Syllabus of the educational discipline
«PARALLEL AND DISTRIBUTED COMPUTING»

Cycle of Higher Education	*First cycle of higher education (Bachelor's degree)*
Field of Study	*12 Information Technologies*
Specialty	*123 Computer engineering*
Educational program	*Computer systems and networks*
Discipline status	*Normative*
Teaching language	*English*
Year of studies, semester	*4 year (7 semester, 8 semester)*
Number of credits ECTS	*6 credits*
Distribution by types of trainings and hours of study	*Lectures, Laboratory studies, Independent training*
Form of final assessment	*Exam, exam*
Teacher	*Mulesa O.Yu, Doctor of technical sciences, Professor of the department of computer systems and networks*
Teacher's contacts	*oksana.mulesa@uzhnu.edu.ua*
Course Schedule	*According to the timetable*

The purpose of studying discipline "Parallel and Distributed Computing" is to give the notion of methods, languages and programming tools for parallel and distributed computing systems. To study basic approaches to solving problems of mutual exclusion and synchronization of processes with semaphores, mutexes, critical sections, monitors mechanisms and to learn programming tools based on shared variable models and message passing

As a result of studying the discipline the student must:

know:

- the basic concepts and principles of the organization of parallel computing; the main trends in the development of parallel architectures, factors affecting the performance of parallel programs, the criteria for program selection, basic terms and concepts, mathematical apparatus and models of parallel and distributed computing; theoretical foundations of the organization of parallel and distributed computing processes, to build a parallel algorithm based on a sequential algorithm, the problems of the organization of parallel and distributed computing

be able to:

- analyze the parallel capabilities of the input tasks, build a parallel algorithm and develop a program structure for further solution on a parallel computing system (PCS); create a program for PCS using the necessary language of parallel programming; perform optimal placement of program parts on the processor nodes of PCS, taking into account the topological features of the computer system, ensuring minimum time of data exchange in the system; study the effectiveness of the developed program

Prerequisites for learning

Algorithms and Methods of Calculations, Programming, Object-Oriented Programming, Computer Systems, Computer Architecture.

Content of the educational discipline

Topic 1. Structures of the CS based on parallel information processing
Topic 2. Processes. Interaction of processes
Topic 3. Construction and analysis of parallel algorithms
Topic 4. Process Interaction. Critical sections
Topic 5. The mechanism of semaphores. Semaphores in the language of Ada
Topic 6. Semaphores and mutexes in C#
Topic 7. Protected modules in the Ada language
Topic 8. Monitors in Java
Topic 9. Monitors in C#
Topic 10. A general scheme of interaction of processes in systems with local memory.
Topic 11. Ada 95. The Rendezvous Mechanism

Topic 12. MPI library
Topic 13. PVM library
Topic 14. Distributed systems
Topic 15. Hardware and software for distributed systems
Topic 16. Organization of distributed computing
Topic 17. Java. Sockets. Java. RMI
Topic 18. Ada 95. RPC

Course page on the Moodle platform (personal training system)	*Syllabus of the educational discipline, hyperlinks to electronic publications of the discipline, recommended literature, students' attendance, lecture materials, presentations, questions for self-control, methodical materials for laboratory works, tests, tasks for checking students' knowledge. https://moodle.uzhnu.edu.ua*

Recommended literature

1. *Andrew S. Tanenbaum. Distributed Systems: Principles and Paradigms. - CreateSpace Independent Publishing Platform; 2nd edition, 2016. - 702p.*
2. *Gregory Andrews. Foundations of Multithreaded, Parallel, and Distributed Programming. - Pearson; 1st edition, 1999. - 688p.*
3. *Cameron Hughes. Parallel and Distributed Programming Using C++. - Addison-Wesley Professional, 2008. - 691p.*

Assessment system of learning outcomes

The ECTS grade that a student receives after studying a credit module of a discipline is determined according to the student's rating. A student's credit module rating consists of the points the student receives during the semester for the following types of work:
1. Modular control work (MCW) duration of 2 acad. hours each. The maximum number of points for the MCW is 50 points.
2. Performance of laboratory works.
During the 7th semester, students complete 5 laboratory works - (maximum number of points - 40) and 4 works in the 8th semester (maximum number of points - 40).

Scores on individual and independent work of students are awarded for: preparation of essays, modernization of tasks, creative approach to task performance, performance of tasks to improve didactic materials on the discipline: 0-10 points for each module.
Each module is assessed a maximum of 100 points. At the end of the discipline a rating score is derived as the arithmetic average of the points from the two modules.

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