Faculty of «Computer technology and cybersecurity»
Department of «Mathematical and computer modeling»

APPROVED BY
Vice-rector for academic affairs
«International Information
Technology University» JSC
_____ Umarov T.F.
(Signature) (Full name)
«30» 03 2021 y.

_____ 6B06101 _____
(Code of Academic Program)

_____ Computer science _____
(Name of Academic Program)

CATALOGUE OF ELECTIVE DISCIPLINES

2021 year of admission

2021 y.

The catalogue of elective disciplines for the AP 6B06101 Computer science is developed on the basis of the working curriculum of the AP.

The catalogue of elective disciplines was discussed at a meeting of the department of Mathematical and computer modeling

minutes No. № 8 from «05» March 2021 y.

Head of Department _____ Ydyrys A.Zh.
signature Full name, position, degree

CED compiler _____ Satybaldina A.N.
signature Full name, position, degree

The catalogue of elective disciplines was approved at a meeting of the Academic Council of "International Information Technology University" JSC

minutes No. 4 from «30» March 2021 year.

Director of Academic Affairs _____ Mustafina A.K.
signature Full name, position, degree

1 TERMS AND ABBREVIATIONS

1. 1 Academic program is a single set of basic characteristics of education, including goals, results and content of training, the organization of educational process, ways and methods for their implementation and criteria for assessing learning outcomes.
The content of academic program of higher education consists of three cycles of disciplines - general education disciplines (hereinafter - GED), basic disciplines (hereinafter - BD) and core disciplines (hereinafter - CD).
The cycle of GED includes disciplines of the compulsory component (hereinafter - CC), the university component (hereinafter - UC) and (or) the component of choice (hereinafter - COC). BD and CD include disciplines of UC and COC.
1. 2 Catalogue of elective disciplines (CED) is a systematic annotated list of all COC disciplines, for the entire training period, containing a brief description indicating the purpose of study, a summary of main sections and expected learning outcomes. CED reflects the prerequisites and postrequisites of each academic discipline. It should provide the students with the possibility of an alternative choice of elective disciplines for the formation of an individual educational trajectory.

On the basis of academic program and CED, the students develop individual curricula with the help of advisers.

1. 3 Individual curriculum (IC) is a curriculum formed by the students independently with the help of an adviser for each academic year on the basis of the academic program, the catalogue of elective disciplines or modules;

IC defines an individual educational trajectory of each student separately. It includes disciplines and types of educational activities (internship, experimental research, forms of final certification) of the compulsory component (CC), the university component (UC) and the component of choice (COC).
1. 4 Advisor is a teacher who performs the functions of an academic mentor of a student (according to the appropriate academic program) and assists in choosing a learning path (creating an individual curriculum) and mastering the academic program during the training period.

1. 5 The university component is a list of compulsory educational disciplines determined by the university independently for the mastering of the academic program.

1. 6 The component of choice is a list of academic disciplines and the corresponding minimum amounts of academic credits offered by the university and independently chosen by students in any academic period, taking into account their prerequisites and postrequisites.

1. 7 Elective disciplines are educational disciplines that are a part of the university component and the component of choice in the framework of established academic credits, introduced by organizations of education reflecting the individual preparation of students and taking into account the specifics of socio-economic development, the needs of a particular region and established scientific schools.

1. 8 Postrequisites are the disciplines and (or) modules and other types of academic work, the study of which requires knowledge, skills and competencies acquired at the end of the study of this discipline and (or) modules;

1. 9 Prerequisites are the disciplines and (or) modules and other types of educational work containing knowledge, abilities, skills and competencies necessary for the mastering of the studied discipline and (or) modules;

1. 10 Competencies are the ability of the practical use of acquired knowledge and skills in professional activities. .

