



ALYTAUS  
KOLEGIJA



SCHOOL OF BUSINESS  
AND MANAGEMENT OF  
TECHNOLOGY OF BSU



DE MONTFORT  
UNIVERSITY  
LEICESTER



Université  
de Lille  
SCIENCES  
ET TECHNOLOGIES



UNIVERSITY OF  
ECONOMY

Innovative ICT Education for Social-Economic Development (IESED)  
574283-EPP-1-2016-1-LT-EPPKA2-CBHE-JP

# OBJECT-ORIENTED PROGRAMMING

Minsk 2017

## PLAN OF EDUCATIONAL DISCIPLINE

Year of study	Semester	Academic hours					Hours of course work	ECTS	Number of hours
		Total	Lectures	Lab	Practice/ seminar	Independent work			
2	3,4	246	64	64		78	40	9	Full-time
2	3,4	128	16	16		96		9	Part-time

### 1. COMPETENCIES

1. Perform modeling, design of software tools and documentation to support activities in various subject areas.
2. To be able to generate new ideas focusing on creativity, critical thinking, communication and collaboration.
3. Build and optimize models of various systems and processes.

### 2. COURSE GOAL

Training of a specialist with fundamental knowledge and practical skills in the object-oriented programming.

### 3. COMPETENCES FORMED DURING THE IMPLEMENTATION OF THE EDUCATIONAL PROGRAM

- make use of applications to develop software using object-oriented programming;
- model the object-oriented architecture of a software application;
- model the object-oriented architecture of a software application;
- explain the principles of implementing object-oriented programming in the language;
- model the object-oriented architecture of a software application;
- explain programming templates to implement software systems.

### 4. EDUCATIONAL AND METHODOLOGICAL MAP (for full-time education)

Section number	Topics, classes; list of issues to be studied	Number of academic hours				Form of knowledge control
		Total	Lecture	laboratory classes	Independent work	
<b>1.</b>	<b>Java language basics</b>	<b>8</b>	<b>4</b>	<b>4</b>		Final examination
1.1.	Fundamental Programming Structures in Java	1	1			
1.2.	Data types	3	1	2		Laboratory work
1.3.	Operators	4	2	2		Laboratory work
<b>2.</b>	<b>Java platform</b>	<b>5</b>	<b>3</b>	<b>2</b>		Final examination
2.1.	Structure of the <b>Java</b> platform	1	1			
2.2.	The life cycle of the program	4	2	2		Laboratory work
<b>3.</b>	<b>.NET platform</b>	<b>5</b>	<b>3</b>	<b>2</b>		Final examination

