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

APPROVED:
Head of the admissions
committee of Simon Kuznets Kharkiv
National University of Economics

_____ Volodymyr PONOMARENKO
«____» ___04___ 2023 р.

МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ
ХАРКІВСЬКИЙ НАЦІОНАЛЬНИЙ ЕКОНОМІЧНИЙ УНІВЕРСИТЕТ ІМЕНІ СЕМЕНА КУЗНЕЦЯ
№02071211

EXAMINATION PROGRAM

the educational degree of "MASTER"

Speciality 12 " Information Technology "

Speciality 122 "Computer Science".
Educational and professional program – "Computer Science"

Kharkiv, 2023

The professional test is a comprehensive exam that includes practical tasks (three tasks of varying degrees of difficulty) within the frame work of the industry standard of higher education of Ukraine to prepare bachelors in the field of knowledge 12 "Information technologies", specialty 122 "Computer science". The tasks include questions on the following normative educational disciplines: object-oriented programming, organization of data bases and knowledge, parallel computing technologies.

The tasks of the professional entrance exam are designed to identify the knowledge, skills and competencies that bachelor's degree holder in the speciality 12 "Information technologies" possesses (Table 1).

Table 1

The main competencies that a bachelor's degree holder in the speciality 12 "Information technologies" should possess

Professional competences
General competences:
- thorough training in the field of programming, mastery of algorithmic thinking, software engineering methods for software implementation, taking into account requirements for its quality, reliability, production characteristics
- thorough mathematical training, as well as training in the theoretical, methodological and algorithmic foundations of information technologies for the use of mathematical apparatus when solving applied and scientific tasks in the field of information systems and technologies
- the ability to design activities in the professional sphere, the ability to build and use models to describe objects and processes, to carry out their qualitative analysis
- mastery of the software develop men technology in accordance with the requirements and limitation soft he customer.
Special (professional, subject) competences:
- the ability to mathematical and logical thinking, knowledge of basic concepts, ideas and methods of fundamental mathematics and the ability to use them when solving specific tasks
- knowledge of modern methods of construction and analysis of effective algorithms and the ability to implement the specific applications
- knowledge of the principles of structural programming, modern procedurally-oriented languages, basic data structures and the ability to apply them during the software implementation of algorithms for professional tasks
- the ability to object-oriented thinking, knowledge of object-oriented programming languages and the ability to apply an object-oriented approach when designing complex software systems
- knowledge of the principles and rules of formalization of economic situations, the ability to apply mathematical methods of substantiation and management and technical decision-making in various situations

Continuation of Table 1

- knowledge of modern technologies and tools for the development of software systems, the ability to apply the mat all stages of the life cycle
- knowledge of modern theories of data base organization and knowledge, methods and technologies of their development according to the following competencies: 1) concepts and principles of DBK (Databank), DB (Database) and KB (Knowledge base) organization; 2) place of DBK in information systems (IS); 3) levels of data abstraction in Data base design; 4) data base requirements and methods of providing them; 5) general principles of building a data base management system and the structure and principles of its operation; 6) basic models and language tools of DBMS; 7) principles of organization of data storage structures and data access methods; 8) differences between transactional and analytical databases; 9) data base design stages; 10) Methods of developing logical and physical models and physical data schema in the environment of modern CASE tools and DBMS tools; 11) principles of interaction of application programs, which are executed in a high-level language, with relational database management systems; - the ability to design logical and physical database models and queries to them, including the following competencies: 1) formulate requirements for the database and ensure its properties; 2) design a conceptual data model of a specific subject area; 3) choose DBMS in the process of technical design based on the evaluation options of databases, user requirements, analysis of technical, economic, functional, service characteristics of DBMS, using scientific and technical reference information; 4) develop the logical structure of the database in the process of technical design using the method of normalization of relations, using methods of relational algebra, levels of data abstraction, requirements of the selected DBMS; 5) develop database tables and the connection between them in terms of technical design with the help of appropriate technical and software, using table constructors; 6) develop the physical structure of the database in the process of working design using the selected DBMS, using modern technical and software tools of the database developer; 7) export data to the external environment and import data from the external environment into databases in the process of integrating user software into Internet information nodes with the help of modern technical and software, using technologies of distributed applications, WEB technologies etc.; 8) create database tables, triggers, stored procedures, indexes in the conditions

