COURSE DESCRIPTION CARD - SYLLABUS

Course name
Programming in C

Course

Field of study	Year/Semester
Electronics and Telecommunications	1/2
Area of study (specialization)	Profile of study
	general academic
Level of study	Course offered in
First-cycle studies	English
Form of study	Requirements
full-time	compulsory

Number of hours

Lecture	Laboratory classes	Other (e.g. online)
30	30	
Tutorials	Projects/seminars	

Number of credit points

6

Lecturers

Responsible for the course/lecturer:
dr inż. Maciej Sobieraj

maciej.sobieraj@put.poznan.pl

tel. +48 616653909

Responsible for the course/lecturer:
dr hab. inż. Mariusz Głąbowski

mariusz.glabowski@put.poznan.pl

tel. +48 616653904

Prerequisites

Student should have basic knowledge of mathematics. Should be able to retrieve and interpret information from books and Internet. Student understands a necessity to acquire a new knowledge and skills stemming from a chosen field of studies.

Course objective

The aim of the subject is to deliver to a student a basic knowledge of algorithms, data structure, computational complexity, and princicples of C programming.

Course-related learning outcomes

Knowledge

1. Knows the principles of construction of computer programs ; has knowledge from the area of computing science; knows the syntax of C.

2. Has a knowledge of implementation in C algorithms (sorting, greedy algorithms, searching, graph algorithms) and data structures (tables, binary trees, graphs).

Skills

1. Is able to write software for basic computational algorithms, using C programming languages.

2. Uses high level programming languages: C.

3. Is able to write and run programs to solve selected problems in telecommunication.

Social competences

1. Is aware of the limitations of his/her current knowledge and skills; is committed to further self-study.

2. Demonstrates responsibility and professionalism in solving technical problems. Is able to participate in collaborative projects.

Methods for verifying learning outcomes and assessment criteria

Learning outcomes presented above are verified as follows:

Knowledge acquired as part of the lecture is verified by an oral and / or written test.

Test issues, on the basis of which questions are prepared, are sent to students by e-mail using the university e-mail system.

The written and / or oral test consists of from 3 to 5 questions for which a descriptive answer is expected. Each answer to a question (oral test) is rated on a scale of 0 to 5 points. Each question is scored equally. The written test consists of from 10 to 15 multiple choice questions. For the correct answer (written test) student can get 1 point. Passing threshold: 50% of points.

In the case of the oral test, students draw questions from a set of 30 questions. In the case of a written test, questions are selected by the teacher.

Skills acquired as part of the laboratory are verified on an ongoing basis. At the end of each laboratory class, the correctness of program implementation is assessed on a scale of 2 to 5. The final grade is the average of grades obtained from individual laboratory classes.

Programme content

As part of the lecture the following issues will be discussed:

- Basic data structures;

- Structure of C programs;

- Operators and expressions;

- Control statements, recursion vs. Iteration;

- Functions;

- Functions with multiple parameters;

- Function call stack;

- Arrays;

- Sorting and searching algorithms;

- Pointers and dynamic memory allocation;

- C characters and strings;

- C data structures;

The following exercises will be carried out as part of the laboratory classes:

- Implementation of a simple C program;

- Implementation of C program with functions;

- Implementation of C program using arrays;

- Implementation of C program using pointers and dynamic memory allocation;

- Implementation of C program with structures;

Teaching methods

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

Laboratory classes: practical exercises on computer with installed C compiler.

Bibliography

Basic

1. D.E. Knuth, The art of computer programming, Addison-Wesley Publishing Company, Reading, MA, 1968, 1973.

2. Paul Deitel, Harvey Deitel, C How to Program, Prentice Hall; 7th edition (March 4, 2013).

Additional

1. T.H. Cormen, C.E. Leiserson, R.L. Rivest, C. Stein, Introduction to Algrithms, The MIT Press; third edition edition (July 31, 2009).

Breakdown of average student's workload

	Hours	ECTS
Total workload	150	6,0
Classes requiring direct contact with the teacher	75	3,0
Student's own work (literature studies, preparation for laboratory classes/tutorials, preparation for tests/exam, project preparation) [1]	75	3,0

[1] delete or add other activities as appropriate