3.1.	Structure of .NET platform	1	1			
3.2.	The life cycle of the program	4	2	2		Laboratory work
<b>4.</b>	<b>Object-oriented programming in Java</b>	<b>28</b>	<b>12</b>	<b>16</b>		Final examination
4.1.	Class definition, instancing	2	2			
4.2.	Fields, methods	2	2			
4.3.	Inheritance	3	1	2		Laboratory work
4.4.	Abstraction	5	1	4		Laboratory work
4.5.	Core Collection Classes	6	2	4		Laboratory work
4.6.	String class	3	1	2		Laboratory work
4.7.	Basics of input / output	3	1	2		Laboratory work
4.8.	Exception Handling	3	1	2		Laboratory work
4.9.	Other features	1	1			Final examination
<b>5.</b>	<b>Object-oriented programming in C#</b>	<b>28</b>	<b>12</b>	<b>16</b>		Final examination
5.1.		2	2			
5.2.		2	2			
5.3.		3	1	2		Laboratory work
5.4.		5	1	4		Laboratory work
5.5.		6	2	4		Laboratory work
5.6.		3	1	2		Laboratory work
5.7.		3	1	2		Laboratory work
5.8.		3	1	2		Laboratory work
5.9.		1	1			
<b>6.</b>	<b>Multithreaded programming in Java</b>	<b>6</b>	<b>4</b>	<b>2</b>		Final examination
6.1.	The Java stream model	1	1			
6.2.	Stream control	1	1			
6.3.	Synchronization	4	2	2		Laboratory work
<b>7.</b>	<b>Multithreaded programming in .NET</b>	<b>6</b>	<b>4</b>	<b>2</b>		Laboratory work
7.1.		1	1			
7.2.		1	1			
7.3.		4	2	2		Laboratory work
<b>8.</b>	<b>Unified modeling language</b>	<b>4</b>	<b>4</b>	<b>0</b>		Final examination
8.1.	Basic concepts.	1	1			
8.2.	Diagrams.	3	3			
<b>9.</b>	<b>General Responsibility Assignment Software Patterns</b>	<b>4</b>	<b>2</b>	<b>2</b>		Final examination
9.1.	Basic principles	1	1			
9.2.	Templates	3	1	2		Laboratory work
<b>10.</b>	<b>Gang of Four templates</b>	<b>12</b>	<b>6</b>	<b>6</b>		Final examination
10.1.	Generating design patterns	4	2	2		Laboratory work
10.2.	Structural design patterns	4	2	2		Laboratory work
10.3.	Behavioral design patterns	4	2	2		Laboratory work
<b>11.</b>	<b>Development of a user interface (GUI) in Java</b>	<b>11</b>	<b>5</b>	<b>6</b>		Final examination
11.1.	Overview of Java GUI Libraries Swing, AWT, JavaFX	1	1			
11.2.	GUI framework architecture	4	2	2		Laboratory work
11.3.	Classes of components Controls	6	2	4		Laboratory work
<b>12.</b>	<b>Development of a user interface (GUI) in .NET</b>	<b>11</b>	<b>5</b>	<b>6</b>		Final examination
12.1.		1	1			

12.2.		4	2	2		Laboratory work
12.3.		6	2	4		Laboratory work
TOTAL		128	64	64		

## 5. EDUCATIONAL AND METHODOLOGICAL MAP (for part-time education)

Section number	Topics, classes; list of issues to be studied	Number of academic hours				Form of knowledge control
		Total	Lecture	laboratory classes	Independent work	
<b>1.</b>	<b>Java language basics</b>	<b>8</b>	<b>0,4</b>	<b>0,5</b>	<b>7,1</b>	Final examination
1.1.	Fundamental Programming Structures in Java	1	0,2		0,8	
1.2.	Data types	3	0,2		2,8	Independent work
1.3.	Operators	4		0,5	3,5	Laboratory work
<b>2.</b>	<b>Java platform</b>	<b>5</b>	<b>0,5</b>	<b>1</b>	<b>3,5</b>	Final examination
2.1.	Structure of the <b>Java</b> platform	1	0,5		0,5	
2.2.	The life cycle of the program	4		1	3	Laboratory work
<b>3.</b>	<b>.NET platform</b>	<b>5</b>	<b>0,5</b>	<b>1</b>	<b>3,5</b>	Final examination
3.1.	Structure of <b>.NET</b> platform	1	0,5		0,5	
3.2.	The life cycle of the program	4		1	3	Laboratory work
<b>4.</b>	<b>Object-oriented programming in Java</b>	<b>28</b>	<b>3,1</b>	<b>2,5</b>	<b>22,4</b>	Final examination
4.1.	Class definition, instancing	2	0,2		1,8	
4.2.	Fields, methods	2	0,2		1,8	
4.3.	Inheritance	3	0,5		2,5	Independent work
4.4.	Abstraction	5	0,5	0,5	4	Laboratory work
4.5.	Core Collection Classes	6	1	1	4	Laboratory work
4.6.	String class	3	0,2		2,8	Independent work
4.7.	Basics of input / output	3	0,2	0,5	2,3	Laboratory work
4.8.	Exception Handling	3	0,2	0,5	2,3	Laboratory work
4.9.	Other features	1	0,1		0,9	Final examination
<b>5.</b>	<b>Object-oriented programming in C#</b>	<b>28</b>	<b>3,1</b>	<b>2,5</b>	<b>22,4</b>	Final examination
5.1.	Class definition, instancing	2	0,2		1,8	
5.2.	Fields, methods	2	0,2		1,8	
5.3.	Inheritance	3	0,5		2,5	Independent work
5.4.	Abstraction	5	0,5	0,5	4	Laboratory work
5.5.	Core Collection Classes	6	1	1	4	Laboratory work
5.6.	String class	3	0,2		2,8	Independent work
5.7.	Basics of input / output	3	0,2	0,5	2,3	Laboratory work
5.8.	Exception handling	3	0,2	0,5	2,3	Laboratory work
5.9.	Other features	1	0,1		0,9	
<b>6.</b>	<b>Multithreaded programming in Java</b>	<b>6</b>	<b>0,8</b>	<b>2</b>	<b>3,2</b>	Final examination
6.1.	The Java stream model	1	0,1		0,9	
6.2.	Stream control	1	0,2		0,8	
6.3.	Synchronization	4	0,5	2	1,5	Laboratory work

