Faculty of Computer Technology and Cybersecurity
Department of Computer Engineering

Approved
Dean of faculty
IITU JSC
_____N.A. Seilova
«31» __August____ 2023

SYLLABUS
(ACADEMIC PROGRAM)

Course (code, title): SFT6301 Algorithmization and Programming

Major (code, title): B057 Information Technology

Educational program: 6B06106 Computer Systems and Software Engineering, 6B06107 Cyberphysical Systems, 6B06110 Software Engineering, 6B06118 Immersive Technologies

Year: 1 **Semester:** 1 **Number of credits**: 6 ECTS

Lectures: 15 hours

Laboratory classes: 30 hours

Practical classes: 15 hours

T/SIS: 120 hours
Total: 180 hours
Cycle: (GER (university disciplines), BS, AS, electives) university disciplines

Final assessment form: Examination

Almaty 2023

Academic Program of the course «Algorithmization and Programming» has been developed on the basis of the State Standard for Higher Education.

Academic program has been reviewed at the meeting of Computer Engineering department.

Minutes №. _1_ dated «_28_» August 2023

Head of the Department _____ PhD, ass.prof., Chinibayeva T.T.

Authors _____ senior lecturer A.A. Balgabek,
senior lecturer N.M. Kurzhangulov

The working academic program was approved at the meeting of the faculty academic quality council faculty of Computer Technology and Cybersecurity

Minutes №_1 dated "31_" _August_ 2023

Department _____ A.Ajibayeva
Signature
for Educational and Methodological Affairs

1. GENERAL INFORMATION	
Faculty	Computer Technology and Cybersecurity
Major code and title	B057 Information Technology
Educational program code and title	6B06106 Computer Systems and Software Engineering, 6B06107 Cyberphysical Systems, 6B06110 Software Engineering, 6B06118 Immersive Technologies
Year, semester	1 year, 1 semester
Subject category	Compulsory, Elective, Profiling
Number of credits (ECTS)	6 ECTS
Prerequisites	
Postrequisites	"Operating Systems", "SDP4-Performance, Data Structures and Algorithms", "Information theory", "Organization of information systems and networks", "Instrumental tools for program elaboration"
Lecturer	Askar A. Balgabek, senior lecturer office 409 a.balgabek@iitu.edu.kz Nursultan M. Kurzhangulov, senior lecturer office 409 n.kurzhangulov@iitu.edu.kz

2. GOALS, OBJECTIVES AND LEARNING OUTCOMES OF THE COURSE

Course goal is the formation of knowledge on the basics of algorithms and using them to develop programs.

The objectives of the course are to study:

- basics of programming in C++ language
- sorting and searching algorithms;
- concept of Big O notation;
- concept of pointer and dynamic memory;

Learning outcomes of the course

Students successfully completing the course will be able to:

- write a simple program in C++ language;
- give an idea of the using correct algorithm of the program;
- introduce the basic concepts of algorithmics and working concepts of the C++ language;
- compare different algorithms regard to their memory space and compilation time;
- manage the memory using pointers;
- elaborate necessary data structures depending on requested task;
- analyse the efficiency of algorithms.

3. Course description

The purpose of studying the discipline "Algorithmization and programming" is to help develop students with algorithmic thinking. It teaches to solve a difficult task, in particular, non-technical or non-mathematical origin, to obtain, prepare and analyze the results of their solving, to draw conclusions about the achievement of the goal and the correctness of the planned actions. It uses

the principles of problem-oriented and object-oriented approaches not only to solve tasks in computer science or other subjects but also in everyday activities. Formation of the knowledge and skills necessary for solving problems using a personal computer and modern software.

4. COURSE POLICY

Attendance should be regular. The student gains points for each performed assignment, including practices, quizzes and course work. If a student misses lab, he/she can attend a class with another group at the same week. If you miss the class and did not attend with another group, you must defend the practice of that week in order to get points. If the student misses more that 20% of the lessons without reasonable excuse, he/she will not be able to pass the final exam.

The midterm and end of term quizzes are paper-based. The course work is implemented individually and must be defended. The final exam consists of a test and a practical part.

