

Документ подписан простой электронной подписью
 Информация о владельце:
 ФИО: Котова Лариса Анатольевна
 Должность: Директор филиала
 Дата подписания: 01.06.2026 19:21:48
 Уникальный программный ключ:
 10730ffe6b1ed036b744b6e9d97700b86e5c04a7

МИНИСТЕРСТВО НАУКИ И ВЫСШЕГО ОБРАЗОВАНИЯ РОССИЙСКОЙ ФЕДЕРАЦИИ
Федеральное государственное автономное образовательное учреждение высшего образования
«Национальный исследовательский технологический университет «МИСИС»
Новотроицкий филиал

Приложение 4

к ОПОП ВО 09.03.03 Прикладная информатика
 Прикладная информатика в технических системах

Рабочая программа дисциплины

Алгоритмизация и программирование

Закреплена за подразделением	Кафедра математики и естествознания (Новотроицкий филиал)	
Направление подготовки	09.03.03 Прикладная информатика	
Образовательная программа	09.03.03 Прикладная информатика / Прикладная информатика в технических системах	
Квалификация	Бакалавр	
Форма обучения	очная	
Общая трудоемкость	8 ЗЕТ	Виды контроля в семестрах:
Часов по учебному плану	288	зачет 1 экзамен 2 контрольная работа 1,2

Распределение часов дисциплины по семестрам

Семестр (<Курс>.<Семестр на курсе>)	1 (1.1)		2 (1.2)		Итого	
	19	18				
Неделя	уп	рп	уп	рп	уп	рп
Вид занятий	уп	рп	уп	рп	уп	рп
Лекции	17	17	17	17	34	34
Лабораторные	34	34	34	34	68	68
В том числе инт.	17	17	17	17	34	34
Итого ауд.	51	51	51	51	102	102
Контактная работа	51	51	51	51	102	102
Сам. работа	57	57	102	102	159	159
В том числе сам. работа в рамках ФОС		14		14		
Часы на контроль			27	27	27	27
Итого	108	108	180	180	288	288

Программу составил(и):

к.т.н., Доцент, Леднов А. В.

Рабочая программа дисциплины

Алгоритмизация и программирование

Составлен на основании учебного плана:

09.03.03_23_Прикладная информатика_ПрПИвТС.plx, утвержденного Ученым советом НИТУ МИСИС в составе соответствующей ОПОП ВО 09.03.03 Прикладная информатика Прикладная информатика в технических системах протокол от 27.11.2025 №68.

Рабочая программа одобрена на заседании

Кафедра математики и естествознания (Новотроицкий филиал)

Протокол от 11.03.2026 г., №3.

Руководитель подразделения Швалёва Анна Викторовна.

1. ЦЕЛИ ОСВОЕНИЯ

1.1	Формирование у обучаемых системного представления о теоретических основах
1.2	информационно-технических дисциплин, основ функционирования программного обеспечения ЭВМ, приобретение практических умений разработки алгоритмов и программ

2. МЕСТО В СТРУКТУРЕ ОБРАЗОВАТЕЛЬНОЙ ПРОГРАММЫ

Блок ОП:		Б1.О
2.1	Требования к предварительной подготовке обучающегося:	
2.2	Дисциплины (модули) и практики, для которых освоение данной дисциплины (модуля) необходимо как предшествующее:	
2.2.1	Языки и среды разработки интернет-приложений	
2.2.2	Теоретическая механика	
2.2.3	Подготовка к процедуре защиты и защита выпускной квалификационной работы	
2.2.4	Теория вероятностей и математическая статистика	
2.2.5	Информационные системы и технологии	
2.2.6	Технологии программирования	
2.2.7	Дискретная математика	
2.2.8	Теория систем и системный анализ	
2.2.9	Компьютерная графика	
2.2.10	Проектирование информационных систем	
2.2.11	Средства информатизации в энергетике	
2.2.12	Средства информатизации в металлургии	
2.2.13	Проектирование систем SCADA	