<b>7.</b>	<b>Multithreaded programming in .NET</b>	<b>6</b>	<b>0,8</b>	<b>2</b>	<b>3,2</b>	Final examination
7.1.	The .NET stream model	1	0,1		0,9	
7.2.	Stream control	1	0,2		0,8	
7.3.	Synchronization	4	0,5	2	1,5	Laboratory work
<b>8.</b>	<b>Unified modeling language</b>	<b>4</b>	<b>1,2</b>	<b>0</b>	<b>2,8</b>	Final examination
8.1.	Basic concepts.	1	0,2		0,8	
8.2.	Diagrams.	3	1		2	
<b>9.</b>	<b>General Responsibility Assignment Software Patterns</b>	<b>4</b>	<b>0,7</b>	<b>0</b>	<b>3,3</b>	Final examination
9.1.	Basic principles	1	0,2		0,8	
9.2.	Templates	3	0,5		2,5	Independent work
<b>10.</b>	<b>Gang of Four templates</b>	<b>12</b>	<b>2,1</b>	<b>0</b>	<b>9,9</b>	Final examination
10.1.	Generating design patterns	4	0,5		3,5	Independent work
10.2.	Structural design patterns	4	0,8		3,2	Independent work
10.3.	Behavioral design patterns	4	0,8		3,2	Independent work
<b>11.</b>	<b>Development of a user interface (GUI) in Java</b>	<b>11</b>	<b>1,2</b>	<b>2</b>	<b>7,8</b>	Final examination
11.1.	Overview of Java GUI Libraries	1	0,2		0,8	
11.2.	GUI framework architecture	4	0,5		3,5	Independent work
11.3.	Classes of components Controls	6	0,5	2	3,5	Laboratory work
<b>12.</b>	<b>Development of a user interface (GUI) in .NET</b>	<b>11</b>	<b>1,2</b>	<b>2</b>	<b>7,8</b>	Final examination
12.1.	Overview of GUI Libraries	1	0,2		0,8	
12.2.	GUI framework architecture	4	0,5		3,5	Independent work
12.3.	Classes of components Controls	6	0,5	2	3,5	Laboratory work
TOTAL		128	16	16	96	

