

SYLLABUS OF THE ACADEMIC DISCIPLINE «COMPUTING AND PROGRAMMING»

Academic degree	Bachelor
Specialty	141 Electrical energetics, electrical engineering and electromechanics
Academic program	Electrical energetics, electrical engineering and electromechanics
Period of study	1, 2 semesters (1-3 terms)
Total workload	5 credits ECTS (150 hours)
Classroom workload:	
lectures:	2 hours
laboratory works:	2 hours (1, 2 terms), 1 hour (3 term)
practical	2 hours (1, 2 terms)
Language of study	English



Distance learning course: <https://do.nmu.org.ua/course/view.php?id=3446>

Teaching department Information Technologies and Computer Engineering (ITCE)

Information about instructor:



**Kashtan Vita
Yuriivna**

Associate Professor,
Candidate of Technical Sciences,
Associate Professor of the Department
of Information Technologies and
Computer Engineering

Personal page

https://it.nmu.org.ua/ua/HR_staff/prepods/kashtan.php

E-mail:

Kashtan.V.Yu@nmu.one

1. Course abstract

In recent years, powerful tools for verifying software and hardware systems have been developed. Computer Engineering has been contributing the world for the overall development and creating job or employment opportunities in both public and private sectors. This curriculum is designed to foster knowledge and skills to the technician required by the compute engineering and information technology related industries, electrical engineering.

This course covers the fundamentals of computer and mobile device hardware and software, and advanced concepts such as security, networking, programming and

the responsibilities of an IT professional. Students who complete this course will be able to describe the internal components of a computer, assemble a computer system, install operating systems, and troubleshoot them using software tools and diagnostics. Students will also be able to connect to the Internet and share resources in a networked environment. New topics in this version include scripting basics, using remote access technologies, IoT device configuration and communication types, documentation and change management best practices, and also, disaster prevention and recovery methods.

Upon completion of the “Computer engineering and programming” course, students will be able to perform the following tasks:

- choose appropriate computer components for creating, repairing, or upgrading personal computers;
- install and configure components for assembling, repairing, or upgrading personal computers;
- to search for malfunctions on personal computers;
- to explain how to troubleshoot laptops and other mobile devices;
- to install the printer according to the requirements;
- to describe virtualization and cloud computing;
- to install Windows operating systems;
- to manage and maintain Windows operating systems;
- to explain how to configure, protect and troubleshoot mobile and operating systems;
- to development of simple console programs based on acquired knowledge of building algorithms and programming skills in C++, and Visual Basic;
- to development of programs with a graphical user interface in the Visual Basic language.

2. Aim and objectives

The aim of the course – formation of competencies regarding the use of computer hardware and software, operating systems, and programming elements to solve the problems of increasing the efficiency of the use of electric power, electrotechnical, and electromechanical equipment.

Course objectives:

- to acquaint students with higher education with the principles of construction and operation of computer equipment;
- learn the simple components of a personal computer:
- teach students to configure PCs: install components for creating, repairing, or upgrading personal computers; choose appropriate computer components for creating, repairing, or upgrading a personal computer;
- teach how to troubleshoot personal computers;
- to learn to develop algorithms for their solution based on a verbal description of problems;
- to acquaint students with the basics of building system software;
- a study of technologies, means, and methods of programming in C++ and Visual Basic languages;
- to teach students to apply technologies for solving application problems in areas of professional activity.

3. Learning outcomes

Disciplinary learning outcomes:

- to know the basics and principles of computer architecture, history of its development, number systems, units of measurement and presentation of data in computer memory;
- to have hardware and software integration of functioning elements of computing equipment;
- to have the principles of algorithmization and software development;
- to be able to implement calculations when developing console programs and programs with a graphical user interface;
- to have skills in using operators, operands, variables, and data of various types;
- to have skills in using branched operators, loops, and array formation;
- to have skills in working with computer system interfaces, data coding in computers, and modern information technologies;
- to be able to install and configure components to build, repair or upgrade personal computers;
- to be able to classify and use system and application software in practice.

