MINISTRY OF EDUCATION AND SCIENCE OF UKRAINE
SIMON KUZNETS KHARKIV NATIONAL UNIVERSITY OF ECONOMICS

"APPROVED"
Head Deputy
(Scientific and Pedagogical work vice-rector)
Mykola ATANASIEV

PARALLEL PROGRAMMING TECHNOLOGIES

Syllabus of the educational discipline

Level of education	***12 Information Technology***
Specialty	***121 Software Engineering***
Educational level	***first (bachelor)***
Educational program	***Software Engineering***

Type of discipline	***Basic***
Language of teaching, studying and assessment	***English***

Head of the Department of
Informatics and Computer Engineering — Sergei UDOVENKO

Kharkiv
2020

МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ
ХАРКІВСЬКИЙ НАЦІОНАЛЬНИЙ ЕКОНОМІЧНИЙ УНІВЕРСИТЕТ
ІМЕНІ СЕМЕНА КУЗНЕЦЯ

"ЗАТВЕРДЖУЮ"
заступник керівника
(проректор з науково-педагогічної роботи)
_____
Микола АФАНАСЬЄВ

ТЕХНОЛОГІЇ ПАРАЛЕЛЬНОГО ПРОГРАМУВАННЯ

робоча програма навчальної дисципліни

Галузь знань	***12 Інформаційні технології***
Спеціальність	***121 Інженерія програмного забезпечення***
Освітній рівень	***перший (бакалаврський)***
Освітня програма	***Інженерія програмного забезпечення***

Статус дисципліни	***базова***
Мова викладання, навчання та оцінювання	***англійська***

Завідувач кафедри
інформатики та компютерної техніки _____ Сергій УДОВЕНКО

Харків
2020

APPROVED
by the Department of Informatics and Computer Engineering meeting
Protocol № 1 dated on August 25, 2020

The developers are:

S.G.Udovenko, Doctor of Technical Sciences, a professor, the head of the Informatics and Computer Engineering Department

O.O.Tiutiunyk, Candidate of Technical Sciences, an associate professor, an associate professor of the Informatics and Computer Engineering Department

I.A.Ushakova, Candidate of Ecomonic Sciences, an associate professor, the head of the Information Systems Department

N. H. Aksak, Doctor of Technical Sciences, an associate professor of the Information Systems Department

Renewal and reapproval list of the academic subject curriculum

Academic year	Date of the Department meeting – the developer of ASWP	Protocol №	Signature of the Department Head

Annotation of an academic subject

Progress in the production of computer technology has far outpaced the development of programming technologies. Parallel architectures have fundamentally improved the performance of programs for solving a lot of issues.

The parallel programming paradigm is connected with the implementation of programs for performing computations on the multiprocessor systems to organize high-performance computing.

Practical tasks of the modern parallel programming usually look like bringing large previously configured programs on the most tenacious programming systems for the C \ C ++ or Fortran languages into a form, which benefits from parallelization using standard tools included in available programming systems.

The academic subject "Parallel Programming Technologies" refers to the group of educational and professional subjects for the preparation of bachelors under the specialty 121 Software Engineering. It is an important part of the computer science cycle. The curriculum of the subject is developed in accordance with the requirements of the field standard of higher education on the basis of the educational and professional training program for the bachelor.

The aim of teaching this academic subject is to form knowledge and skills regarding the basic principles and prospects for the further development of methods for fast processing of huge data through parallel programming on the multiprocessor systems, as well as the acquisition of certain practical experience in this area of activity. At the same time, much attention is paid to the practical work of students on the multicore computers using parallel programming technologies.

The subject "Parallel programming technologies" introduces students to the basic principles of huge data processing and solving complex problems. Much attention is paid to the development of parallel software for high-performance computer systems based on OpenMP technologies and using the parallel programming architecture on the .NET Framework.

During the process of studying, students receive the necessary knowledge during the classroom sessions: lectures and laboratory classes. Also, it is very important for students during the studying process and consolidating of knowledges to perform thei self-sustained work.