## 6. THEORETICAL CONTENT

№	Topic	Content
<b>1.</b>	<b>Java language basics</b>	
1.1.	Fundamental Programming Structures in Java	Syntax Details. Comments. The main() method. Constants. Statement. Packages. The import Statement. Static Imports. Defining Packages. Package Scope. Namespaces.
1.2.	Data types	Primitive Data types. Variables. Initializations. Assignments. Expressions. Strings. Arrays. Type conversion.
1.3.	Operators	If Statements. Switch Statements. Loop Statements. Enhanced for Loop.
<b>2.</b>	<b>Java platform</b>	
2.1.	Structure of the Java platform	Compiling and Interpreting Applications. Java Virtual Machine. Java Development Kit. Java Runtime Environment.
2.2.	The life cycle of the program	Working with the program in the IDE and using the command line. Project directory structure. CLASSPATH. Import of packages. The structure of the JAR Scope and lifetime of objects. Garbage collector
<b>3.</b>	<b>.NET platform</b>	
3.1.	Structure of .NET platform	Common Language Runtime. just in time compilation, Framework Class Library. Common Language Infrastructure (CLI) specification.
3.2.	The life cycle of the program	Working with the program in the IDE. Project directory structure. Solutions. Garbage Collection. Application manifest. Name spaces and application
<b>4.</b>	<b>Object-oriented programming in Java</b>	
4.1.	Class definition, instancing	Defining a Class. Instancing.
4.2.	Fields, methods	Encapsulation, Restrictions on access to class members. Constructors. Parameterization methods. Using <b>this</b> key-word.

<b>№</b>	<b>Topic</b>	<b>Content</b>
4.3.	Inheritance	Casting. Polymorphism. Overloading and redefining methods. Modifiers for restricting access to elements during inheritance. The notion of superclass and subclass and the organization of connections in inheritance. Static class members
4.4.	Abstraction	The concept and use of abstract classes and interfaces. Use the final keyword when inheriting. Implementing an Interface. Interfaces: Iterable, Runnable.
4.5.	Core Collection Classes	The Collections Framework. Set, Queue, Map, List Interfaces. Set, Queue, Map, List Implementation Classes. Interface Iterable.
4.6.	String class	String operations. Convert strings. Methods
4.7.	Basics of input / output	Overview of Streams, Buffered byte streams. Using stream input / output. Serializable interface. Classes and interfaces of Java input / output streams.
4.8.	Exception Handling	Basic principles of exception handling. Types of exceptions. Using the try and catch clauses. The throw statement. Methods with the keyword throws. The finally block. Built-in Java exceptions. Create your own subclasses of exceptions
4.9.	Other features	Anonymous inner classes. Parameterized classes.
<b>5.</b>	<b>Object-oriented programming in C#</b>	
5.1.	Class definition, instancing	Defining a Class. Instancing. Finalizers. Allocating and deallocating memory for objects
5.2.	Fields, methods	Encapsulation, Access Modifiers and Access Levels. Constructors. Properties, Extension methods
5.3.	Inheritance	Casting. Polymorphism. Overloading and redefining methods. The notion of superclass and subclass and the organization of connections in inheritance. Static class members
5.4.	Abstraction	The concept and use of abstract classes and interfaces. Use the final keyword when inheriting. Implementing an Interface. Interfaces: Iterable, Runnable.
5.5.	Core Collection Classes	The Collections Framework. Set, Queue, Map, List Interfaces. Set, Queue, Map, List Implementation Classes. Interface Iterable.
5.6.	String class	String operations. Convert strings. Methods
5.7.	Basics of input / output	Overview of Streams, Buffered byte streams. Using stream input / output. Serializable interface. Classes and interfaces of Java input / output streams.
5.8.	Exception Handling	Basic principles of exception handling. Types of exceptions. Using the try and catch clauses. The throw statement. Methods with the keyword throws. The finally block. Built-in Java exceptions. Create your own subclasses of exceptions
5.9.	Other features	Nested Classes, Anonymous Types, Generalized classes. Generics. Delegates.
<b>6.</b>	<b>Multithreaded programming in Java</b>	
6.1.	The Java stream model	The Thread class and the Runnable interface. The main thread. Create a thread. Creating multiple threads.
6.2.	Stream control	isAlive () and join () methods. Suspend, resume and stop the flow. Priorities of flows.
6.3.	Synchronization	Message passing, synchronized methods. Blocking. The operator is synchronised
<b>7.</b>	<b>Multithreaded programming in .NET</b>	
7.1.	The Java stream model	Managed Threading. System.Threading namespace
7.2.	Stream control	Starting and Stopping Threads. Thread Properties. Thread Timers. Thread Pooling
7.3.	Synchronization	Foreground and Background Threads. lock Keyword. Monitors. Synchronization Events and Wait Handles
<b>8.</b>	<b>Unified modeling language</b>	