3. РЕЗУЛЬТАТЫ ОБУЧЕНИЯ ПО ДИСЦИПЛИНЕ, СООТНЕСЕННЫЕ С ФОРМИРУЕМЫМИ КОМПЕТЕНЦИЯМИ

УК-1: Способен осуществлять поиск, критический анализ и синтез информации, умение анализировать процессы и системы с использованием соответствующих аналитических, вычислительных и экспериментальных методов, применять системный подход для решения поставленных задач	
Знать:	
УК-1-31 Основные этапы решения прикладных задач с использованием языка программирования	
ОПК-7: Способен выбирать и применять методики проектирования и актуальные инструментальные средства, проектировать и разрабатывать алгоритмы и программы, пригодные для практического применения	
Знать:	
ОПК-7-31 Основы алгоритмизации и программирования	
УК-1: Способен осуществлять поиск, критический анализ и синтез информации, умение анализировать процессы и системы с использованием соответствующих аналитических, вычислительных и экспериментальных методов, применять системный подход для решения поставленных задач	
Уметь:	
УК-1-У1 формализовать прикладную задачу, анализировать данные, осуществлять поиск алгоритма и технологию реализации	
ОПК-7: Способен выбирать и применять методики проектирования и актуальные инструментальные средства, проектировать и разрабатывать алгоритмы и программы, пригодные для практического применения	
Уметь:	
ОПК-7-У1 формализовать прикладную задачу, анализировать данные, осуществлять поиск алгоритма и технологию реализации	
УК-1: Способен осуществлять поиск, критический анализ и синтез информации, умение анализировать процессы и системы с использованием соответствующих аналитических, вычислительных и экспериментальных методов, применять системный подход для решения поставленных задач	
Владеть:	
УК-1-В1 основами современных технологий программирования, основными понятиями, методами и принципами разработки программ	
ОПК-7: Способен выбирать и применять методики проектирования и актуальные инструментальные средства, проектировать и разрабатывать алгоритмы и программы, пригодные для практического применения	

Владеть:

ОПК-7-В1 основами современных технологий программирования, основными понятиями, методами и принципами разработки программ

4. СТРУКТУРА И СОДЕРЖАНИЕ

Код занятия	Наименование разделов и тем /вид занятия/	Семестр / Курс	Часов	Формируемые индикаторы компетенций	Литература и эл. ресурсы	Примечание	КМ	Выполняемые работы
	Раздел 1. Введение в алгоритмизацию и обработку данных							
1.1	1. Понятие алгоритма. Свойства алгоритмов. Основные алгоритмические конструкции. Этапы разработки алгоритмов. Алгоритмическая реализация информационных задач. Примеры алгоритмов. /Лек/	1	4	УК-1-31 УК-1-У1 УК-1-В1 ОПК-7-31 ОПК-7-В1	Л1.4 Л1.5 Л1.6Л2.1Л3. 1 Э1 Э2 Э3 Э4 Э5		КМ1	
1.2	Линейные алгоритмы. Разветвляющиеся алгоритмы. Циклические алгоритмы /Лаб/	1	4	УК-1-31 УК-1-У1 УК-1-В1 ОПК-7-31 ОПК-7-В1	Л1.4 Л1.5 Л1.6Л2.1Л3. 1 Э1 Э2 Э3 Э4 Э5			Р2
	Раздел 2. Язык программирования Python							
2.1	Знакомство со средой программирования. Основные базовые конструкции. /Лек/	1	4	УК-1-31 УК-1-У1 УК-1-В1 ОПК-7-31 ОПК-7-В1	Л1.1 Л1.2 Л1.3 Л1.6 Л1.8Л2.1Л3. 1 Э1 Э2 Э3 Э4 Э5		КМ1	Р1
2.2	Типы данных. Коллекции Python. /Лек/	1	9	УК-1-31 УК-1-У1 УК-1-В1 ОПК-7-31 ОПК-7-В1	Л1.1 Л1.2Л2.1Л3. 1 Э1 Э2 Э3 Э4 Э5		КМ1	Р1
2.3	Составление программ линейной, разветвляющейся, циклической структуры /Лаб/	1	6	УК-1-31 УК-1-У1 УК-1-В1 ОПК-7-31 ОПК-7-В1	Л1.3 Л1.7 Л1.8Л2.1Л3. 1 Э1 Э2 Э3 Э4 Э5		КМ4	Р2
2.4	Типы данных. Множества. Строки. Индексация. Срезы. /Лаб/	1	8	УК-1-31 УК-1-У1 УК-1-В1 ОПК-7-31 ОПК-7-В1	Л1.7 Л1.8Л2.1Л3. 1 Э1 Э2 Э3 Э4 Э5		КМ4	Р2
2.5	Типы данных. Списки. Кортежи. Преобразование коллекций. Методы списков и строк. /Лаб/	1	8	УК-1-31 УК-1-У1 УК-1-В1 ОПК-7-31 ОПК-7-В1	Л1.2 Л1.5Л2.1Л3. 1 Э1 Э2 Э3 Э4 Э5		КМ4	Р2
2.6	Вложенные списки. Словари. Использование функций. /Лаб/	1	8	УК-1-31 УК-1-У1 УК-1-В1 ОПК-7-31 ОПК-7-В1	Л1.2 Л1.5Л2.1Л3. 1 Э1 Э2 Э3 Э4 Э5		КМ4	Р3
2.7	Изучение материалов в электронной библиотеке и в среде Canvas. /Ср/	1	20	УК-1-31 УК-1-У1 УК-1-В1 ОПК-7-31 ОПК-7-В1	Л1.1 Л1.3Л2.1Л3. 1 Э1 Э2 Э3 Э4 Э5		КМ3	Р3