Characteristic of an academic subject

Course	3
Semester	5
Number of credits ECTS	5
Final control form	Credit

Structural and logical scheme of an academic subject

Previous disciplines	Following disciplines
Computer architecture	Methods and tools for modeling dynamic systems
Algorithms and methods of calculations	Data mining
Programming	Methods and means of computer calculations
Discrete Math	

Competences and results of studying under this subject

Competences	Results of studying
The ability to master modern technologies of parallel computing, to develop computational	Knowledge of the basic principles of organization and functioning of high-

models and algorithms for parallel problem solving, considering the architectural principles of constructing high-performance computing systems.	performance computing systems. Knowledge of the basic methods and tools for developing parallel programs for shared memory systems.
The ability to develop parallel programs using OpenMP tools. The ability to develop multithreaded programs in Visual Studio 2010 using the .NET Framework	Skills in using modern programming tools. Skills in using libraries for parallel programming.
The ability to develop theoretical models of solvable problems and tasks	Knowledge of decomposition methods by task and data
The ability to organize parallel computing in a local area network	Knowledge of the basic methods and tools for developing parallel programs for distributed memory systems.

Academic subject program

Content module 1. Basics of parallel programming

Topic 1. Introduction. The subject matter of the academic subject, its content and objectives. The purpose of parallel data processing

1.1. Introduction. The subject matter of the academic subject, its content and objectives. The purpose of parallel data processing. Amdahl's Law. Classification of multiprocessor systems.

1.2. OpenMP interface. General information. Basic provisions and basic constructions of OpenMP. The performance of the OpenMP program. Examples of application implementation.

1.3. Parallelizing of the loop. Loop planning and its decomposition.

The condition of the race. Shared and private data management. Loop planning and its decomposition. Effective use of Reduction. Examples of application implementation.

1.4. OpenMP: Performance-oriented programming. Expanding the task queue in OpenMP. Data traffic in parallel sections. Compilation and Debugging. Examples of application implementation.

Topic 2. Algorithmization of parallel computing

2.1. Difficulties in parallel programming. Parallelism. Fundamental concepts of parallel programming. Distributed programming models. Application of parallelism. Some features of parallelization.

2.2. Modeling and analysis of parallel computing. Computational model in the form of a graph “operand operations”. Description of the scheme of parallel performance of the algorithm. Determination of the performance time of the parallel algorithm. Performance indicators of the parallel algorithm. Analysis of the scalability of parallel computations.

2.3. Parallel numerical methods. Matrix multiplication. Multiplication of a matrix by a vector. Achieving the highest possible performance. Using mid-level concurrency. Using a limited set of processors. The choice of a parallel method of calculation. Assessment of performance indicators of the algorithm. The choice of topology of the computing system.

2.4. Parallel numerical methods. A parallel method of multiplying block-represented matrices. Fox's algorithm. Sorting. Bubble sort. Sort Shell. Quick Sort. Graph processing. Finding the minimum spanning tree. Prim's algorithm. Search for shortest ways.

Content module 2. Programming in multiprocessor systems

Topic 3. Parallel programming using the .NET Framework

3.1. Parallel programming in the .NET Framework. Introduction to Parallel Programming .NET Framework 4. PFX Library. PFX library components. Task and Data of the Parallelism Library (TPL).

3.2. Multithreading in the .NET Framework. Definition of a multithreading. The basics of multi-threaded processing. Thread class. Creation of secondary flows. Flow priority assignment. Flow control. CLR thread pool.

3.3. Synchronization of flows. The Lock operator. .NET Framework Synchroniation Classes. Interlocked. Monitor class. Methods of the Monitor class: thread synchronization facts.

3.4. Parallelism of tasks. Creation and launch of the parallel task. Creation of applicated tasks. Waiting for tasks. Handling errors in tasks. Canceling the performance of tasks. Continuation. Continuation and Task <TResult>. Continuation and branch tasks. Continuation of the previous tasks. Task Schedulers and User Interface.

3.5. Data parallelism. The class Parallel. Parallel.Invoke; Parallel.For; Parallel.FoReach.

Topic 4. Distributed computing.

4.1. The MPI standard. Programming for messaging systems. General organization of MPI. The basic functions of mpi. Point-to-point communication operations. Collective operations. Derived data types. Process group management. Communicator management.

4.2. MapReduce programming model. Big data paradigm. Functional programming paradigm. MapReduce capabilities. Map function. Reduce function. Indicators of efficiency of MapReduce programs. Distributed file system. History of Hadoop. Composition Hadoop. The example of the analysis of meteorological data.