2 ELECTIVE DISCIPLINES

№	Cycle of discipline	Code of discipline	Name of discipline	Semester	Number of credits	Prerequisites
			2 year			
1	CD COC	CED6501	Optional discipline No. 1 from CED	3	6	
		SFT6531	Programming in Java			
		SFT6541	Programming in C#			
			3 year			
2	CD COC	CED6502	Optional discipline No. 2 from CED	5	5	
		SFT6532	Data Science 1			ICT, Introduction to programming
		SFT6542	Programming on Internet of Things (IOT)			ICT, Introduction to programming
			4 year			
3	CD UC	CED6503	Optional discipline No. 3 from CED	7	5	
		SFT6513	Data Science 2			Data Science 1
		SFT6523	Amazon Web Services Foundations (AWS Foundations)			ICT
		SFT6533	Human-computer interaction			ICT, Introduction to programming
		SFT6543	Parallel programming			Programming in Java, Computer system architecture
4	CD UC	CED6504	Optional discipline No. 4 from CED	8	5	
		SFT6514	Integration of business processes (SAP)			ICT
		SFT6524	Advanced Python			Programming in Python
		SFT6534	Deep learning in Computer vision			Programming in Python
5	CD UC	CED6505	Optional discipline No. 5 from CED	8	5	
		SFT6515	Development of mobile applications on IOS			Object-oriented programming
		SFT6525	Development of mobile applications on Android			Programming in Java

3 DESCRIPTION OF ELECTIVE DISCIPLINES

Description of discipline	
Code of discipline	SFT6531
Name of discipline	Programming in Java
Number of credits (ESTS)	6
Course, semester	2, 3
Department	MCM
Course author (s)	Olzhayev O.M.
Prerequisites	-
Postrequisites	-
The aim of study of a discipline	The course will introduce students to object-oriented programming using Java. Students are expected to know the basics of scalar types (integers, strings, booleans) and fundamental control structures in procedural programming (loops, assignment statements, conditional expressions). Finally, it will include a short introduction to the Java Framework and Java JDBC.
Brief course description (main sections)	This course was designed to introduce the student to the Java language. Java GUI, Java Database will be studied in this course. Java's unique architecture allows programmers to develop a single application that can run smoothly and reliably across multiple platforms. In this hands-on course, students gain extensive experience with Java and its object-oriented features. Students learn to create robust console and graphical applications, and store and retrieve data from relational databases.
Expected Learning Outcomes (knowledge, abilities, skills and competencies acquired by students)	□ Build robust console and graphical applications. □ Understand the concept of OOP, as well as the purpose and principles of using inheritance, polymorphism, encapsulation, and method overloading. □ Determine the classes, objects, members of the class and the relationship between them, necessary for a specific problem. □ Build Java applications using robust OOP techniques (such as interfaces and APIs) and properly structuring the program (for example, using access control identifiers, automatic documentation via comments, handling error exceptions).

Description of discipline	
Code of discipline	SFT6541
Name of discipline	Programming in C#
Number of credits (ESTS)	6
Course, semester	2, 3
Department	MCM
Course author (s)	Zhanabekov Zh.
Prerequisites	-
Postrequisites	-
The aim of study of a discipline	Create a knowledge system about the .NET Framework class library and the object-oriented C # .NET language. Generate knowledge and skills for developing applications using C # .NET. Develop understanding and taking advantage of the .NET platform.
Brief course description (main sections)	The course is designed to develop students' knowledge of some of the tools available in the .NET Framework Class Library. The course will also improve students' knowledge of the C # programming language and teach how to apply object-oriented architecture and design principles to .NET applications written in C # .NET.
Expected Learning Outcomes (knowledge, abilities, skills and competencies acquired by students)	□ Creation of console / window applications in Visual Studio.NET. □ Create and use classes and objects in a C # application. □ Use the concepts of encapsulation, inheritance and polymorphism in console / window applications. □ Handling error process. □ Creation of charts and themes. □ Explain the compiled program documentation.

