

FACULTY OF ELECTRICAL
ENGINEERING**SUBJECT CARD**

Name in Polish: **Programowanie w języku C**
 Name in English: **Programming in the C language**
 Main field of study (if applicable): **Electrical Engineering**
 Specialization (if applicable):
 Level and form of studies: **1st level, part-time**
 Kind of subject: **obligatory**
 Subject code: **ELR042565**
 Group of courses: **NO**

	Lecture	Classes	Laboratory	Project	Seminar
Number of hours of organized classes in University (ZZU):	20		20		
Number of hours of total student workload (CNPS):	54		54		
Form of crediting:	crediting with grade		crediting with grade		
For group of courses mark (X) final course:					
Number of ECTS points:	2		2		
including number of ECTS points for practical (P) classes :			2		
including number of ECTS points for direct teacher-student contact (BK) classes:	1.40		1.40		

PREREQUISITES RELATING TO KNOWLEDGE, SKILLS AND OTHER COMPETENCES

1. Knowledge of basic problems of computer science (Information technology).
2. Abilities of handling computer with the operating system WINDOWS.

SUBJECT OBJECTIVES

- C1. Knowing and acquiring proficiency in use of principles of structural approach to creating algorithms.
 C2. Knowing of principles of programming in the C language.
 C3. Mastering a skill of writing programs in the C language.

SUBJECT EDUCATIONAL EFFECTS*relating to knowledge:*

PEK_W01 The student has knowledge in the scope of structural programming.

PEK_W02 The student knows fundamentals of the C programming language.

relating to skills:

PEK_U01 The student is able to make use of principles of structural programming.

PEK_U02 The student is able to write a simple program in the C programming language.

relating to social competences:

PEK_K01 The student can independently develop algorithms and simple programs in the C programming language.

PROGRAMME CONTENT		
Form of classes - lecture		Number of hours:
Lec 1	An introduction. Algorithm. Flow charts. Idea of the structural programming.	2
Lec 2	A structure of a program in C. Identifier, data types (fundamental data types: integer, real, character, logical), declaration and initialisation of variables, defining constants. Communication through console. Arithmetical, logical, increment, decrement and assignment operators. Calculating expressions.	2
Lec 3	Flow control in a program: bifurcation of control and jumps, loops, nested loops. A single instruction, a block of instructions; the conditional structures, conditional expressions, the repetitive structures.	2
Lec 4	Preprocessor: directives, macrodefinition. Functions: the structure of a function, arguments of a function, data returned by the function, global definitions and declarations, arguments of the main function, recursivity.	2
Lec 5	A summary of utilization of instructions in the C language. Test.	2
Lec 6	Arrays (simple and multidimensional arrays), strings.	2
Lec 7	Pointers. Dynamic memory allocation.	2
Lec 8	Data structures, unions: declaration of structures, declaration of structure variable, table of structures, pointers and data structures.	2
Lec 9	Files: opening and closing, reading and writing data. String operations. Formatting and binary input/output.	2
Lec 10	A summary of utilization of data structures in the C language. Final test.	2
Total hours:		20

Form of classes - laboratory		Number of hours:
Lab 1	Introduction. Flow charts for simple algorithms.	2
Lab 2	Writing, compiling and running simple programs – writing numbers and strings in the screen.	2
Lab 3	Writing programs with use of bifurcation of control, jumps and loops.	2
Lab 4	Writing more complex programs with use of bifurcation of control, jumps and loops.	2
Lab 5	Utilization of directives and macrodefinitions. Programming with use of functions.	2
Lab 6	Programming table operations.	2
Lab 7	Utilization of strings.	2
Lab 8	Introduction pointers into program.	2
Lab 9	Writing programs with use of data structures and unions.	2
Lab 10	Creating programs which read input data from files and write results into files.	2
Total hours:		20

TEACHING TOOLS USED
N1. Multimedia presentation. N2. Information lecture. N3. Preparation in the form of reports. N4. The C programming environment.

EVALUATION OF SUBJECT EDUCATIONAL EFFECTS ACHIEVEMENT		
Evaluation <i>F - forming (during semester)</i> <i>P - concluding (at semester end)</i>	Educational effect number	Way of evaluating educational effect achievement
F1(W)	PEK_W01 PEK_W02	activity at the classes
F2(W)	PEK_W01 PEK_W02	average of the grades from tests
P(W)	0.1 F1 + 0.9 F2	
F1(L)	PEK_U01 PEK_U02	activity at the classes
F2(L)	PEK_U01 PEK_U02	reports from the classes
P(L)	0.3 F1 + 0.7 F2	

PRIMARY AND SECONDARY LITERATURE

PRIMARY LITERATURE:

- [1] Wirth N., Algorytmy + struktury danych = programy, WNT, Warszawa 2001.
- [2] Kernighan B. W., Ritchie D. M., Język ANSI C, WNT, Warszawa 2003.
- [3] Sexton C., Język C to proste, Wyd. RM, Warszawa 2001.
- [4] Grębosz J., Symfonia C++, Kallimach, Kraków 2000.

SECONDARY LITERATURE:

- [1] Kubiak M. J., Programuję w językach Turbo Pascal i C/C++: programowanie strukturalne z elementami programowania obiektowego, Mikom, Warszawa 2001.
- [2] Stec K., Wybrane elementy języka C, Wyd. Pol. Śląskiej, Gliwice 2001.
- [3] Sexton C., Programowanie w C++ - to proste, RM, Warszawa 2001.

SUBJECT SUPERVISOR

Kazimierz Wilkosz, kazimierz.wilkosz@pwr.edu.pl

MATRIX OF CORRELATION BETWEEN EDUCATIONAL EFFECTS FOR SUBJECT **ELR042565 - Programming in the C language** AND EDUCATIONAL EFFECTS FOR MAIN FIELD OF STUDY **Electrical Engineering**

Subject educational effect	Correlation between subject educational effect and educational effects defined for main field of study and specialization (if applicable)	Subject objectives	Programme content	Teaching tool number
PEK_W01	K1ETK_W15	C.1 C.2	Lec1	N.1 N.2
PEK_W02	K1ETK_W15	C.2	Lec2 Lec3 Lec4 Lec5 Lec6 Lec7 Lec8 Lec9 Lec10	N.1 N.2
PEK_U01	K1ETK_U12	C.4	Lab1	N.3 N.4
PEK_U02	K1ETK_U12	C.3 C.4	Lab2 Lab3 Lab4 Lab5 Lab6 Lab7 Lab8 Lab9 Lab10	N.3 N.4
PEK_K01	K1ETK_K06	C.3 C.4	Lab1 Lab2 Lab3 Lab4 Lab5 Lab6 Lab7 Lab8 Lab9 Lab10	N.3 N.4