Continuation of Table 1

of data base development using software and technical database design tools, using visual tools of integrated software developer shells; 9) develop input, modification, extraction, display of data in database tables in the process of working design with the help of technical and software tools, using data input and modification forms and tabular modes; 10) develop dataset navigation under the conditions of access to the desired record of the data set using DBMS software, using object navigation methods; 11) develop methods of sorting, filtering, searching data in the process of selecting the necessary data that meet any criteria, using DBMS software, using methods of changing field indexes.
knowledge of modern classifications of parallel computing systems (Flynn's and Jackson's classifications), principles of building architectures of parallel computing systems according to Flynn's classification, which determine work with data flows and commands (tasks)
knowledge of the principles of the model of parallel computing processes, the concept to fun limited parallelism
knowledge of the principles of operation of multiprocessor systems with shared and distributed memory, architectures of systems with shared and distributed memory, vector-conveyor systems, systems with massive parallelism, and computing clusters
knowledge of data parallelism and task parallelism paradigms, parallelization efficiency indicators: acceleration, parallel algorithm efficiency, Amdahl's law
knowledge of the basic model of the algorithm in the form of an "operand - operation" graph, representation of the algorithm in the form of a data flow graph, the schedule of parallel calculations
knowledge of the principles of building parallel methods and algorithms: decomposition of a task into subtasks; separation of information dependencies between half-tasks; scaling of subtasks; distribution of subtasks between system processors
knowledge of the basis of the technology and the general scheme of executing a parallel program when using Open MP technology - the Fork/Join scheme
knowledge of Open MP structure: directives, a library of functions, a set of environment variables
knowledge of scopes of Open MP directives: parallel fragment, parallel region, parallel section; format of Open MP directives
knowledge and ability to use parallelization methods in Open MP: parallelization by cycles and sections; definition of general and local variables; means of simultaneous processing of local variables (reduction operation); synchronization tools, methods of managing the number of streams in Open MP
knowledge and ability to use Open MP library functions to create parallel programs

Ending of Table 1

knowledge and ability to apply the basic principles of MPI (data transfer interface); paired (point-to-point) operations between two processes and collective (collective) communication actions for the simultaneous interaction of several processes; the concept of communicators
knowledge of the main functions of the interaction of parallel processes in MPI (Message Passing Interface)
knowledge of the structure of a parallel program, operations of initialization and completion of an MPI program
ability to use data transfer operations from one process to all program processes; transferring data from all processes to one process (reduction operations; basic types of MPI operations for data reduction functions)
knowledge of blocking point functions of message transmission; functions of simultaneous transmission and reception of messages in MPI
knowledge and ability to use collective data transfer operations in MPI; knowledge of the purpose and composition of global computing operations on distributed data; composition and purpose of global reduction operations in MPI
knowledge and ability to use methods of managing groups of processes and communicators; operations of creating process groups and obtaining information about process groups in MPI; operations to create and manage communicators in MPI
knowledge of the composition and assignment of derived data types in MPI.

CONTENT OF PROFESSIONAL ENTRANCE EXAMS

TOPIC 1. THE MAIN PROVISIONS OF THE OBJECT-ORIENTED APPROACH

1.1. Simple and complex software systems.

Decomposition of software systems. Methods of decomposition. Object-oriented decomposition. The concept of an object. Characteristics of the object. The concept of class. The relationship between a class and its object. Object-oriented analysis and its purpose. The concept to the subject area. The main types of requirements for the software system.

1.2. Object-oriented designing.

Elements of application architecture. Definition of subject area classes. Principles of class design. Object-oriented programming. Principles of object-oriented approach: abstraction, encapsulation, hierarchy, polymorphism.

TOPIC 2. INTRODUCTION TO THE MICROSOFT.NET PLATFORM AND THE C# LANGUAGE

2.1. Microsoft.NET platform.

Microsoft.NET platform architecture, application development tools, program compilation and execution, base class library, typing system.