Online platform: Microsoft Teams, Platonus

5. LITERATURE

Basic literature:

1. 1. Introduction to Programming with C++ (3rd Edition), Y. Daniel Liang, 2015
2. C++ Fundamentals, Francesco Zoffoli, 2019
3. C++ Crash Course: A Fast-Paced Introduction, Josh Lospinoso, 2019

Supplementary literature:

1. Thinking In C++. Second Edition by Bruce Eckel, 1995
2. The Art of Computer Programming. 3: Sorting and Searching (2nd ed.), 1998

6. Course schedule

Week/ date	Course topics	References	Lectures (h/w)	Practical sessions (h/w)	Lab. sessions (h/w)	TSIS (h/w)	SIS (h/w)
1	Introduction to C++ programming	B[1],S[2]	1	1	2	1	7
2	If...else selection statement and logical operators (&&, ||, <, >, ==)	B[1],S[2]	1	1	2	1	7
3	Loops: for, while, do…while	B[1],S[2]	1	1	2	1	7
4	The basics of array	B[1], B[3]	1	1	2	1	7
5	Sort and Big-O notation	B[1], B[3]	1	1	2	1	7
6	Multidimensional Arrays	B[1], B[3]	1	1	2	1	7
7	Introduction to functions	B[3], S[2]	1	1	2	1	7
8	Recursion	B[1], B[3], S[2], S[3]	1	1	2	1	7
9	Introduction to pointers	B[3], B[4], S[2]	1	1	2	1	7
10	Double pointers	B[3], B[4], S[2]	1	1	2	1	7
11	Vector and String	B[3], B[4], S[2]	1	1	2	1	7
12	Introduction to structures	B[4], S[2]	1	1	2	1	7
13	Nested structures	B[3], B[4], S[2]	1	1	2	1	7
14	Streams, file input/output stream	B[3], B[4], S[2]	1	1	2	1	7
15	Associative Containers		1	1	2	1	7
	Total hours:		**15**	**15**	**30**	**15**	**105**

7. List of topics/ assignments for laboratory classes

№	Topic Title	Number of hours	References	Form of reporting	Deadline
1	2	3	4	5	6
1	Writing simple program, variables and data types, libraries, system functions	3	B[1],S[2]	Report	Week 2
2	Selection statement. If-Else statement. Switch case statement, logical operator	3	B[1],S[2]	Report	Week 3
3	Loops. For, while, do...while statements.	3	B[1],S[2]	Report	Week 4
4	One-dimensional arrays	3	B[1], B[3], S[2]	Report	Week 5
5	Searching algorithms, bubble sort, selection sort	3			Week 6
6	Multi-dimensional arrays, nested cycles	3	B[1], B[3], S[2]	Report	Week 7
7	Functions, return statement, void function, function overloading	3	B[1], B[3], S[2]	Report	Week 8
8	Usage of Recursion logic algorithms	3	B[3], S[2]	Report	Week 9
9	Pointers, iteration, dynamic memory, resizing arrays.	3	B[1], B[3], S[2], S[3]	Report	Week 10
10	Double pointing, memory control	3	B[3], B[4], S[2]	Report	Week 11
11	String, string functions, vectors	3	B[3], B[4], S[2]	Report	Week 12
12	Structures	3	B[4], S[2]	Report	Week 13
13	Nested structures, array of structures	3		Report	Week 14
14	Working with txt files, parsing	3	B[3], B[4], S[2]	Report	Week 15
15	Associative Containers	3			
	Total	45			

8. List of topics/assignments for Defence

№	Topic/Assignment title	Number of hours	References	Form of reporting	Deadline
1	2	3	4	5	6
1	Basic math, conditional logic, loops	3		Written assignment	Week 4
2	Arrays, sorting algorithims and complexity	2		Written assignment	Week 6
3	Pointers, double pointers	5		Written assignment	Week 11
4	Vector, string, struct	5		Written assignment	Week 13
	Total	15			

9. System for evaluating student performance in a discipline:

Period	Assignments	Score	Total
1st attestation	**Laboratory works:**	**35**	**100**
	Lab work 1,	5	
	Lab work 2,	5	
	Lab work 3,	5	
	Lab work 4,	5	
	Lab work 5,	5	
	Lab work 6,	5	
	Lab work 7,	5	
	Exams:	**30**	
	Defence 1	15	
	Defence 2	15	
	Mid-term exam	**35**	
2nd attestation	**Laboratory works:**	**35**	**100**
	Lab work 8,	5	
	Lab work 9,	5	
	Lab work 10,	5	
	Lab work 11,	5	
	Lab work 12,	5	
	Lab work 13,	5	
	Lab work 14,	5	
	Exams:	**30**	
	Defence 3	15	
	Defence 4	15	
	End-term exam	**35**	
Exam			**100**
Total	**0,3*1stAtt+0,3*2ndAtt+0,4*Final**		

