



Engineering studies in IT - międzynarodowy program studiów
prowadzonych przez Wydział Matematyki i Informatyki UAM w Poznaniu
Nr projektu POWR.03.03.00-IP.08-00-MPK/16

FOUNDATIONS OF PROGRAMMING

Learning module description

GENERAL INFORMATION

1. Module title: Foundations of programming
2. Module code: DPPR LI0-E
3. Term: winter
4. Duration: 30h lectures + 30h laboratories
5. ECTS: 6
6. Module lecturer: Marcin Witkowski
7. E-mail: mw@amu.edu.pl
8. Language: English

DETAILED INFORMATION

1. Module aim is to demonstrate structural programming on the example of ANSI C and Python. Discussed are the process of creating a program, the concept of algorithm, the formal language, the compiler and the interpreter. The methods of controlling the flow of the program, the data types and their representation in memory are presented. Compound data types, expressions, static variables, pointers, dynamic variables, and dynamic memory management are also discussed.
2. Pre-requisites in terms of knowledge, skills and social competences (where relevant):

SYLLABUS:

- Week 1: Software development. Algorithm, processor, program, programming language. Alphabet, syntax, and semantics. Machine language and higher order languages. Compilation, interpretation and consolidation of the program. Variables, types, first program.
- Week 2: Program flow control. Truth and false, introduction to logical expressions. Compound instructions and control instructions. Name validity ranges, variable life, name overrides. Conditional instruction. Loops. Break and continue loop execution. Jump instruction. Selection Guide.
- Week 3: Program flow control. Truth and false, introduction to logical expressions. Compound instructions and control instructions. Name validity ranges, variable life, name overrides. Conditional instruction. Loops. Break and continue loop execution. Jump instruction. Selection Guide. (cont.)
- Week 4: Basic data types. Representation of numbers and characters. Concept of type. Type role in the programming process, constant and variable. Role of the type in the compilation process. Numeric types. Character types. Defining constants of different types. Type conversions, type projections.
- Week 5: Basic data types. Representation of numbers and characters. Concept of type. Type role in the programming process, constant and variable. Role of the type in the compilation process. Numeric types. Character types. Defining constants of different types. Type conversions, type projections. (cont.)
-



Engineering studies in IT - międzynarodowy program studiów
prowadzonych przez Wydział Matematyki i Informatyki UAM w Poznaniu
Nr projektu POWR.03.03.00-IP.08-00-MPK/16

- Week 6: Variables, operators, and expressions. Defining variables. Operators. Operator Priorities. The order of operations execution.
- Week 7: Complex data types. Arrays, strings, records (structures), and unions. Initiating compound variables. Formatting input / output.
- Week 8: Procedures and functions. Concept of function; Passing parameters and returning the result. Prototype function (function type). Life time and name validity. Hiding information. Library functions.
- Week 9: Procedures and functions. Concept of function; Passing parameters and returning the result. Prototype function (function type). Life time and name validity. Hiding information. Library functions (cont.).
- Week 10: Overloading of operators and functions. Identity of types. Arguments of overloaded functions.
- Week 11: Memory management. Pointers, references and dereferences. Dynamic memory allocation, heap. Pointers and tables. Pointers for passing function parameters. Pointers for functions. Pointers without type.
- Week 12: I / O operations. I / O operations in C (stdio library). I / O operations in C ++. iostream library; stream; Predefined streams. Format control. I / O operations on files. Error handling.
- Week 13: Python. Specific properties of Python language.
- Week 14: Python. Specific properties of Python language. (cont.)
- Week 15: Differences in Python syntax in relation to ANSI C.