Description of discipline	
Code of discipline	SFT6532
Name of discipline	Data Science 1
Number of credits (ESTS)	5
Course, semester	3, 5
Department	MCM
Course author (s)	Omarov B.S.
Prerequisites	ICT, Introduction to programming
Postrequisites	Data Science 2
The aim of study of a discipline	The aim of the course is to provide the untrained student with an initial understanding of the principles of data mining. Along with the practical goal, the course implements educational goals that contribute to expanding the horizons of students, improving their general culture and education in the field of data mining.
Brief course description (main sections)	This course introduces students to the study of data management and data transformation. Students will have some common challenges that arise when working with data. These tasks range from assembling different datasets into more usable forms and how to apply functions to different parts of the datasets..
Expected Learning Outcomes (knowledge, abilities, skills, and competencies acquired by students)	As a result of studying the discipline, the student will be able to: • create a "dataset" from various data stores, for example, a relational model or text on the Internet, taking into account its structure and semantics, in order to build hypotheses and interpret the results . • prepare the database using denormalization, assembly and discretization.

Description of discipline	
Code of discipline	SFT6542
Name of discipline	Programming on Internet of Things (IOT)
Number of credits (ESTS)	5
Course, semester	3, 5
Department	MCM
Course author (s)	
Prerequisites	ICT, Introduction to programming
Postrequisites	-
The aim of study of a discipline	The focus will be on the opportunities offered by various technologies and creative thinking techniques for finding innovative applications of combinations of such technologies in real-world scenarios.
Brief course description (main sections)	The Internet of Things (IOT) is a course about a new paradigm for the interaction of objects with people, information systems and other objects. The course will focus on creative thinking and practical project development.
Expected Learning Outcomes (knowledge, abilities, skills and competencies acquired by students)	As a result of studying the discipline, the student learns about: - IoT concepts - IoT standards - Components of the Internet of Things system. - Relevance of the Internet of Things for the future. - IoT applications. - IOT for smart cities - Challenges in the implementation of the Internet of Things.

Description of discipline	
Code of discipline	SFT6513
Name of discipline	Data Science 2
Number of credits (ESTS)	5
Course, semester	4, 7
Department	MCM
Course author (s)	Amantayeva A.B.
Prerequisites	Data Science 1
Postrequisites	-
The aim of study of a discipline	The aim of the course is to provide the untrained student with an initial understanding of the principles of data mining. Along with the practical goal, the course implements educational goals that contribute to expanding the horizons of students, improving their general culture and education in the field of data mining.
Brief course description (main sections)	This course introduces students to the study of data management and data transformation. Students will have some common challenges that arise when working with data. These tasks range from assembling different datasets into more usable forms and how to apply functions to different parts of the datasets..
Expected Learning Outcomes (knowledge, abilities, skills and competencies acquired by students)	As a result of studying the discipline, the student will be able to: • identify the characteristics, options, advantages and limitations of controlled classification methods with statistical support; • describe the characteristics of instance-based unsupervised methods; • Conduct training assessments using error assessment supported by training, validation and testing set concepts; compare models and present results.