2.2. General information about the C# language

Features of language use, alphabet, data types, comparison of value types and reference types, built-in value types, built-in reference types, one-dimensional and multi-dimensional arrays, operations, operators, program structure, comments, features of using functions, parameter transfer mechanisms, namespaces, basics of using the.NET base class library

TOPIC 3. IMPLEMENTATION OF THE MAIN CONCEPTS OF OBJECT-ORIENTED PROGRAMMING IN THE C# LANGUAGE

3.1.Abstract data types.

Abstract data type designing. Syntax of structures and classes in the C# language. Class elements. Access to class elements. Link this. Overloading class methods.

3.2. Objects in the program.

Object creation sequence. Constructors. Basic properties of constructors. Freeing memory. Garbage collection system. Static data and methods: purpose, properties, features of use. Aggregation relation. Implementation of aggregation in the C# language. Inheritance relationship. Inheritance syntax in C#. Initialization of the base class object. Options for using inheritance. Overriding methods. Prohibition of inheritance. String representation of the object.

3.3. Implementation of the principle of polymorphism in the C# language.

Early and late bonding. Virtual methods. Abstract classes and methods. Implementation of polymorphic behavior based on an abstract class. Rules of application of abstract classes. Interfaces. Implementation of polymorphic behavior based on the interface. Rules for using interfaces. Principles of operation overloading. Features of using the operator function. Indexers. Properties.

3.4. Handling of exceptional situations

Types of errors in programs. Problems with the traditional approach to error handling. Exception handling mechanism. .NET Standard Library exception classes. Exception handling syntax. Arithmetic overflow check.

TOPIC 4. MICROSOFT .NET FRAMEWORK STANDARD CLASS LIBRARIES

4.1.Sources and consumers of data.

General information about data input-output streams. Algorithms of data input-output flows. Core classes of the .NET standard library to support data I/O.

4.2. Collections.

General information about collections. Core Elements and Data Structures of the .NET Collections Standard Library Typed Collections. Using LINQ and lambda expressions to work with collections.

4.3. String data type.

Features of the implementation of the string data type in the .Net platform. Classes of the .NET standard library for string representation and features of their use. String formatting. Purpose and use of regular expressions. Support for regular expressions in the .NET standard library. Special characters used in regular expressions.

4.4.Attributes.

Introduction to attributes. Program elements to which attributes can be applied. Defined attributes. Using conditional compilation attributes. Compile module level attributes.

4.5. Saving and restoring the state of objects in .NET

Serialization and deserialization. "Graph" of objects during serialization. Creation of classes whose objects can be serialized. Serialization and deserialization processes. Serialization formats. Serialization and deserialization of objects in binary and XML formats.

4.6. Object-relational display of data

General information about object-relational data mapping. Concepts of ORM frameworks for .NET and Java SE platforms. Introduction to models. Model class. Setting up models. Database migrations.

4.7. Using TCP sockets

Distributed software systems. Software clients and servers. General information about TCP sockets. The basics of using TCP sockets on the .NET platform.

TOPIC 5. DEVELOPMENT OF DLL LIBRARIES

5.1. Libraries and their use. Static and dynamic libraries. DLL libraries.

5.2. Development of a DLL library on the Microsoft .NET platform.

TOPIC 6. FUNDAMENTALS OF EVENT-DRIVEN PROGRAMMING

6.1. Delegates and events.

General information about delegates. Declaring and using delegates in C#. Anonymous methods. Group delegates. General information about events. Generating events.

6.2. The basics of using Windows Forms technology.

"Traditional" programming model on the .NET platform. "Windows programming" model on the .NET platform Windows Forms technology. Forms General structure of a graphical user interface application on the .NET platform.

Developing Windows Forms applications using an integrated environment. Form level events.

TOPIC 7. PRINCIPLES AND TEMPLATES OF OBJECT-ORIENTED DESIGNING

7.1. Principles of SOLID object-oriented designing.

A system of SOLID principles. The principle of sole responsibility. The opening-closing principle. The principle of inversion of dependencies. The principle of substitution of Barbara Liskov. The principle of interface isolation.

7.2. Design templates

General information about design templates. Advantages and disadvantages of using design templates. Design template elements. Classification of GoF design patterns. Application of basic GoF design patterns. GRASP Responsibilities Allocation Templates.