<b>№</b>	<b>Topic</b>	<b>Content</b>
8.1.	Basic concepts.	Diagram types. Basic elements. Packages in the UML language, their graphic representation. Basic semantic constructions of the language, their description. Semantics of relationships between elements
8.2.	Diagrams.	Class diagrams. Use case diagrams. Sequence diagrams. Statechart diagrams. Component diagrams.
<b>9.</b>	<b>GRASP Templates</b>	
9.1.	Basic principles	Low Coupling, High Cohesion
9.2.	Templates	Exert Template. Creator Template. Controller Template
<b>10.</b>	<b>GoF templates</b>	
10.1.	Generating design patterns	Abstract Factor, Builder, Singleton
10.2.	Structural design patterns	Adapter, Composite, Decorator, Façade, Proxy.
10.3.	Behavioral design patterns	Chain of responsibility, Command, Memento, Observer.
<b>11.</b>	<b>Development of a user interface (GUI) in Java</b>	
11.1.	Overview of Java GUI Libraries	Swing, AWT, JavaFX.
11.2.	GUI framework architecture	Event-oriented architecture. Data Binding.
11.3.	Classes of components Controls	Add, align, and position controls. Developing Custom Controls.
<b>12.</b>	<b>Development of a user interface (GUI) in .NET</b>	
12.1.	Overview of Java GUI Libraries	Windows Forms, WPF.
12.2.	GUI framework architecture	Event-oriented application supported by the Microsoft .NET Framework. Windows Forms Data Binding.
12.3.	Classes of components Controls	Add, align, and position controls. Developing Custom Windows Forms Controls

## 7. LABORATORY PRACTICE

<b>№</b>	<b>Topic number</b>	<b>Content</b>
<b>1.</b>	<b>Java language basics</b>	
1.2.	Data types	Creating Java console application for simple calculation.
1.3.	Operators	Creating Java console application for iteration calculation.
<b>2.</b>	<b>Java platform</b>	
2.2.	The life cycle of the program	Installation and setup Java. Installation and setup IDE Eclipse. Working with the program in the IDE and using the command line. Create project directory structure.
<b>3.</b>	<b>.NET platform</b>	
3.2.	The life cycle of the program	Installation and setup .NET framework. Installation and setup Visual Studio. Working with the program in the IDE and using the command line. Create project directory structure.
<b>4.</b>	<b>Object-oriented programming in Java</b>	
4.3.	Inheritance	Creating and using class hierarchy, using inheritance, overloading and redefining methods.
4.4.	Abstraction	Development and implementation of interfaces for creating software architecture.
4.5.	Core Collection Classes	Using collection classes for solving application tasks. Creating own implementations of collection interfaces.
4.6.	String class	Using of String class for text processing.
4.7.	Basics of input / output	Using streaming classes to organize data storage on disk.
4.8.	Exception Handling	Creating and using own Exception classes.
<b>5.</b>	<b>Object-oriented programming in C#</b>	
5.3.	Inheritance	Creating and using class hierarchy, using inheritance, overloading and redefining methods.
5.4.	Abstraction	Development and implementation of interfaces for creating software architecture.
5.5.	Core Collection Classes	Using collection classes for solving application tasks. Creating own implementations of collection interfaces.