Описание дисциплины	
Code of discipline	SFT6523
Name of discipline	Amazon Web Services Foundations (AWS Foundations)
Number of credits (ESTS)	5
Course, semester	4, 7
Department	IS
Course author (s)	Senior-lecturer Maulenov Ye.S., PhD, assoc.prof. Kassymova A.B.
Prerequisites	ICT
Postrequisites	-
The aim of study of a discipline	The course is designed for students seeking a general understanding of cloud computing concepts, regardless of specific technical roles. It provides an in-depth overview of cloud concepts, core AWS services, security, architecture, pricing, and support. The course can be recommended not only for students of technical specialties, but also for students of business and management specialties. After completing this course, you will be recommended to pass the AWS Certified Cloud Practitioner exam and, after successfully passing, become the owner of the AWS International Certificate (https://aws.amazon.com/certification/certified-cloud-practitioner/). As IITU is an Amazon AWS Academy partner, you will receive 50% off your first exam attempt and free access to the paid practice exam.
Brief course description (main sections)	The following are the main topics / sections that will be covered in the course: 1) Overview of cloud concepts 2) Cloud economy and billing 3) Overview of the global AWS infrastructure 4) Cloud security 5) Networking and content delivery 6) Computing 7) Storage 8) Databases 9) Cloud architecture 10) Automatic scaling and monitoring
Expected Learning Outcomes (knowledge, abilities, skills and competencies acquired by students)	Upon completion of this course, students will be able to: - Define the AWS Cloud - Explain the AWS Pricing Philosophy - Define the components of the global AWS infrastructure - Describe security and compliance measures in the AWS Cloud, including AWS Identity and Access Management (IAM) - Create a Virtual Private Cloud (VPC) with Amazon Virtual Private Cloud (Amazon VPC) - Demonstrate when to use Amazon Elastic Compute Cloud (Amazon EC2), AWS Lambda, and AWS Elastic Beanstalk - Explain the differences between Amazon Simple Storage Service (Amazon S3), Amazon Elastic Block Store (Amazon EBS), Amazon Elastic File System (Amazon EFS), and Amazon Simple Storage Service Glacier (Amazon S3 Glacier) - Demonstrate when to use AWS database services, including Amazon Relational Database Service (Amazon RDS), Amazon DynamoDB, Amazon Redshift, and Amazon Aurora - Explain the architectural principles of the AWS Cloud - Examine and understand the key concepts associated with elastic load balancing Ki: Amazon CloudWatch and Amazon EC2 Auto Scaling. Bonus / Outcome: You will receive a 50% discount on the AWS Certified Cloud Practitioner official exam.

Description of discipline	
Code of discipline	SFT6533
Name of discipline	Human-computer interaction
Number of credits (ESTS)	5
Course, semester	4, 7
Department	MCM
Course author (s)	Bayekova G.Ye.
Prerequisites	ICT, Introduction to programming
Postrequisites	-
The aim of study of a discipline	The goal of this course is to introduce students to the concept of designing systems that can effectively interact with people. The field of human-computer interaction includes the understanding and creation of methods and artifacts that improve human life, objectives, goals and social environment through teaching design, computer science, behavioral and social sciences. In this course, students will explore the principles of design and human behavior, as well as empirical research methods used to solve real-world problems in the design and use of technology.
Brief course description (main sections)	The course provides students with the opportunity to work independently as well as in small teams to solve design problems and use HCI techniques and principles to model problems, create solutions, and study the impact of their projects.
Expected Learning Outcomes (knowledge, abilities, skills and competencies acquired by students)	Course learning outcomes • Use HCI tools, techniques, and concepts to design systems that can effectively interact with people. • Use principles of design and human behavior, computer science and empirical research methods used to solve real-life problems in the design and use of technology. • Design user interfaces from a user perspective, creating designs that support the existing beliefs, attitudes, and behaviors of intended users associated with the tasks the system is designed to support. • Use an iterative design process to develop interfaces that provide a more efficient and satisfying user experience. • Design, plan and conduct a usability test and use the test results to make recommendations for design improvement.

Description of discipline	
Code of discipline	SFT6543
Name of discipline	Parallel programming
Number of credits (ESTS)	5
Course, semester	4, 7
Department	MCM
Course author (s)	Alpar S.D.
Prerequisites	Programming in Java, Computer system architecture
Postrequisites	-
The aim of study of a discipline	The goal of mastering the discipline "Parallel programming" is to develop students' theoretical knowledge and practical skills in programming parallel and distributed systems. Considerable attention is paid to issues related to the development of basic knowledge in the field of architecture of modern multiprocessor computing systems, parallel information processing, technologies for organizing parallel computing on multiprocessor computing systems with distributed or shared RAM.
Brief course description (main sections)	The following are the main topics / sections that will be covered in the course: thread creation, synchronization, common errors, profiling, thread pools and templates, clusters, memory models, linearizability.
Expected Learning Outcomes (knowledge, abilities, skills and competencies acquired by students)	As a result of mastering the discipline, the student must: - Know the basic approaches to the development of parallel programs; basic technologies and models of parallel programming; methods of creating parallel programs for typical multithreaded programming tasks. - Be able to create parallel programs for computing systems with distributed, shared RAM; parallelization of computational algorithms; build a model for executing parallel programs; evaluate the efficiency of parallel computing; analyze the complexity of computations and the possibility of parallelization of the developed algorithms; apply general schemes for the development of parallel programs for the implementation of their own algorithms; evaluate the main parameters of the resulting parallel programs, such as speedup, efficiency, and scalability. - Have the skills (gain experience) of creating parallel programs for computing systems with distributed, shared RAM; constructing parallel analogs of computational algorithms.