TOPIC 8. SOFTWARE PRODUCT DEPLOYMENT

8.1. Structure of the compilation module. Private and shared compilation modules. Global cache of compilation modules.

8.2. Creation of shared compilation modules. General information about application deployment. Types of deployment. Installation and deployment projects.

TOPIC 9. DATABASE SYSTEMS. BASIC CONCEPTS AND ARCHITECTURE

9.1. Basic concepts of databases. Info logic and data logic levels. The rule of three "no's". Concepts of "information system", "data bank" and "database". Concept of DBMS, their advantages and disadvantages.

9.2. Database architecture. Conceptual level. External level. Internal level.

TOPIC 10. DATA MODELS

10.1. Data modelling.

Concept of data modelling. Classification of models. Hierarchical data model. Hierarchical data structure. Operations on the hierarchical structure, its advantages and disadvantages.

Concept of data modelling. Classification of models. Hierarchical data model. Hierarchical data structure. Operations on the hierarchical structure, its advantages and disadvantages.

10.2. Relational data model

Relational data structure. Basic concepts and definitions.

Relational algebra. Operations of relational algebra. Examples of application of relational algebra. Properties of relational algebra operations.

10.3. Semantic modelling of the subject area

Model "essence - connection". Concept of essence, properties, connection. ER modelling of the subject area. Types of ER diagrams. Types of communication on ER diagrams. Database design using the ER-modelling method.

10.4. The theory of normalization of the relational data model

Anomalies when performing operations in the database. Functional dependencies. Axiomatic of functional dependencies. Logical derivation of functional dependencies. Armstrong's axioms. Lossless decomposition. Heath's theorem. Definition of the first normal form. Incomplete functional dependencies and second normal form. Transitive dependencies and the third normal form. Boyce-Cod normal form. Multi-valued dependencies. Feigin's theorem and the fourth normal form. Connection dependencies are fifth normal form. General normalization procedure. Renormalization of relations

TOPIC 11. SQL LANGUAGE

11.1. Database schema operations. Creating a database. CREATE DATABASE statement. Creating a table. CREATE TABLE statement. Modification of the table. ALTER TABLE statement. Deleting a table. DROP TABLE statement. Deleting a database. DROP DATABASE statement.

11.2. Data retrieval tools. Basic language constructs designed to select data. Expressions, conditions, and operators. Selecting from multiple tables. Use of aggregate functions. GROUP BY, HAVING, ORDER BY phrase. Subqueries

11.3 Data manipulation tools. Adding rows to a table. INSERT statement. Updating data. The UPDATE statements. Deleting table rows. The DELETE statement.

TOPIC 12. DATABASE DESIGN

12.1. Database design methodology. Stages of database design. Analysis of the subject area. Conceptual modelling of the subject area. Logical and physical design.

12.2. Functions of the database design administrator group.

TOPIC 13. TRANSACTIONS AND DATA INTEGRITY

13.1. Data integrity.

The concept of integrity constraints. NULL values and three-valued logic. Integrity of entities and foreign keys. Operations that violate referential integrity. Strategies for maintaining referential integrity.

Domain, attribute, tuple, relation, and database constraints. Integrity Constraints in SQL.

13.2. Transactions and data integrity. Problems of parallel operation of transactions

Concept of transaction. Properties of transactions. Implementation of transactions by means of SQL.

Concept of transaction mix. Problems of parallel operation of transactions. The problem of losing update results. The problem of unfixed dependency (reading "dirty" data, unrepeatable reading). The problem of incompatible analysis.

Implementation of isolation of transactions by means of SQL.

TOPIC 14. CONCEPT AND CLASSIFICATION OF PARALLEL COMPUTING SYSTEMS (PCS). CLASSIFICATION OF PCS. PRINCIPLES OF BUILDING PCS WITH SHARED AND DISTRIBUTED MEMORY

14.1. Concept and classification of parallel computing systems (PCS). Classification of PCS according to Flynn. Multiprocessor and multicomputer systems.

The concept of a parallel computing system (PCS). PCS paradigms: simultaneous execution of one or more instructions on one or more memory areas. Concept of multiprocessor and multicomputer system.