<b>№</b>	<b>Topic number</b>	<b>Content</b>
5.6.	String class	Using of String class for text processing.
5.7.	Basics of input / output	Using streaming classes to organize data storage on disk.
5.8.	Exception Handling	Exception handling.
<b>6.</b>	<b>Multithreaded programming in Java</b>	
6.3.	Synchronization	Developing a multi-threaded application using thread synchronization.
<b>7.</b>	<b>Multithreaded programming in .NET</b>	
7.3.	Synchronization	Developing a multi-threaded application using thread synchronization.
<b>9.</b>	<b>GRASP Templates</b>	
9.2.	Templates	Development of an application using GRASP templates.
<b>10.</b>	<b>GoF templates</b>	
10.1	Generating design patterns	Using generating patterns to improve the structure of the application.
10.2	Structural design patterns	Using structural patterns to improve the structure of the application.
10.3	Behavioral design patterns	Using behavioral patterns to improve the structure of the application.
<b>11.</b>	<b>Development of a user interface (GUI) in Java</b>	
11.2.	GUI framework architecture	Creating an event-driven application in GUI framework.
11.3	Classes of components Controls	Create and use own visual control component.
<b>12.</b>	<b>Development of a user interface (GUI) in .NET</b>	
12.2	GUI framework architecture	Creating an event-driven application in GUI framework.
12.3	Classes of components Controls	Create and use own visual control component.

## 8. ASSIGNMENT FOR INDEPENDENT WORK

<b>№</b>	<b>Tasks</b>
1.	Modeling of two-dimensional structures (surfaces, labyrinths, game fields) by means of arrays
2.	Implement a class hierarchy for performing operations on rational fractions.
3.	Working with Date class: conversion, formatting, comparison, other methods.
4.	Create a visual control component called Calculator.
5.	Developing WPF application.
6.	Developing JavaFX application.
7.	Create class using Builder pattern for constructor method.
8.	Create class hierarchy using Composer pattern.
9.	Create application for advantage searching in text files.
10.	Create String extending class with special features (for example: replace the first letters in each word with capital letters)

## 9. SYSTEM OF ASSESSMENT OF KNOWLEDGE AND SKILLS (ACCORDING TO THE NATIONAL REQUIREMENTS)

The evaluation criteria of the results on a ten-point scale.

A ten-point scale, depending on the grade and the mark, includes the following criteria:

10 (ten) points, passed:

- systematized, deep and full knowledge on all sections of the curriculum of the institution of higher education in the academic discipline, as well as on major issues that go beyond its limits;
- accurate use of scientific terminology (including in a foreign language), competent, logically correct statement of the answer to questions;
- perfect mastering of the tools of the academic discipline, the ability to use it effectively in formulation and solution of scientific and professional problems;
- the expressed ability independently and creatively to solve complex problems in non-standard situations;



- complete and profound studying of basic, additional literature on the subject of the discipline;
- the ability to freely navigate in theories, concepts and directions on the discipline and give them an analytical assessment, use the scientific achievements of other disciplines;
- creative independent work on practical, laboratory classes, active creative participation in group discussions, high level of the culture of performance of tasks.

9 (nine) points, passed:

- systematized, deep and full knowledge on all sections of the curriculum of the institution of higher education on the academic discipline;
- accurate use of scientific terminology (including in a foreign language), competent, logically correct statement of the answer to questions;
- mastering of the tools of the academic discipline, the ability to use it effectively in formulation and solution of scientific and professional problems;
- ability independently and creatively to solve complex problems in non-standard situations within the curriculum of the institution of higher education on the academic discipline;
- complete studying of basic, additional literature on the subject of the discipline, recommended by the curriculum of the institution of higher education on the discipline;
- the ability to navigate in theories, concepts and directions on the discipline and give them an analytical assessment;
- Systematic, active independent work on practical, laboratory classes, active creative participation in group discussions, high level of the culture of performance of tasks.