*If the number of absences exceeds 20%, student will be automatically scheduled for a Retake (summer semester)

10. Assessment criteria:

Option 1
Example of assessment criteria on a 5-point scale for laboratory works:

Points	Assessment criterion
5	The work was completed in full and correct answers were received for additional questions from the teacher within the framework of the program.
4	The work was completed in full, but mistakes were made when answering additional questions from the teacher.
3	The work was completed in full, correct conclusions were made, however, there are some non-compliance with design requirements, for example, errors in the design of graphs, tables, or in recording measurement results. After teacher's instructions, these shortcomings are eliminated.
2	The work was performed in an incomplete volume, for example, fault calculations were not carried out or carried out incorrectly, some results are incorrect, the conclusions do not correspond to reality, there are significant errors in the graphical data. After teacher's instructions, the main shortcomings were eliminated, and the graphs were corrected.
1	Work is performed in an incomplete volume, for example, there are errors in the calculations of most or all of the desired values, no faults, the results are mostly

	present, but not true, the conclusions do not correspond to reality, there are significant errors in the design, there are no graphs, calculation formulas are not specified, etc. After teacher's instructions, the main shortcomings are eliminated.
0	The work is incomplete, for example, there are errors in the calculations of most or all of the required values, there are no faults, the results are mostly present, but not true, the conclusions do not correspond to reality, there are significant errors in the design, there are no graphs, calculation formulas are not specified, etc.

Option 2

The point-rating letter system for assessing the educational achievements of students with their interpretation in the traditional grading scale:

Letter Grade	Numerical equivalent	Points (%)	Traditional system assessment	General description of grading criteria
A	4,0	95-100	Excellent	The student has knowledge of the subject in the full scope of the curriculum, understands the discipline deeply enough; shows a high level of knowledge that exceeds the volume provided by the syllabus, gives an exhaustive answer
A-	3,67	90-94		The student has knowledge of the subject in the full scope of the curriculum, understands the discipline deeply enough; gives an exhaustive answer
B+	3,33	85-89	Good	The student shows a complete, well-founded knowledge of the subject, but the answers did not always highlight the main idea, rational methods of calculation were not always used; the answers were mostly brief and sometimes unclear.
B	3,0	80-84		
B-	2,67	75-79		
C+	2,33	70-74		
C	2,0	65-69	Satisfactory	The student demonstrates sufficient knowledge of the subject, but without proper depth and justification, the answers are unclear and without proper logical sequence.
C-	1,67	60-64		
D+	1,33	55-59		
D	1,0	50-54		
FX	0,5	25-49	Unsatisfactory	The student demonstrates insufficient knowledge of the subject, positive answers were not given to individual questions.
F	0	0-24		The student demonstrates a very low level of knowledge of the subject.

11. Assessment and evaluation materials (exam questions)

- Form of assessment (exam): written.
- Example of examination question

Task #1(40 points)
Given a matrix N by M
Fill matrix by positive and negative numbers
You need to **zero out** (transform to a zero) all negative elements, except for minimum

Task #2(30 points)
create structure Candidate
name (string with spaces. Ex: Fiona Mccenley)
work (string. Ex: businessman)
income (int)

We have a private elite club
Create an array of candidate to enter our super private party
There are some rules
We do not like banker
We do not take people that take less than 1000$
We do not like people with surname "Johnson" or "Edisson"
(Note that name and surname is one string. You need to divide it in the program)
Print the list of candidates, which we can pass to the club

Task #3(30 points)
create structure Investor
name (string)
salary (int)
invest_free (boolean)

create structure Company
company_name (string)
cost (int)
investors_number (int)
Investor* investors

We have some companies, that need money to be created.
Each company has an amount of money that needs to be collected **(cost)**
Each company have an array of investors **(investors)**, with some size **(investors_number)**
Investor has name and salary. Each of them, pay 10 percent of their salary per year to create company
Also, there is an investors that don't pay anything **(invest_free == true)**
We have some companies
You need to list companies that can collect enough money (cost) from their investors by the 3 years