4. Course program

LECTURES

<i>1 semester</i>	
LECTURES	
Lecture 1	1. Introduction to computer engineering and programming
	Information, its types and properties.
	The concept of personal computers, their role in Computing and programming.
	Classification of computer architectures. Von Neumann's architecture
	The main areas of application of computer technology in the field of electrical engineering.
	Significance and main directions of application of computer technology in the field of electrical engineering.
Lecture 2	2. Personal Computer Hardware
	Case and power supplies
	Electricity: Ohm's law. Voltage fluctuations in electrical networks.
	Motherboard Components
	CPUs and Cooling Systems
	Types of Memory
	Adapter Cards and Expansion Slots
	Hard disk drives and SSDs
	Optical Storage Devices
	Personal computer input, output devices.
Characteristics of the main parts of the laptop	

	Electrical Power
	Arithmetic basics of a personal computer
	Logical basics of a personal computer
	Configurations for Specialized Computers
Lecture 3	3. Computer Assembly and Disassembly
	General and Fire Safety
	Install the Motherboard Components
	Install the RAM
	Ports, Connectors, and Cables
	Identify the tools and software used with personal computer components and their purpose.
	Steps of installation work
	Boot the computer after it is assembled
	Computer configuration of the system components
Lecture 4	4. Preventive Maintenance and Troubleshooting
	Personal computer preventive maintenance overview
	Apply Troubleshooting Process to Computer Components and Peripherals
	Setting a computer system in BIOS Setup
	Interaction of the automatic control system with the automatic diagnostic system. POST.
Lecture 5	5. Basic concepts of computer programming
	Features of programming technology
	Object-oriented programming
	Types and composition of programming systems
	A generation of programming languages.
	Programming systems.
	Data Representation in computers. Representing Numbers.
Lecture 6	6. Algorithmization of computational processes
	The concept of algorithm and its main properties.
	Basic concepts of algorithmization of computational processes.
	Variants to set algorithms.
	Structures of algorithms.
	Schemes of algorithms.
	Examples of ways to solve algorithm structures.
Examples of solving problems for compiling algorithms.	
Lecture 7	7. The VBA development environment and its syntax
	The object structure of the VBA language. Main characteristics of components, syntax.
	Basic principles of creating programs in the VBA language.

	Object-oriented programming in VBA.
	Custom classes. Creating object variables.
	Development of the program interface. Components: Form, properties, and methods of the UserForm object; Button, Label, Text Box, and Image.
<i>2 semester</i>	
LECTURES	
Lecture 1	1. Introduction to the C ++ programming language
	General characteristics of language
	Software development technology
	Alphabet and identifiers
	Operations, expressions and operators
	Classification of data types
	Values in C++
	The task of constants
	Existence time and scope of variables
Lecture 2	2. Branch programming
	Development of structured programs
	Conditional instructions: <i>if, else, switch</i>
	Examples of using the if and switch case operators
	Conditional instructions: <i>if then else end if; select case</i>
Lecture 3	3. Loops programming
	The ' <i>while</i> ' loop
	The ' <i>do ... while</i> ' loop
	The <i>for</i> statement
	Operators: <i>for... next; do...loop; while...wend</i>
	Examples of using loop operators.
	Nested loops
	Recommendations for choosing loops
	Control operators in loops
Examples of using loops	
Lecture 4	4. Arrays
	Declaring and initializing arrays
	One-dimensional and two-dimensional arrays
	Examples of using arrays

LABORATORY WORKS

<i>1 semester</i>	
Laboratory work 1	Basic components of a personal computer in HWINFO64.
Laboratory work 2	Components on the motherboard.
Laboratory work 3	Build a Specialized Computer System.
Laboratory work 4	Diagnostic Software.

Laboratory work 5	Post test. BIOS Setup Utility and Common Trouble Shooting.
Laboratory work 6	A block diagram of algorithms in the MS Office 365 environment.
Laboratory work 7	Execute simple data structure programs and their implementation in the form of projects in the VBA environment. Work with data input and output.
Laboratory work 8	Familiarity with components: Form, Button, Label, Text Box, and Image. Compiling programs using these components.
2 semester	
Laboratory work 1	Introduction to Microsoft Visual C++. Types of projects. Creating a project in Microsoft Visual Studio.
Laboratory work 2	Working with Data Types. Arithmetic and logic operations. Development of algorithms and construction of block diagrams.
Laboratory work 3	Linear programming is based on the C++ language.
Laboratory work 4	Branched programming is based on the C++ language.
Laboratory work 5	Cyclic programming in the C++ language.
Laboratory work 6	Development of the user interface in the Visual Basic language based on branched, cyclical processes.