14.2. Principles of building PCS with shared and distributed memory. Types of communication link topologies.

Concept of shared and distributed memory. Types of connection topologies: ring, lattice, star, fully connected graph, hypercube. Main characteristics of topologies and their calculation.

TOPIC 15. SEQUENTIAL AND PARALLEL PROGRAMMING MODELS. PARALLEL PROGRAMMING MODELS: TASK PARALLELISM AND DATA PARALLELISM. BASIC STAGES OF DEVELOPING A PARALLEL PROGRAM (ALGORITHM). MULTI-THREADED APPLICATIONS

15.1. Sequential and parallel programming models. Parallel programming models: task parallelism and data parallelism.

Sequential and parallel programming models.

15.2. Basic stages of developing a parallel program (algorithm). Multi-threaded applications.

Stages of developing a parallel program: definition of global and local variables. A paradigm of message passing between program fragments and results. Multithreading as a means of implementing a parallel program.

Concept of process and flow. Means of data exchange in PCS: messagepassinginsystemswithdistributedmemoryandsharedvariablesinsystemswithsharedmemory.

TOPIC 16. GRAPH MODELS OF EXECUTION OF PARALLEL PROGRAMS

16.1.Graphmodels of execution of parallel programs (operations-operands).

The concept and representation of a graph for the execution of a parallel program. Composition of actions occurring at the vertices of the graph. Composition of actions that occur in the arcs of the graph. Examples of the application of the graph model in the execution of simple programs of parallel calculations.

TOPIC 17. PARALLEL PROGRAMMING TECHNOLOGIES AND LANGUAGES

17.1. Features of parallel programming languages and technologies as a reflection of paradigms based on data parallelism and task parallelism and the use of shared and distributed memory.

Principles of building parallel methods and algorithms: task decomposition into subtasks; separation of information dependencies between half-tasks; scaling of subtasks; distribution of subtasks between system processors. The concept of data parallelism and task parallelism as a technology for working with shared and distributed memory.

TOPIC 18. BASICS OF OPENMP TECHNOLOGY WITH PARALLELIZATION PROGRAMS

18.1. Open MP is a parallelism technology for running multi-threaded applications in shared memory systems.

Principles of organization of execution of the Open MP program. Fork and Join model. Memory model.

TOPIC 19. SOFTWARE FEATURES OF OPENMP PROGRAM DEVELOPMENT

19.1. Directives, Open MP variable classes.

Basic concepts: directives and clauses. The concept of a structural block. Compilation of Open MP programs. Environment variables that control the execution of Open MP programs. Definition of general and local variables, compatible processing of local variables (reduction operation).

19.2. Directives for defining a parallel domain.

Directives for the distribution of calculations within a parallel area: directives for, sections, single. Parallelization by cycles and by sections. Synchronization tools, methods of managing the number of streams in Open MP.

TOPIC 20. PARALLELIZATION TECHNOLOGY (STANDARD) BASED ON THE MPI MESSAGE TRANSFER INTERFACE

20.1. Basic principles of MPI work.

Paired (point-to-point) operations between two processes and collective (collective) communication actions for simultaneous interaction of several processes.

The concept of the structure of a parallel program, operations of initialization and completion of an MPI program.

20.2. Basic MPI functions to support message passing.

Fields of communication and communicators. MPI functions support bilateral and collective operations. Organization of bilateral (point) interactions. Block parallel MPI algorithms. Purpose and composition of collective data transfer operations in MPI.

20.3. Functions to support execution of global operations in MPI.

Purpose and composition of global computing operations on distributed data; composition and purpose of global reduction operations in MPI.

20.4. Management of communicators.

Methods of managing groups of processes and communicators; operations for creating process groups and obtaining information about process groups in MPI.

RECOMMENDED REFERENCES

1. Бази даних у питаннях і відповідях : навчальний посібник / В. В.Чубук, Р. М. Чен, Л. А. Павленко та ін. – Х.: Вид. ХНЕУ, 2004. – 288с.

