Syllabus of the course

"Programming"

Specialty	*121 Software Engineering*
Study Programme	*Software engineering*
Study cycle (Bachelor, Master, PhD)	*the first (Bachelor) level of higher education*
Course status	*mandatory*
Language	*English*
Term	*1 course; 1, 2 semesters*
ECTS credits	*10*
Workload	*Lectures - 48 hours.*
	Laboratory studies - 48 hours.
	Self-study - 204 hours.
Assessment system	*Exam*
Department	*Department of Information Systems, 61166, Kharkiv, Nauky Av., 9a, Simon Kuznets KhNUE, main building, office 413* *phone. +38(057)702-18-31 (add. 4-37)* *website: https://kafis.hneu.net/*
Teaching staff	*Fedorchenko Volodymyr Mykolayovych, PhD (Information technology), associate professor*
Contact	*V.M. Fedorchenko volodymyr.fedorchenko@hneu.net ,*
Course schedule	*According to the current class schedule*
Consultations	*At the department of information systems, face-to-face, according to the consultation schedule, individual, chat in PNS*

Learning objectives and skills:

assimilation of the necessary knowledge about the basic concepts of algorithmization and the technique of applying basic algorithmic structures and data types in programming, studying the main stages of the software design process and determining the principles of procedural programming for the development of programs in the C/C++, Python languages.

Structural and logical scheme of studying an academic discipline

Prerequisites	**Post-requisites**
	Architecture of computers and computer networks
	Algorithms and data structures
	Object-oriented programming
	Databases
	Web programming
	Distributed and parallel computing

Content of the academic discipline

Content module 1 . Programming in C/C++ languages. Lexical foundations of C/C++ languages. Lexical foundations of C/C++ languages

Topic 1. Stages of program development and implementation.

Topic 2. Computer architecture, principles of John von Neumann.

Topic 3. Positional counting systems.

Topic 4. Elements of algorithmic languages C/C++: concept of data types, names, values, pointers, variables, constants, operations, expressions.

Topic 5. Structured programming: sequence, branching and loops.

Topic 6. Preprocessing.

Topic 7. Procedural-oriented programming. Recursion.

Topic 8. Dynamic layout libraries (DLL).

Topic 9. Program development methodologies: bottom-up and top-down design, modular programming.
Content module 2 Programming in C/C++ languages. Basics of programming in C/C++ languages.
Topic 10. Arrays.
Topic 11. Derived data types. Rows in the style of S.
Topic 12. Structures and associations. Dynamic data structures.
Topic 13. Introduction to the C/C++ input-output system. File data structures.
Topic 14. Templates. Standard template library.
Topic 15. Standard string class.
Topic 16. Exception handling. Features of standards C11, C++11, C++14.
Content module 3. Programming in Python. Lexical foundations of the Python language.
Topic 17. Syntax and semantics of the Python language, the concept of data types, names, values, pointers, variables, constants, operations, expressions.
Topic 18. Python language operators.
Topic 19. Functions in the Python language.
Topic 20. Work with modules.
Topic 21. Rules for writing and documenting code in the Python language.
Content module 4. Programming in Python. Basics of Python programming.
Topic 22. Exception handling in the Python language.
Topic 23. Lists, tuples and dictionaries.
Topic 24. Working with files.
Topic 25. Work with lines.
Topic 26. Classes and objects.
Topic 27. Basic built-in modules.

Teaching environment (software)

Material and technical support: multimedia projector, computer classrooms (25 computers), multimedia projector, PNS KhNEU named after S. Kuznetsa, ZOOM
Software: IDE Visual Studio 2019,2022; PyCharm IDE

Assessment system

The university uses a 100-point accumulative system for evaluating the learning outcomes of students of higher education.

Current control is carried out during lectures and laboratory classes and is aimed at checking the level of readiness of a higher education student to perform a specific job and is evaluated by the sum of points scored.

The final control of the first and second semesters includes the semester control, which is conducted in the form of an exam.

The maximum possible number of points for current control during the semester is 60, and the minimum possible number of points is 35.

Current control includes the following control measures: protection of laboratory works, written control works.

More detailed information on the evaluation and accumulation of points for the academic discipline is provided in the work plan (technological map) for the academic discipline.

Course policies

The teaching of the academic discipline is based on the principles of academic integrity. Violations of academic integrity include: academic plagiarism, fabrication, falsification, writing off, deception, bribery, biased assessment. For violation of academic integrity, students of education are subject to the following academic responsibility: repeated assessment of the corresponding type of educational work.

More detailed information about competencies, learning outcomes, teaching methods, assessment forms, self-study is given in the Course program