4.3. Graphics processor programming. Graphics processor architecture. Graphics processor programming technologies. NVIDIA CUDA Composition. The main limitations of CUDA. Technical details of CUDA. The example of compiling CUDA. Triggered shared memory. CUDA programming model. CUDA memory model. Programming environment. Examples.

The list of practical (seminar)/laboratory classes, as well as questions and tasks for independent work is given in the table "Rating-plan of an academic subject".

Methods of teaching and learning

Methods of teaching - the interaction between teacher and students, during which there is a transfer and acquisition of knowledge, skills and abilities from teacher to student, as well as independent and individual work of the student.

The following methods are used when conducting the types of classes defined by the plan:

1. During the presentation of educational material: - verbal (conversation, explanation, story, instruction); - visual (illustration, demonstration, independent observation); - practical (exercises, practical work, research work).

2. By the organizational nature of training: - methods of organization and implementation of educational and cognitive activities; - methods of stimulation and motivation of educational and cognitive activities; - methods of control and self-control in learning; - binary (combination of theoretical, visual, practical) teaching methods.

3. By the logic of perception and assimilation of educational material: inductive-deductive, reproductive, pragmatic, research, problem, etc.

Procedure for assessing studying results

Khneu named after S. Kuznets uses a cumulative (100-point) evaluation system.

Assessment is carried out by the following types of control:

Current control, which is carried out during the semester during lectures, laboratory classes and is estimated by the amount of points scored (maximum amount - 100 points; the minimum amount that allows a student to get credit - 60 points);

final / semester control, conducted in the form of a semester test, according to the schedule of the educational process.

The test is set as the total amount of points scored on the results of the current control.

Current control includes student assessment during:

Lectures - active work in pairs (0.5 points for each lesson) provided that the student participates in the discussion of the lecture. The total number of points is 7.5.

Laboratory classes - active work in pairs (0.5 points for each class), provided that the student performs tasks on laboratory work. The protection of laboratory work is estimated at 15 points (each laboratory work). The grade for laboratory work is received by the student in the presence of the report on laboratory work, the executed tasks of laboratory work, the expanded answer to questions and performance of control examples. The total number of points is 67.5.

Assessment of student knowledge during lectures and laboratory classes is carried out according to the following criteria: the ability to develop computational models and algorithms for parallel problem solving, taking into account the architectural principles of building high-performance computing systems; ability to develop parallel programs using OpenMP tools and to develop multithreaded programs in Visual Studio 2010 using the .NET Framework. Ability to develop and implement theoretical models of solvable problems and tasks. Ability to implement the organization of parallel computing in a local area network.

The final control of students' knowledge and competencies in the academic subject is performed under the semester pass-fail, the task of which is to verify a student’s understanding of the program material as a whole, the logic and connections between individual sections, the ability to creatively use the accumulated knowledge, the ability to formulate their attitude to a particular problem of the academic subject, etc.

A student shall be considered as certified one if the sum of points received under the results of the final/semester test of progress is equal to or exceeds 60.

The final grade in the academic subject is calculated considering the points obtained during the current control under this system. The total result in points for a semester is: “60 or more points – credited”, “59 or less points – not credited ” and is recorded in the “Progress Report” of the academic subject.

Final control:

The final grade is set according to the scale given in the table "Grade scale: national and ECTS".

Forms of assessment and distribution of points are given in the table "Rating-plan of an academic subject".

Assessment scale: national system and European system of transfer and accumulation of points

<table>
<tr><td rowspan="2">The sum of points under all the types of studying activities</td><td rowspan="2">Score under the European system of transfer and accumulation of points</td><td colspan="2">Score under the national system</td></tr>
<tr><td>for an exam, a term paper, practice for a pass-fail</td><td>For a pass-fail</td></tr>
<tr><td>90 – 100</td><td>A</td><td>Excellent</td><td rowspan="5">Credit</td></tr>
<tr><td>82 – 89</td><td>B</td><td rowspan="2">Good</td></tr>
<tr><td>74 – 81</td><td>C</td></tr>
<tr><td>64 – 73</td><td>D</td><td rowspan="2">Satisfactory</td></tr>
<tr><td>60 – 63</td><td>E</td></tr>
<tr><td>35 – 59</td><td>FX</td><td>Failed</td><td>not credited</td></tr>
</table>