PRACTICAL WORKS

Practical work 1	Electricity and Ohms Laws.
Practical work 2	Complete the Computer Assembly.
Practical work 3	Disassemble a Computer.
Practical work 4	Assembly and disassemble the Laptop
Practical work 5	Numbering systems
Practical work 6	Arithmetic operations in positional number system

5. Technical equipment and/or software

1. A personal computer or laptop with constant access to the Internet.
2. Activated university mail account (student.i.p. @ Nmu.one) at Office365.
3. Active account in the distance education system Moodle.
4. Software:
 - Windows 10;
 - HWINFO64;
 - Internet browser;
 - MS Visual Studio Community 2019;
 - MS Visual Basic;
 - LibreOffice 6.4;
 - MS Office 365;
 - Virtual Desktop;
 - Virtual Laptop;
 - distance learning platform Moodle, MS Teams.

6. Evaluation system and requirements

6.1. The academic achievements of higher education applicants based on the results of the completion of the course will be evaluated on the scale below:

Rating	Institutional
90 – 100	Excellent
74 – 89	Good
60 – 73	Satisfactory
0 – 59	Fail

The general criteria for achieving the learning outcomes correspond to the descriptions of the 6th qualification level of the NQF.

6.2. Higher education applicants can receive a **final grade** in this course based on the formative assessments of knowledge, provided that the number of points from the formative assessments and individual work is at least 60 points.

Maximum rating I-II term:

Theoretical part	Practical part		Laboratorial part		Total
	in time turn-ins	late turn-ins	in time turn-ins	late turn-ins	
20	40	40	40	30	100

Maximum rating I-III term:

Theoretical part	Laboratorial part		Total
	in time turn-ins	late turn-ins	
40	60	50	100

Practical, laboratorial works are accepted according to the control questions for each of the works.

The theoretical part is assessed by the results of a survey of students according to the plan of the theoretical course and independent work.

6.3. Criteria for evaluating the final work. If the higher education applicant has scored less than 60 points and / or seeks to improve the assessment, a final assessment is carried out in a form of a comprehensive control work (CCW), which contains tasks that cover key learning outcomes. The number of specified tasks of the CCW corresponds to the allotted time for assessment. The number of CCW options ensures individualization of the task.

Tickets represent 2 open questions and a task. The correct answer to the open-ended question is estimated at 30 points. Open-ended questions are evaluated by comparison with the standard answers.

Properly solved problem is estimated at 40 points, besides:

- 40 points – compliance with the standard, with units of measurement;
- 25 points – compliance with the standard, without units of measurement or errors in calculations;
- 15 points – minor errors in formulas, without units of measurement;

- 5 points – significant errors in the solution;
- 1 point – the given formulas do not correspond to the standard completely;
- 0 points – the solution is not given.

The maximum number of points for diff. test: 100

The value of the grade for the implementation of the CCW is determined by the average assessment of the components (specified tasks) and is final.

7. Course policy

7.1. Academic Integrity Policy. The academic integrity of higher education applicants is an important condition for mastering the results of training in the discipline and obtaining a satisfactory grade on the formative and summative assessments. Academic integrity is based on condemnation of the practices of copying (writing with external sources other than those allowed for use), plagiarism (reproduction of published texts by other authors without indication of authorship), fabrication (fabrication of data or facts used in the educational process). The policy on academic integrity is regulated by the Regulation "Regulations on the system of prevention and detection of plagiarism at the Dnipro University of Technology https://www.nmu.org.ua/ua/content/activity/us_documents.pdf .

For practical classes, the student should prepare for the relevant topic and be active. Presentations, abstracts, video complexes of exercises should be formatted according to the sample and have correct text links to the literature used.

In case of violation of academic integrity by a student (copying, plagiarism, fabrication), the work is evaluated unsatisfactorily and must be repeated. The teaching professor reserves the right to change the topic of the task.

7.2. Communication policy. Students must have activated university mail.

All written questions to teaching professor regarding the course are to be sent to the university e-mail.