8 (eight) points, passed:

- systematized, deep and full knowledge on all sections of the curriculum of the institution of higher education in the academic discipline in the volume of the curriculum of the institution of higher education on the discipline;
- use of scientific terminology (including in a foreign language), competent, logically correct statement of the answer to questions, the ability to make sound conclusions and generalizations;
- mastering of the tools of the academic discipline (methods of complex analysis, information technology), the ability to use it effectively in formulation and solution of scientific and professional problems;
- ability independently to solve complex problems within the curriculum of the institution of higher education on the academic discipline;
- studying of basic, additional literature, recommended by the curriculum of the institution of higher education on the discipline;
- the ability to navigate in theories, concepts and directions on the discipline and give them an analytical assessment;
- active independent work on practical, laboratory classes, systematic participation in group discussions, high level of the culture of performance of tasks.

7 (seven) points, passed:

- systematized, deep and full knowledge on all sections of the curriculum of the institution of higher education on the academic discipline;
- use of scientific terminology (including in a foreign language), competent, logically correct statement of the answer to questions, the ability to make sound conclusions and generalizations;
- mastering of the tools of the academic discipline, the ability to use it effectively in formulation and solution of scientific and professional problems;
- free possession of generic solutions within the curriculum of the institution of higher education on the academic discipline;

- studying of basic, additional literature, recommended by the curriculum of the institution of higher education on the discipline;
- the ability to navigate in basic theories, concepts and directions on the discipline and give them an analytical assessment;
- independent work on practical, laboratory classes, participation in group discussions, high level of the culture of performance of tasks.

6 (six) points, passed:

- sufficiently full and systematized knowledge in the volume of the curriculum of the institution of higher education on the discipline;
- use of the necessary scientific terminology, competent, logically correct statement of the answer to questions, the ability to make sound conclusions and generalizations;
- mastering of the tools of the academic discipline, the ability to use it effectively in solution of scientific and professional problems;
- ability independently to apply generic solutions within the curriculum of the institution of higher education on the academic discipline;
- studying of basic literature, recommended by the curriculum of the institution of higher education on the discipline;
- the ability to navigate in basic theories, concepts and directions on the discipline and give them a comparative assessment;
- active independent work on practical, laboratory classes, periodic participation in group discussions, high level of the culture of performance of tasks.

5 (five) points, passed:

- sufficient knowledge in the volume of the curriculum of the institution of higher education on the discipline;
- use of scientific terminology, competent, logically correct statement of the answer to questions, the ability to make sound conclusions;
- mastering of the tools of the academic discipline, the ability to use it in solution of scientific and professional problems;
- ability independently to apply generic solutions within the curriculum of the institution of higher education on the academic discipline;
- studying of basic literature, recommended by the curriculum of the institution of higher education on the discipline;
- the ability to navigate in basic theories, concepts and directions on the discipline and give them a comparative assessment;
- active independent work on practical, laboratory classes, periodic participation in group discussions, high level of the culture of performance of tasks;
- independent work on practical, laboratory classes, periodic participation in group discussions, sufficient level of the culture of performance of tasks.

4 (four) points, passed:

- sufficient knowledge within the educational standard of higher education;
- studying of basic literature, recommended by the curriculum of the institution of higher education on the discipline;
- use of scientific terminology, logical statement of the answer to questions, the ability to make sound conclusions;
- ability to draw conclusions without essential errors;
- mastering of the tools of the academic discipline, the ability to use it in solution of standard (typical) tasks;

- ability to solve standard (typical) tasks under the guidance of a teacher;
- ability to navigate in basic theories, concepts and directions on the discipline and give them an assessment;
- work under the guidance of a teacher on practical, laboratory classes, the permissible level of the culture of performance of tasks.