1 – 34	F		

Rating-plan of an academic subject

Topic	Forms and types of education		Forms of evaluation	Max point
Topic 1.	***Classroom work***			
	Lecture 1	Introduction. The subject matter of the academic subject, its content and objectives. The purpose of parallel data processing. Amdahl's Law. Classification of multiprocessor systems.	Active work on pair	0,5
	Laboratory work	Laboratory work 1. OpenMP parallel scope directives. Basic Terms	Active work on pair	0,5
	Individual work			
	Questions and tasks for individual study	Search, selection and review of literary sources on a given topic. Defining the task for the laboratory workshop and acquaintance with the subject area		
	Classroom work			
	Lecture 2	OpenMP interface. General information. Basic provisions and basic constructions of OpenMP. The performance of the OpenMP program.	Active work on pair	0,5
	Laboratory work	Laboratory work 1. OpenMP parallel scope directives. Basic Terms	Active work on pair	0,5
	Individual work			
	Questions and tasks for individual study	Search, selection and review of literary sources on a given topic. Defining the task for the laboratory workshop and acquaintance with the subject area		
	Classroom work			
	Lecture 3	Parallelizing of the loop. Loop planning and its decomposition.	Active work on pair	0,5
	Laboratory work	Laboratory work 1. OpenMP parallel scope directives. Basic Terms	Active work on pair	0,5
	Individual work			

	Questions and tasks for individual study	Search, selection and review of literary sources on a given topic. Preparation for control work	Control work 1	8
	Classroom work			
	Lecture 4	OpenMP: Performance-oriented programming. Expanding the task queue in OpenMP. Data traffic in parallel sections. Compilation and Debugging.	Active work on pair	0,5
	Laboratory work	Laboratory work 1. OpenMP parallel scope directives. Basic Terms	Active work on pair	0,5
			Laboratory work (protection)	15
	Individual work			
	Questions and tasks for individual study	Search, selection and review of literary sources on a given topic. Defining the task for the laboratory workshop and acquaintance with the subject area		
Topic 2.	***Classroom work***			
	Lecture 5	Difficulties in parallel programming. Parallelism. Fundamental concepts of parallel programming.	Active work on pair	0,5
	Laboratory work	Laboratory work 2. Implementation of numerical algorithms using OpenMP technology	Active work on pair	0,5
	Individual work			
	Questions and tasks for individual study	Search, selection and review of literary sources on a given topic. Defining the task for the laboratory workshop and acquaintance with the subject area		
	Classroom work			
	Lecture 6	Modeling and analysis of parallel computing.	Active work on pair	0,5
	Laboratory work	Laboratory work 2. Implementation of numerical algorithms using OpenMP technology	Active work on pair	0,5
	Individual work			
	Questions and tasks for individual study	Search, selection and review of literary sources on a given topic. Defining the task for the laboratory		

		workshop and acquaintance with the subject area		
	Classroom work			
	Lecture 7	Parallel numerical methods. Matrix multiplication.	Active work on pair	0,5
	Laboratory work	Laboratory work 2. Implementation of numerical algorithms using OpenMP technology	Active work on pair	0,5
	Individual work			
	Questions and tasks for individual study	Search, selection and review of literary sources on a given topic. Preparation for control work	Control work 2	9
	Classroom work			
	Lecture 8	Parallel numerical methods. Sorting. Graph processing.	Active work on pair	0,5
	Laboratory work	Laboratory work 2. Implementation of numerical algorithms using OpenMP technology	Active work on pair	0,5
			Laboratory work (protection)	15
	Individual work			
	Questions and tasks for individual study	Search, selection and review of literary sources on a given topic. Defining the task for the laboratory workshop and acquaintance with the subject area		
Topic 3	***Classroom work***			
	Lecture 9	Parallel programming in the .NET Framework	Active work on pair	0,5
	Laboratory work	Laboratory work 3. Development of applications with tasks parallelization in the .NET Framework	Active work on pair	0,5
	Individual work			
	Questions and tasks for individual study	Search, selection and review of literary sources on a given topic. Defining the task for the laboratory workshop and acquaintance with the subject area		
	Classroom work			
	Lecture 10	Multithreading in the .NET Framework	Active work on pair	0,5

