of study
full-time

Requirements
elective

Number of hours

Lecture
20

Laboratory classes
20

Other (e.g. online)
0

Tutorials
0

Projects/seminars
0

Number of credit points

2,00

Coordinators

dr inż. Paweł Wojciechowski
pawel.wojciechowski@put.poznan.pl

Lecturers

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:
K2st_W2: has a well-grouded knowledge of the problem of creating artificial intelligence in video games
K2t_W3: has advanced detailed knowledge of the issues related to the creation of artificial intelligence in games, including the use of modern game engines and the use of components prepared therein in terms of the requirements of the level created for the bots, and the mechanisms themselves for building intelligent behaviour and the presentation of the game environment and the movement of the

character
K2st_W5: has advanced and detailed knowledge of the life-cycle processes controlling the artificial intelligence of bots in video games
K2st_W6: has advanced tools such as Unreal Engine 5 used in solving complex engineering tasks for creating intelligent behaviour of agents controlling characters in computer games

Skills:
K2st_U6: is able to assess the relevance and applicability of new developments (methods and tools) and new IT products for the use of artificial intelligence in computer games
K2st_U8: is able to critically analyse and evaluate how the behaviour of an actor controlled by artificial intelligence scripts functions in a game
K2st_U9: is able to assess the usefulness of methods and tools such as BehaviourTree and Environment Query System in Unreal Engine 5 for building intelligent AI-driven character behaviour
K2st_U10: is able to solve a non-routine task concerning the processing of various types of data, their synthesis into knowledge and conclusions useful for improving the strategy of actors controlled by artificial intelligence scripts
K2st_U11: is able to design advanced models of intelligent behaviour and to implement them using the components of the Unreal Engine used for this purpose

Social competences:
K2st_K1: understands that in the filed of IT with particular emphasis on the artificial intelligence in games, knowledge and skills are rapidly becoming obsolete, while recognising the need for continuing education and improvement of own competences
K2st_K2: understands the importance of using the latest knowledge related to computer science and artificial intelligence to solve practical problems critical to the functioning of individuals and companies in application areas such as entertainment
K2st_K4: is aware of the need to develop professional achievements and comply with the rules of professional ethics

Methods for verifying learning outcomes and assessment criteria

Learning outcomes presented above are verified as follows:

Lectures: a written assessment with open questions conducted in the last lecture. Each question will be scored. A score of >50% is required to pass.
Labs: Classes will be grouped into thematic blocks. Passing each block requires the preparation of artificial intelligence scripts and a description of how they work. In addition, two blocks conclude with a tournament in which student-prepared scripts compete against each other. Completion of all assignments is required to pass.

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	50	2,00
Classes requiring direct contact with the teacher	40	1,50
Student's own work (literature studies, preparation for laboratory classes/ tutorials, preparation for tests/exam, project preparation)	10	0,50