Description of discipline	
Code of discipline	SFT6514
Name of discipline	Integration of business processes (SAP)
Number of credits (ESTS)	5
Course, semester	4, 8
Department	MCM
Course author (s)	Karashbayeva Zh.O.
Prerequisites	ICT
Postrequisites	-
The aim of study of a discipline	To acquaint students with important branches of the ERP system and its applications. Develop your knowledge and skills in SAP ERP systems in a way that encourages confidence and ensures satisfaction and pleasure. Develop an understanding of core principles and a high appreciation for SAP. During the educational process, students should familiarize themselves with and be able to apply methods and tools to solve various problems..
Brief course description (main sections)	An Enterprise Resource Planning (ERP) system is software that manages all business areas of an organization, including Accounting and Finance, Human Resources, Sales and Distribution, Manufacturing, Purchasing and Inventory. It's cross-functional, process-oriented, real-time, and based on industry best practices - from service to manufacturing to nonprofit. It is important that business and systems engineers have a working knowledge of these systems as they will be ERP users, auditors, consultants and / or developers in their careers. This course covers ERP theory and practice. The course content includes the evolution of ERP systems, business process reengineering, process mapping, ERP life cycle, ERP functionality, as well as audit and risk issues.
Expected Learning Outcomes (knowledge, abilities, skills and competencies acquired by students)	• Understanding the technical aspects of ERP systems • Learn the concepts of reengineering and how they relate to the implementation of ERP systems. • Be able to map business processes using process mapping methods in SAP • Understand the activities and activities within the life cycle of a SAP system • Be able to identify and describe the typical functionality of an ERP system • Get hands-on experience with SAP transaction flow and SD, FI, CO, PP, HR, MM configuration.

Description of discipline	
Code of discipline	SFT6524
Name of discipline	Advanced Python
Number of credits (ESTS)	5
Course, semester	4, 8
Department	MCM
Course author (s)	Nurtas M.
Prerequisites	Programming in Python
Postrequisites	-
The aim of study of a discipline	Learn to write relatively advanced, well-structured computer programs in Python; Be familiar with the principles and techniques of optimizing the performance of Python numeric applications; have an understanding of parallel computing and how parallel applications can be written in Python; Experiment with developing GPU-accelerated Python applications develop Python applications that use big data services such as Hadoop and Spark.
Brief course description (main sections)	In this course, we will look at a number of best practices for improving the performance of Python programs, including parallel computing and GPU acceleration. We'll also look at how Python can be used to analyze big data using frameworks such as Apache Hadoop and Apache Spark. Students will have the opportunity to use these techniques and gain hands-on experience in developing cutting-edge Python applications.
Expected Learning Outcomes (knowledge, abilities, skills and competencies acquired by students)	As a result of mastering the discipline, the student must: Be able to write relatively advanced, well-structured computer programs in Python Be familiar with the principles and techniques of optimizing the performance of Python numeric applications Understand parallel computing and how parallel applications can be written in Python Experiment with developing GPU-accelerated Python applications Develop Python applications using big data services such as Hadoop and Spark