	Laboratory work	Laboratory work 3. Development of applications with tasks parallelization in the .NET Framework	Active work on pair	0,5
	Individual work			
	Questions and tasks for individual study	Search, selection and review of literary sources on a given topic. Defining the task for the laboratory workshop and acquaintance with the subject area		
	Classroom work			
	Lecture 11	Synchronization of flows	Active work on pair	0,5
	Laboratory work	Laboratory work 3. Development of applications with tasks parallelization in the .NET Framework	Active work on pair	0,5
	Individual work			
	Questions and tasks for individual study	Search, selection and review of literary sources on a given topic. Preparation for control work	Control work 3	8
	Classroom work			
	Lecture 12	Parallelism of tasks.	Active work on pair	0,5
	Laboratory work	Laboratory work 3. Development of applications with tasks parallelization in the .NET Framework	Active work on pair	0,5
			Laboratory work (protection)	15
	Individual work			
	Questions and tasks for individual study	Search, selection and review of literary sources on a given topic. Defining the task for the laboratory workshop and acquaintance with the subject area		
	Classroom work			
	Lecture 13	Data parallelism	Active work on pair	0,5
	Laboratory work	Laboratory work 4. Development of applications with data parallelization in the .NET Framework	Active work on pair	0,5
	Individual work			

	Questions and tasks for individual study	Search, selection and review of literary sources on a given topic. Defining the task for the laboratory workshop and acquaintance with the subject area		
Topic 4	***Classroom work***			
	Lecture 14	The MPI standard. Programming for messaging systems.	Active work on pair	0,5
	Laboratory work	Laboratory work 4. Development of applications with data parallelization in the .NET Framework	Active work on pair	0,5
	Individual work			
	Questions and tasks for individual study	Search, selection and review of literary sources on a given topic. Defining the task for the laboratory workshop and acquaintance with the subject area		
	Classroom work			
	Lecture 15	MapReduce programming model	Active work on pair	0,5
	Laboratory work	Laboratory work 4. Development of applications with data parallelization in the .NET Framework	Active work on pair	0,5
			Laboratory work (protection)	15
	Individual work			
	Questions and tasks for individual study	Search, selection and review of literary sources on a given topic. Defining the task for the laboratory workshop and acquaintance with the subject area		

List of recommended resources

Basic resources

1. Aksak N. H. Parallel and distributed calculations: textbook. /N. H. Aksak, O. H. Rudenko, A. M Hurzhii.-Kh .: SMIT Company, 2009.- 480 p.

2.Herhel V. P., Stronhin R. H. Fundamentals of parallel computing for multiprocessor computing systems. - N. Novhorod, NNSU, 2001.

3.Bohachev K. Yu. Fundamentals of parallel programming. - M.: BINOM. Knowledge Laboratory, 2003.

4.Voevodin V. V., Voevodin Vl. V. Parallel calculations. – St Petersburg.: BHV-Petersburg, 2002.

5.Nemniuhin S., Stecik O. Parallel programming for multiprocessor computer systems - St. Petersburg: BHV-Petersburg, 2002.

6.Knut D. The skill of computer programming. V. 3. Sorting and searching. – M.: Mir, 1981.

7. Endriu H. R. Fundamentals of multi-threaded, parallel and distributed programming.: Translation from English.- M.: Williams Publishing House, 2003.-512 p .: illustr.- Paral. tit. eng.

8. Shpakovskii H. I., Serikova N. V. Programming for multiprocessor systems in the MPI standard. - Minsk.: Publishing House of BSU, 2002. – 323 p.

Optional resources

1. Bukatov A. A., Datsiuk V. N., Zhehulo A. I. Programming of multiprocessor computer systems. Rostov-na-Donu. TsVVR LCC Publishing House, 2003, 208 p.

2. Antonov A. S. Parallel programming using the MPI technology: textbook. - M.: Publishing House of MSU, 2004. – 71 p.

Information resources

1. Parallel programming technologies / O. O. Tiutiunyk // https://pns.hneu.edu.ua/enrol/index.php?id=7000

2. OpenMP: Specifications www.openmp.org

3. Open MPI: Open Source High Performance Computing www.open-mpi.org/

4. Message Passing Interface (MPI) Forum Home Page www.mpi-forum.org/

5. MapReduce Tutorial https://hadoop.apache.org/docs/r1.2.1/ mapred_tutorial.html