COURSE DESCRIPTION CARD - SYLLABUS

Course name

Computer Science

Course

Field of study

Safety Engineering

Area of study (specialization)

Level of study

First-cycle studies

Form of study

part-time

Year/Semester

1/2

Profile of study

general academic

Course offered in

Polish

Requirements

compulsory

Number of hours

Lecture	Laboratory classes	Other (e.g. online)
10	12	
Tutorials	Projects/seminars	

Number of credit points

2

Lecturers

Responsible for the course/lecturer:

Ph.D., Eng. Krzysztof Hankiewicz

Mail to: krzysztof.hankiewicz@put.poznan.pl

Faculty of Engineering Management

ul. J. Rychlewskiego 2, 60-965 Poznań

Responsible for the course/lecturer:

Ph.D., Eng. Aleksander Jurga

Mail to: aleksander.jurga@put.poznan.pl

Faculty of Engineering Management

ul. J. Rychlewskiego 2, 60-965 Poznań

Prerequisites

1. Knowledge: Basic knowledge of secondary school.

2. Skills: Basic computer literacy.

3. Social competencies: Able to work in computer laboratory group.

Course objective

TStudents should be made familiar with algorithmic thinking, the ways algorithms are developed and coded in programming languages. They should be able to design and implement simple algorithms in modern development environment. They should be provided with the introduction to computer science disciplines the most relevant to further study of safety engineering.

Course-related learning outcomes

Knowledge

1. Student is able to explain what is an algorithm and how it is converted into a computer program. Knows the evolution of programming languages and its impact on programming efficiency. Understands the issue of computational complexity of exact algorithms and the role of heuristic and simulation methods. Understands the basic terminology of net oriented application programs [P6S_WK_04].

2. He knows contemporary trends and best practices in information and IT techniques [P6S_WK_03].

3. He knows the basic methods, techniques, tools and materials used in preparation for conducting scientific research and solving simple engineering tasks with the use of information technology, information protection and computer support [P6S_WK_04].

Skills

1. Is able to design and analize flowcharts of algorithms and explain how they work [P6S_UW_04].

2. Is able to generate in Visual Basic a graphical user interface for simple application, and to program simple engineering task [P6S_UO_01].

3. Can properly select sources and information derived from them, make an evaluation, critical analysis and synthesis of this information [P6S_UW_01].

4. Can use various techniques in order to communicate in a professional environment and in other environments [P6S_UW_02].

5. Is able to identify changes in requirements, standards, regulations and technical progress and the reality of the labor market, and on their basis define the need for supplementing knowledge [P6S_UU_01].

Social competences

1. Is aware of the recognition of the importance of knowledge in solving problems in the field of safety engineering and continuous improvement [P6S_KK_02].

2. Is aware of the understanding of non-technical aspects and effects of engineering activities, including its impact on the environment and the related responsibility for decisions made [P6S_KK_03].

3. He can initiate activities related to the formulation and transfer of information and cooperation in the society in the field of security engineering [P6S_KO_02].

Methods for verifying learning outcomes and assessment criteria

Learning outcomes presented above are verified as follows:

Formative assessment:

a) In the field of lectures: Scored written tests (closed questions) or on the eKursy platform at the end of individual thematic blocks of lectures. Passing threshold min. 50 points. Each lecture ends with control questions as help to solve tests.

b) In the field of laboratory classes: implementation of exercises, practical test on a komputer. Passing threshold min. 50 points.

Summary:

a) In the field of lectures: assessment based on the sum of test points accumulated.

b) In the field of laboratory classes: assessment based on the sum of accumulated points.

Programme content

Lectures:

General knowledge of the problems of basic IT departments.The concept of the algorithm, methods of representing algorithms in the form of block diagrams and pseudocode.The relationship between the way the algorithm is represented and the capabilities of the target programming language.Stages of development of programming languages, with particular emphasis on structural and object-oriented languages. Structural control instructions. Computer architecture and main trends of its development.Basics of Boolean algebra.

Laboratories:

Graphical user interface objects. Event-driven applications. Introduction to object-oriented programming with the help of tools for rapid application generation (Visual Studio).

Teaching methods

Information lecture: multimedia presentation, illustrated with examples on the board.

Work with a book.

Demonstration method.

Laboratory method: multimedia presentation illustrated with examples given on a blackboard and performance of tasks given by the teacher - practical exercises.

Bibliography

Basic

1. Jurga A., Sławińska M., Wybrane aspekty projektowania systemów informacyjnych wspomagających procesy logistyczne, [w:] Gospodarka Magazynowa i Logistyka, 2011.

2. Stallings W., Organizacja i architektura systemu komputerowego, WNT, Warszawa, 2000.

3. Harel D., Rzecz o istocie informatyki. Algorytmika, WNT, Warszawa, 2000.

4. Reichel W., Visual Basic dla studentów : podstawy programowania w Visual Basic 2010, Witkom (Salma Press), Warszawa 2011.

5. Jan Bielecki J.,.Visual Basic do Windows : programowanie zdarzeniowe, Wyd. PLJ, Warszawa 1991.

Additional

1. Samolej S. i inni, Wprowadzenie do informatyki : skrypt dla studentów kierunków nieinformatycznych na uczelniach technicznych. 1, Architektura komputerów, algorytmika, paradygmaty i języki programowania, Oficyna Wydawnicza Politechniki Rzeszowskiej, Rzeszów, 2014.

2. Avery J., [tł. Garbacz B, Kaczmarek D.], 100 sposobów na Visual Studio, Helion, Gliwice, 2005.

Breakdown of average student's workload

	Hours	ECTS
Total workload	50	2
Classes requiring direct contact with the teacher	22	1
Student's own work (literature studies, preparation for laboratory classes, preparation for tests)	28	1