2.8	Подготовка отчётов по лабораторным работам /Ср/	1	11	УК-1-31 УК-1-У1 УК-1-В1 ОПК-7-31 ОПК-7-В1	Л1.1 Л1.2 Л1.3 Л1.4 Л1.5 Л1.6 Л1.7 Л1.8Л2.1Л3. 1 Э1 Э2 Э3 Э4 Э5		КМ3	Р4
2.9	Подготовка к зачёту /Ср/	1	6	УК-1-31 УК-1-У1 УК-1-В1 ОПК-7-31 ОПК-7-В1	Л1.5Л2.1Л3. 1 Э1 Э2 Э3 Э4 Э5		КМ3	Р2,Р3,Р4
2.10	Введение в ООП. Принципы ООП. Классы. /Лек/	2	9	УК-1-31 УК-1-У1 УК-1-В1 ОПК-7-31 ОПК-7-В1	Л2.1Л3.1 Э1 Э2 Э3 Э4 Э5		КМ1	Р2,Р3
2.11	Классы. Проектирование и разработка классов. Понятие библиотеки языка программирования. Библиотеки Python. PyQT. Файлы, таблицы. /Лек/	2	8	УК-1-31 УК-1-У1 УК-1-В1 ОПК-7-31 ОПК-7-В1	Л2.1Л3.1 Э1 Э2 Э3 Э4 Э5		КМ1	Р2,Р3
2.12	PyQT. Определение операторов. /Лаб/	2	6	УК-1-31 УК-1-У1 УК-1-В1 ОПК-7-31 ОПК-7-В1	Л1.1 Л1.2 Л1.3 Л1.4 Л1.5 Л1.6 Л1.7 Л1.8Л2.1Л3. 1 Э1 Э2 Э3 Э4 Э5		КМ4	Р3,Р4
2.13	Проектирование классов /Лаб/	2	8	УК-1-31 УК-1-У1 УК-1-В1 ОПК-7-31 ОПК-7-В1	Л1.1 Л1.5Л2.1Л3. 1 Э1 Э2 Э3 Э4 Э5		КМ4	Р3,Р4
2.14	Файлы в Python. Типы файлов и работа с ними. /Лаб/	2	8	УК-1-31 УК-1-У1 УК-1-В1 ОПК-7-31 ОПК-7-В1	Л1.1 Л1.2 Л1.3 Л1.4 Л1.5 Л1.6 Л1.7 Л1.8Л2.1Л3. 1 Э1 Э2 Э3 Э4 Э5		КМ1,КМ4	Р3,Р4
2.15	Работа с табличными данными в PyQT. /Лаб/	2	12	УК-1-31 УК-1-У1 УК-1-В1 ОПК-7-31 ОПК-7-В1	Л1.1 Л1.2 Л1.3 Л1.4 Л1.5 Л1.6 Л1.7 Л1.8Л2.1Л3. 1 Э1 Э2 Э3 Э4 Э5		КМ4	Р3,Р4
2.16	Изучение материалов в электронной библиотеке, в информационно-справочных системах; аналитическая обработка текста (конспектирование, реферирование) /Ср/	2	30	УК-1-31 УК-1-У1 УК-1-В1 ОПК-7-31 ОПК-7-В1	Л1.1 Л1.2 Л1.3 Л1.4 Л1.5 Л1.6 Л1.7 Л1.8Л2.1Л3. 1 Э1 Э2 Э3 Э4 Э5		КМ3	Р1,Р2,Р3,Р4

