

Syllabus of the educational discipline «PROGRAMMING»

Cycle of Higher Education	<i>First cycle of higher education (Bachelor's degree)</i>
Field of Study	<i>F Information Technologies</i>
Specialty	<i>F7 Computer engineering</i>
Educational program	<i>Computer systems and networks</i>
Discipline status	<i>Normative</i>
Teaching language	<i>English</i>
Year of studies, semester	<i>1 year (1, 2 semester)</i>
Number of credits ECTS	<i>8 credits</i>
Distribution by types of trainings and hours of study	<i>Lectures, Practical studies, Laboratory studies, Independent training</i>
Form of final assessment	<i>1 semester – Exam, 2 semester – Test</i>
Teacher	<i>Korol Yu. Yu., Associate Professor of the Department of Computer systems and networks, PhD Syniavska O. O., Associate professor of Department of Probability Theory and Mathematical Analysis, PhD</i>
Teacher's contacts	<i>petro.horvat@uzhnu.edu.ua, olga.syniavska@uzhnu.edu.ua</i>
Course Schedule	<i>According to the timetable</i>
<p>The purpose of the discipline " Programming " is to acquaint students with the basic terminology of the field of information technology, mastering the theory of programming for practical use; acquiring the ability to analyze tasks and build algorithms for their solution, and to implement basic algorithms in the programming language.</p> <p>At the end of this course, students should:</p> <p><i>know:</i></p> <ul style="list-style-type: none"> - basic principles of structured, modular and visual programming. <p><i>be able to</i></p> <ul style="list-style-type: none"> - create flowcharts and programs for solving a wide variety of problems in engineering practice; - demonstrate appropriate design, coding, testing, and documenting of computer programs; - apply computer programming concepts to new problems or situations. 	
<p>Prerequisites for learning High School Informatics Course</p>	
<p>Content of the educational discipline Semester 1</p>	
<p>Module 1</p> <p>Topic 1. Fundamentals of programming.</p> <p>Topic 2. C++ programming language.</p> <p>Topic 3. Data Types in C++.</p> <p>Topic 4. Variables and expressions in C++.</p> <p>Topic 5. Operator Expressions in C++.</p> <p>Topic 6. Conditional Statements.</p> <p>Topic 7. Understanding syntax and utility of Iterative Statements .</p> <p>Topic 8. Use of break and continue in Loops.</p> <p>Modular control work</p> <p>Module 2</p> <p>Topic 9. Understanding a Pointer Variable.</p> <p>Topic 10. Arrays.</p> <p>Topic 11. Strings.</p> <p>Topic 12. Enums in C++.</p> <p>Topic 13. Structures.</p> <p>Topic 14. Unions.</p> <p>Modular control work</p>	

Exam			
Semester 2			
Module 3			
Topic 15. Functions in C++.			
Topic 16. Recursion in C++.			
Topic 17. Function Overloading.			
Topic 18. Function Templates.			
Topic 19. Standard Library Function.			
Topic 20. The #include Directive.			
Topic 21. The #define directive.			
Topic 21. Conditional compilation directives			
Modular control work			
Module 4			
Topic 15. Visual programming tools Visual Studio C++			
Topic 16. Visual Studio C++: form, button, label, textbox.			
Topic 17. Components listBox, comboBox and dataGridView.			
Topic 18. Creating Bookmarks and menu.			
Modular control work			
Test			
Course page on the Moodle platform (personal training system)		<i>Syllabus of the educational discipline, hyperlinks to electronic publications of the discipline, recommended literature, students' attendance, lecture materials, presentations, questions for self-control, tests, tasks for checking students' knowledge.</i> https://moodle.uzhnu.edu.ua	
Recommended literature			
1. Schildt H. C++: A Beginner's Guide, Second Edition. McGraw-Hill Osborne Media; 2 edition (December 3, 2003) McGraw-Hill Osborne Media; 2 edition (December 3, 2003). 576 p.			
2. C: How to Program/ Paul J. Deitel, Harvey M. Deitel, Pearson Prentice Hall, 2010. 998p.			
3. Prata S. C++ Primer Plus, Sixth Edition. Addison-Wesley Professional. Pearson Education. 2012. 1438 p.			
4. https://www.programiz.com/c-programming			
5. https://www.tutorialspoint.com/cprogramming/index.htm			
Assessment system of learning outcomes			
<i>Current control carried out the semester and evaluated by the amount of points (max is 100 points). A minimum amount, that allows a student to get credit is 35 (max is 100 points).</i>			
<i>During the first, second and third modules, students complete 4 computer works (maximum number of points for each computer workshop: 5 points). During the fourth module, students complete 1 computer work (maximum number of points 20 points).</i>			
<i>The student's rating (R_s) in the course consists of points that he/she receives for defended computer works (R₁) and the modular control work(R₂):</i>			
$R_s = R_1 + R_2 = 40 + 60 = 100 \text{ points.}$			
<i>Each module is rated at a maximum of 100 points. At the end of the discipline displayed rating score, which is defined as the arithmetic mean of two modules.</i>			
<i>The students whose overall points at the end of the semester are more or equal to 60 points can:</i>			
– get their final grade according to the rating system;			
– pass an exam/resr in order to increase the grade.			
<i>Final (semester) control is carried out in the form of exam/test and evaluated in points (max is 100 points, min is 35 points).</i>			
ECTS and national grading scale			
Mark scale	ECTS	Exam	Test
90 - 100	A	Excellent	Satisfied

82 - 89	B	Good	
74 - 81	C		
64 - 73	D	Satisfactory	
60 - 63	E		
35 - 59	FX	“Unsatisfactory” with possibility to pass the exam again	“Not satisfied” with possibility to pass the exam again
1 - 34	F	“Unsatisfactory” with obligatory repeated study of the discipline	“Not satisfied” with obligatory repeated study of the discipline