3 (three) points, failed:

- insufficient knowledge within the educational standard of higher education;
- studying of basic literature, recommended by the curriculum of the institution of higher education on the discipline;
- knowledge of a part of the basic literature, recommended by the curriculum of the institution of higher education on the discipline;
- use of scientific terminology, presentation of answers to questions with significant, logical errors;
- weak possession of the tools of the academic discipline, incompetence in solving standard (typical) tasks;
- inability to navigate in basic theories, concepts and directions on the discipline;
- work under the guidance of a teacher on practical, laboratory classes, the permissible level of the culture of performance of tasks.
- passivity on practical, laboratory classes, low level of the culture of performance of tasks.

2 (two) points, failed:

- fragmented knowledge within the educational standard of higher education;
- knowledge of individual literary sources, recommended by the curriculum of the institution of higher education on the discipline;
- inability to use scientific terminology of the academic discipline, the presence in the answer rude, logical errors;
- passivity on practical, laboratory classes, low level of the culture of performance of tasks.

1 (one) point, failed:

- lack of knowledge and (competences) within the educational standard of higher education, failure to answer, failure to appear for attestation without good cause.

## **10. METHODS AND MEANS OF IMPLEMENTATION OF THE CONTENT OF THE EDUCATIONAL PROGRAM AND TRAINING OF EDUCATIONAL, TRAINING AND METHODOLOGICAL MATERIALS**

The training will be conducted using classical methods, project methods and distance learning technologies, implemented on the portal SDO.VSTU.BY. Trainees will be provided with access to the SDO.VSTU.BY portal.

In the classroom, students will learn the discipline directly in the computer class. The following software (software) will be used during the training:

- IDE Eclipse,
- JDK,
- web-browser.

The following tools and technologies were used in the preparation of training, teaching and methodological materials:

- Adobe PDF.
- LMS Moodle.
- Eclipse.
- JDK.

- Программы для записи и редактирования видео.

## 11. THE INFORMATION-METHODOLOGICAL PART

### Main literature

№	Authors	Bibliographic description
1.	Блинов, И.Н., Романчик, В. С.	Java. Методы программирования : уч.-мет. пособие / И. Н. Блинов, В. С. Романчик. — Минск : издательство «Четыре четверти», 2013. — 896 с.
2.	Блинов, И.Н.	Java. Промышленное программирование : практ. пособие / И.Н. Блинов, В.С. Романчик. – Минск : УниверсалПресс, 2007. – 704 с.
3.	Хорстманн, Кей С.	Java. Библиотека профессионала, том 1. Основы. 10-е изд.: Пер. с англ. — М.: ООО "И.Д. Вильямс", 2016. — 864 с.
4. 18.	Хорстманн, Кей С.	Java. Библиотека профессионала, том 2. Расширенные средства программирования, 10-е изд. : Пер. с англ. — СПб. : ООО "Альфа-книга", 2017. — 976 с.
5. 23.	Мурат Йенер, Алекс Фидом	Java EE. Паттерны проектирования для профессионалов. - СПб.: Питер, 2016. - 240 с.:

### Additional literature

6.	Хорстманн, Кей С., Корнелл, Гари.	Java. Библиотека профессионала, том 1. Основы. 9-е изд. : Пер. с англ. — М. : ООО "И.Д.Вильямс", 2014. — 864 с.
7.	Хорстманн, Кей С., Корнелл, Гари.	Java. Библиотека профессионала, том 2. Расширенные средства, 9-е изд. : Пер. с англ. — М.: ООО "И.Д. Вильямс", 2014. — 1008 с.
8.	Блинов, И.Н.	Java 2: практ. рук. / И.Н. Блинов, В.С. Романчик. – Мн.: УниверсалПресс, 2005. – 400 с.
9.	Куликов, С. С.	Работа с MySQL, MS SQL Server и Oracle в примерах : практ. пособие. / С. С. Куликов. — Минск: БОФФ, 2016. — 556 с.
10. 19.	Фаулер, Мартин.	Шаблоны корпоративных приложений. : Пер. с англ. — М. : ООО "И.Д. Вильямс", 2016. — 544 с.