2. Основи об'єктно-орієнтованого програмування : навчальний посібник / О. В. Щербаков, Ю. Е. Парфьонов, В. М. Федорченко. – Харків : ХНЕУ ім. С. Кузнеця, 2019. – 237 с. [Електронний ресурс]. – Режим доступу: http://repository.hneu.edu.ua/handle/123456789/23847

3. Booch G. Object-Oriented Analysis and Design with Applications / G.Booch, R.A.Maksimchuk et. al. – Boston : Addison-Wesly, 2019. – 717 p.

4. Troelsen A. Pro C# 9 with .NET 5 : Foundational Principles and Practices inProgramming / A. Troelsen, P. Japikse – Berkly : Apress, 2021. – 1353 p.

5. C# documentation [Electronic resource]. – Access mode: https://docs.com/en-us/dotnet/csharp/

6. C# Tutorial [Electronic resource]. – Access mode: https://www.tutorials point.com/csharp/index.htm

7. Тарасов О. В. Використання мови SQL для роботи з сучасними системами керування базами даних / О. В. Тарасов, В. В. Федько, М. Ю. Лосєв. – Х. : Вид. ХНЕУ, 2013. – 348 с.

8. Тарасов О. В. Проектування баз даних : навч. посіб. / О. В. Тарасов, В. В. Федько, М. Ю. Лосєв. – Х. : Вид. ХНЕУ, 2011. – 200 с.

9. Лосєв М.Ю. Бази даних: навчально-практичний посібник для самостійної роботи студентів [Електронний ресурс] / М. Ю. Лосєв, В. В.

Федько. –. Х. : Вид. ХНЕУ, 2019. – 233 с.

10. Харів Н. О. Бази даних та інформаційні системи: навчальний посібник / Н. О. Харів. – Рівне : НУВГП, 2018. – 127 с.

11. Демиденко М.А. Введення в сучасні бази даних: навчальний посібник / М.А. Демиденко. – Д. : НТУ Дніпровська політехніка. –2020. – 138 с.

12. Трофименко О.Г. Організація баз даних : навч. посібник / О. Г. Трофименко, Ю. В. Прокоп, Н. І. Логінова, І. М. Копитчук. – Одеса : Фенікс, 2019. – 246 с.

13. Семеренко В. П. Технології паралельних обчислень : навчальний посібник / В. П. Семеренко. – Вінниця : ВНТУ, 2018. – 104 с.

14. Малашонок Г. І. Паралельні обчислення на розподіленій пам'яті: OpenMPI, Java, Math Partner : підручник / Г. І. Малашонок, А. А. Сідько. – Київ : НаУКМА, 2020. – 266 с.

15. Минайленко Р.М. Паралельні та розподілені обчислення: навч. посіб. / Р.М.Минайленко. – Кропивницький: Видавець Лисенко В. Ф., 2021. – 153 с.

16. Луцків А.М. Паралельні та розподілені обчислення / А.М. Луцків, С.А. Луценко, В. В. Пасічник. – Львів : видавництво «Магнолія», 2015. – 566 с.

17. Коцовський В. М. Теорія паралельних обчислень: навч. посібник. / В. М. Коцовський. – Ужгород: ПП «АУТДОР-Шарк», 2021. – 188 с.

18. Малашонок Г. І. Паралельні обчислення на розподіленій пам'яті: OpenMPI, Java, MathPartner : підручник. / Г. І. Малашонок, А. А. Сідько. – Київ : НаУКМА, 2020. – 266 с.

19. MPI: A Message-Passing Interface Standard Version 3.1. – [Electronic resource]. – Access mode: http://mpi-forum.org/docs/mpi-3.1/mpi31-report.pdf.

20. OpenMP Application Program Interface Version 4.0, July 2013. – [Electronic resource]. – Access mode: http://www.openmp.org/mp-documents/OpenMP4.0.0.pdf.

21. Parallel Programming in MPI and OpenMP. – [Electronic resource]. – Access mode: https://web.corral.tacc.utexas.edu/CompEdu/pdf/pcse/EijkhoutParallelProgramming.pdf

22. Introduction to Parallel Programming with MPI and OpenMP. – [Electronic resource]. – Access mode: https://princetonuniversity.github.io/PUbootcamp/sessions/parallel-programming/Intro_PP_bootcamp_2018.pdf.