7.3. Test/Exam repeat policy. Works that are submitted in violation of deadlines without good reason are evaluated at a lower grade. Repeat takes place with the permission of the dean's office if there are good reasons (for example, sick leave).

7.4 Evaluation appeal policy. If a student does not agree with the assessment of his knowledge, one may appeal the assessment made by the teaching professor in the prescribed manner.

7.5. Attendance. Full-time students are required to attend classes. Good reasons for not attending classes are illness, participation in university events, business trips, which must be confirmed by documents in case of prolonged (two weeks) absence. The student must inform the teaching professor either in person or through the leader of the learning group about the absence from class and the reasons for absence. If a student is ill, it is highly recommended to stay at home and study utilizing the distance learning platform. Students whose health is unsatisfactory and may affect the health of other students will be encouraged to leave the class (such absence will be considered an absence due to illness).

For objective reasons (for example, international mobility), learning can take place remotely - online, in agreement with the teaching professor.

7.6. Bonuses. Applicants who regularly attended lectures (have no more than two leaves without good reason) and have a written notes of lectures receive an additional 2 points to the results of the assessment to the final grade.

8 Recommended sources of information

1 semester

1. Kashtan V.Yu. Methodological instructions for the implementation of laboratory works in the discipline “Computer Engineering and Programming” for students of specialty 141 “Power engineering, electrical engineering and electromechanics” [Electronic resource], Part1. – 2021. URL: https://it.nmu.org.ua/ua/scientific_method_materials/teaching_materials.php.
2. Каштан В.Ю. Обчислювальна техніка та програмування для студентів спеціальності 141 “ Електроенергетика, електротехніка та електромеханіка”, 2022 Moodle. URL: <https://do.nmu.org.ua/course/view.php?id=5304>
3. Іванов В.Г. Основи інформатики та обчислювальної техніки: підручник / В. Г. Іванов, В. В. Карасюк, М. В. Гвозденко; за заг. ред. В. Г. Іванова. — Х.: Право, 2015. — 312 с.
4. Sarah L. Harris, David Harris. Digital Design and Computer Architecture: ARM Edition 1st Edition. – Morgan Kaufmann. – 2015. – 584p.
5. Sommerville I. Software Engineering, 10th ed. — Addison-Wesley / Pearson Education Limited, 2015. — 816 p.
6. Воробйова О.М. Електроніка та мікросхемотехніка: підручник / О.М. Воробйова, І.П. Панфілов, М.П. Савицька, Ю.В. Флейта. – Одеса: ОНАЗ ім. О.С. Попова, 2015. – 298 с.
7. Albert Paul Malvino. Digital computer electronics. – New Delhi : Tata Mcgraw Hill Education Pvt. Ltd. – 2011. – 522 p.
8. James Lance. The Beginner's Guide to Engineering: Computer Engineering. - CreateSpace Independent Publishing Platform. – 2013. – 158p. ISBN-10 : 1492981540.
9. Cisco Academy IT Essentials Interactive Tutorial: <https://netacad.com>.

2 semester

1. Булгакова О. С. Алгоритмізація і програмування: теорія та практика : навчальний посібник для дистанційного навчання / О. С. Булгакова, В. В. Зосімов, Г. В. Ходякова. – Миколаїв: СПД Румянцева, 2021. – 138 с.
2. Трофименко О.Г. С++. Алгоритмізація та програмування : підручник / О.Г. Трофименко, Ю.В. Прокоп, Н.І. Логінова, О.В. Задерейко. 2-ге вид. перероб. і доповн. Одеса : Фенікс, 2019. 477 с.
3. Roger Mayne. Introduction To Windows And Graphics Programming With Visual C++ (With Companion Media Pack), 2nd Edition. – World Scientific, 2015. – 480 p. ISBN-10: 9814699403, ISBN-13: 978-9814699402.
4. Harvey M. Deitel, Paul J. Deitel. С++ How no Program / 10th Edition. Pearson Education: 2017. 1074 p.
5. Вступ до програмування мовою С++. Організація обчислень: навч. посіб. / Ю. А. Белов, Т. О. Карнаух, Ю. В. Коваль, А. Б. Ставровський. – К.: Видавничополіграфічний центр "Київський університет", 2012. – 175 с.

Information resources:

1. Repository of the Dnipro University of Technology: <http://ir.nmu.org.ua/>