COURSE DESCRIPTION CARD - SYLLABUS

Course name
Advanced information technologies [S2Elmob1-SPE>ZTI]

Course

Field of study
Electromobility

Year/Semester
2/3

Area of study (specialization)
Energy Processing Systems

Profile of study
general academic

Level of study
second-cycle

Course offered in
Polish

Form of study
full-time

Requirements
compulsory

Number of hours

Lecture
15

Laboratory classes
30

Other
0

Tutorials
0

Projects/seminars
0

Number of credit points

3,00

Coordinators

dr inż. Konrad Górny
konrad.gorny@put.poznan.pl

dr hab. inż. Wojciech Pietrowski prof. PP
wojciech.pietrowski@put.poznan.pl

Lecturers

Prerequisites

A student starting this course should have basic knowledge of the use of: computer, Windows operating system and the ability to structure and object-oriented programming in Python, C ++, Java or C #. A student beginning this course should have basic knowledge of computer science, operating systems, algorithms and data structures, programming languages. A student starting this course should have basic knowledge of analytic and differential geometry, matrix calculus.

Course objective

Acquainting with the basic issues and concepts related to the Python language in the scope enabling the processing of character sequences and text files. Acquainting with the practical application of scripting languages for signal processing. Learning about the basic information about the platform of the advanced visual development environment .NET. Acquisition of the ability to design and implement window applications in C#. Testing the developed applications. Acquainting with modern methods of creating three-dimensional computer graphics. Getting to know the principle of operation of the discussed algorithms for creating graphics. Getting to know the advanced structures of neural networks, elements of machine learning.

Course-related learning outcomes

Knowledge:
Student ma poszerzoną i pogłębioną wiedzę w zakresie wybranych działów matematyki niezbędną do opisu elementów, układów i systemów stosowanych w elektromobilności.
Student ma poszerzoną wiedzę w zakresie technik programowania oraz stosowania nowoczesnych narzędzi informatycznych do analizy i syntezy układów elektrycznych pojazdów hybrydowych i elektrycznych w tym trakcyjnych.
Student ma podbudowaną teoretycznie wiedzę na temat nowoczesnych metod gromadzenia, przetwarzania i analizy danych, także w zakresie stosowania uczenia maszynowego.

Skills:
The student is able to obtain information (in Polish and English) from various sources, interpret it, critically evaluate it, analyze it and synthesize it, as well as draw conclusions and formulate and justify opinions.
The student is able to use modern information and communication tools, advanced programming techniques and machine learning methods when collecting, processing and analyzing data.

Social competences:
The student is aware of the importance of the latest scientific and technical achievements in solving research and practical problems and, if necessary, supporting themselves with expert opinions.

Methods for verifying learning outcomes and assessment criteria

Learning outcomes presented above are verified as follows:

Lecture: credit on the basis of a test consisting of general and test questions. Rating scale 51-60% points satisfactory, 61-70% points satisfactory+, 71-80% points good, 81-90% points good +, 91-100% points very good.
Laboratory: rewarding practical knowledge gained during previous laboratory exercises, checking practical programming skills in Python (final test), assessment of knowledge and skills related to the implementation of individual and group programming projects.
Obtaining additional points for activity during classes, especially for: the ability to cooperate as part of a team practically carrying out a detailed task in the laboratory, the use of elements and techniques that go beyond the material of the lecture and laboratory exercises, aesthetic diligence of completed projects.

Programme content

Language C++, C#. Object-oriented programming. Introduction to the .NET platform. Presentation of the structure of the .NET platform. Characteristics of the Microsoft Visual Studio package. Basics of programming in C#. Advanced elements of the C# language. Use of advanced programming platforms in issues related to artificial intelligence, machine learning, neural network learning. Use of GPGPU, parallel computing parallel computing and hardware acceleration in neural network learning, machine learning.