Описание дисциплины	
Code of discipline	SFT6534
Name of discipline	Deep learning in Computer vision
Number of credits (ESTS)	5
Course, semester	4, 8
Department	MCM
Course author (s)	Nurtas M., Alimbekov A.
Prerequisites	Programming in Python
Postrequisites	-
The aim of study of a discipline	The goal of this course is to provide students with a basic understanding of modern neural networks and their applications in computer vision and natural language understanding.
Brief course description (main sections)	Deep learning has added a huge boost to the already rapidly developing field of computer vision. With deep learning, many new applications of computer vision techniques have been introduced and are now becoming a part of our daily life. These include face recognition and indexing, photo styling, or machine vision in self-driving cars. We will cover both image and video recognition, including image classification and annotation, object recognition and image search, various object detection methods, motion estimation, object tracking in video, human action recognition and finally image styling, editing and the next generation. images. During the project, students will learn how to create a facial recognition and manipulation system in order to understand the internal mechanics of this technology, probably the most famous and often demonstrated in films and TV shows example of computer vision and AI.
Expected Learning Outcomes (knowledge, abilities, skills and competencies acquired by students)	Chatterbot, Tensorflow, Deep Learning, Natural Language Processing

Description of discipline	
Code of discipline	SFT6515
Name of discipline	Development of mobile applications on IOS
Number of credits (ESTS)	5
Course, semester	4, 8
Department	IS
Course author (s)	
Prerequisites	Object-oriented programming
Postrequisites	-
The aim of study of a discipline	The goal of this course is to teach students how to design, implement, test, debug, and publish applications for smartphones on iOS. Students will learn how to bring their innovative ideas from concept to the AppStore through a series of rigorous hands-on programming assignments and group projects. This is an introductory course designed for senior students with Objective-C programming experience. However, there are a significant number of programs in this course that require commitment on the part of the student. The objectives of the course are as follows: - use the iOS development environment; - Explain the key programming paradigms; - Explore the design of the user interface, including views and actions; - demonstrate the persistence of data, including SQLite; - study content providers; - training in messaging and networking; phone sensors, location-based services, background services; - give broadcast receivers.
Brief course description (main sections)	IOS programming concepts are reinforced by a set of topical programming exercises that introduce these topics and gradually allow the student to build a complex application; that is, the programming labs form a set of components that together implement a continuous sensing application. The resulting phone app allows the user to log their exercises and display them on Google maps.
Expected Learning Outcomes (knowledge, abilities, skills and competencies acquired by students)	Students who successfully complete this course will be able to: - be competent in the characteristics and architecture of mobile applications; - be competent in understanding the enterprise-wide requirements for mobile applications; - be competent in the design and development of mobile applications using one application development platform.

Описание дисциплины	
Code of discipline	SFT6525
Name of discipline	Development of mobile applications on Android
Number of credits (ESTS)	5
Course, semester	4, 8
Department	CEIS
Course author (s)	Dauletbek E.T.
Prerequisites	Programming in Java
Postrequisites	-
The aim of study of a discipline	The goal of this course is to teach students how to design, implement, test, debug, and publish Java-based smartphone applications on Android phones. Students will learn how to bring their innovative ideas from concept to the Android market through a series of rigorous hands-on programming assignments and group projects. This is an introductory course designed for senior students with Java programming experience. However, there are a significant number of programs in this course that require commitment on the part of the student. The objectives of the course are as follows: - use the Android development environment, including the Android Studio IDE; - Explain the key programming paradigms; - Explore the design of the user interface, including views and actions; - demonstrate the persistence of data, including SQLite; - study content providers; - training in messaging and networking; phone sensors, location-based services, background services; - give broadcast receivers; - illustrate cloud programming with App Engine; - teach to publish applications on the Android Market.
Brief course description (main sections)	Android programming concepts are reinforced by a set of topical programming exercises that introduce these topics and gradually allow the student to build a complex application; that is, the programming labs form a set of components that together implement a continuous sensing application. The resulting phone app allows the user to log their exercises and display them on Google maps.
Expected Learning Outcomes (knowledge, abilities, skills and competencies acquired by students)	Students who successfully complete this course will be able to: - be competent in the characteristics and architecture of mobile applications; - be competent in understanding the enterprise-wide requirements for mobile applications; - be competent in the design and development of mobile applications using one application development platform.