Course topics

Lecture:
C++, C# language. Object-oriented programming. Classes, objects, inheritance, data encapsulation, polymorphism and virtual methods, abstract classes, function templates, class templates. Introduction to the .NET platform. Presentation of the structure of the .NET platform. .NET types and types. Outline

of the .NET Framework. Overview of programming languages for the .NET platform. Characteristics of the Microsoft Visual Studio package. Presentation of the programming environment. CLR (Common Language Runtime) environment. Core CLR features and services. Memory and resource management. Basics of programming in C#. Overview of language syntax: statements, variables, operators and data types. File access and operations. Rules for creating classes, methods, constructors and objects of these classes. Using the arrays. Tool overview in the SDK (Software Development Kit). Advanced elements of the C# language.
Procedures for creating three-dimensional computer graphics in a high-level language. Graphical presentation of research results with OpenGL. Analysis of a fragment of the real world in order to build your own computer graphics. Preparation of a script and implementation of computer animation.
Learning about advanced structures of neural networks and elements of machine learning.
Laboratory:
Creating and processing objects, objects as function arguments. Components, forms, properties, events, exception handling. Editing forms. Launching the application. Creating a sample application for Windows. Preprocessor directives. Event handling. Handling errors with exceptions. Using Windows interface components. Windows Forms Library. Namespace usage, initial form and its properties. Add controls, specify their properties, and define event handling functions. Mouse and keyboard support, menu, status bar, toolbar. Basic controls: buttons, text boxes, drop-down lists, etc. Creating charts. Dialog boxes: standard and custom dialogs. Bookmarks. Obtaining the graphics object. Graphic tools: fonts, pens, brushes. Drawing and filling functions. Image processing. Pixel processing, image processing. Drawing objects in three dimensions. Geometric transformations, rotation, translation, scaling. Perspective and orthogonal projection. Coloring and shading. Light and shadows. Texture mapping. Color mixing and transparency. Antialiasing. Parametric curves and surfaces
The use of advanced programming platforms in issues related to artificial intelligence, machine learning, and neural network training. The use of GPGPU, parallel computing and hardware acceleration in training neural networks and machine learning.

Teaching methods

Lecture: presentation of issues with the use of multimedia, examples (e.g. computational) given on the blackboard, discussion on problem issues.
Laboratory: performing laboratory exercises in teams under the supervision of the teacher.

Bibliography

Basic:
Stroustrup Bjarne, Programowanie : teoria i praktyka z wykorzystaniem C++, Helion 2020
Gręboszz Jerzy, Opus Magnum C++11 : programowanie w języku C++, Helion 2020
Zieliński Józef, Podstawy programowania w języku C++, Oficyna Wydawnicza "Impuls", 2013
Graham Sellers, Richard S. Wright Jr., Nicholas Haemel, OpenGL Superbible: Comprehensive Tutorial and Reference (7th Edition), Helion 2016
Von Glitschka, Vector Basic Training: A Systematic Creative Process for Building Precision Vector Artwork (2nd Edition), Helion 2016
Online documentation for the Visual Studio.NET development system
Python 3 : kompletne wprowadzenie do programowania / Mark Summerfield ; [tł. Robert Górczyński]., Gliwice, Helion 2010.

Additional:
Griffiths Ian, Adams Matthew, Liberty Jesse, C#. Programowanie, Helion, 2012
F. P. Preparata, M. I. Samos, Geometria obliczeniowa, Helion 2003
M. Jankowski, Elementy grafiki komputerowej, WNT 2006.
P. Kiciak, Podstawy modelowania krzywych i powierzchni. Zastosowania w grafice komputerowej, WNT 2005.
Graham Sellers, Richard S. Wright Jr., Nicholas Haemel, OpenGL Superbible: Comprehensive Tutorial and Reference (7th Edition), Helion 2016

Breakdown of average student's workload

	Hours	ECTS
Total workload	87	3,00
Classes requiring direct contact with the teacher	47	1,50
Student's own work (literature studies, preparation for laboratory classes/ tutorials, preparation for tests/exam, project preparation)	40	1,50