2.17	Подготовка и сдача лабораторных работ. /Ср/	2	21	УК-1-31 УК-1-У1 УК-1-В1 ОПК-7-31 ОПК-7-В1	Л1.1 Л1.2 Л1.3 Л1.4 Л1.5 Л1.6 Л1.7 Л1.8Л2.1Л3.1 Э1 Э2 Э3 Э4 Э5		КМ3	Р1,Р2,Р3,Р4
2.18	Подготовка к экзамену /Ср/	2	37	УК-1-31 УК-1-У1 УК-1-В1 ОПК-7-31 ОПК-7-В1	Л1.1 Л1.2 Л1.3 Л1.4 Л1.5 Л1.6 Л1.7 Л1.8Л2.1Л3.1 Э1 Э2 Э3 Э4 Э5		КМ3	Р1,Р2,Р3,Р4
2.19	Выполнение контрольной работы /Ср/	1	6	УК-1-31 УК-1-У1 УК-1-В1 ОПК-7-31 ОПК-7-В1	Л1.1 Л1.4 Л1.5Л2.1Л3.1 Э1 Э2 Э3 Э4 Э5		КМ4	Р1,Р2,Р3,Р4
Раздел 3. Подготовка к контрольным мероприятиям и выполняемым работам								
3.1	Объем часов самостоятельной работы на подготовку к КМ /Ср/	2	6	УК-1-31 УК-1-У1 УК-1-В1 ОПК-7-31 ОПК-7-В1	Л1.1 Л1.2 Л1.3 Л1.4 Л1.5 Л1.6 Л1.7 Л1.8Л2.1Л3.1 Э1 Э2 Э3 Э4 Э5		КМ3	Р1,Р2,Р3,Р4
3.2	Объем часов самостоятельной работы на подготовку к ВР /Ср/	2	8	УК-1-31 УК-1-У1 УК-1-В1 ОПК-7-31 ОПК-7-В1	Л1.1 Л1.2 Л1.3 Л1.4 Л1.5 Л1.6 Л1.7 Л1.8Л2.1Л3.1 Э1 Э2 Э3 Э4 Э5		КМ3	Р1,Р2,Р3,Р4
Раздел 4. Подготовка к контрольным мероприятиям и выполняемым работам								
4.1	Объем часов самостоятельной работы на подготовку к КМ /Ср/	1	6	УК-1-31 УК-1-У1 УК-1-В1 ОПК-7-31 ОПК-7-В1	Л2.1Л3.1 Э1 Э2 Э3 Э4 Э5			
4.2	Объем часов самостоятельной работы на подготовку к ВР /Ср/	1	8	УК-1-31 УК-1-У1 УК-1-В1 ОПК-7-31 ОПК-7-В1	Л2.1Л3.1 Э1 Э2 Э3 Э4 Э5			

5. ФОНД ОЦЕНОЧНЫХ МАТЕРИАЛОВ

5.1. Контрольные мероприятия (контрольная работа, тест, коллоквиум, экзамен и т.п), вопросы для самостоятельной подготовки

Код КМ	Контрольное мероприятие	Проверяемые индикаторы компетенций	Вопросы для подготовки
--------	-------------------------	------------------------------------	------------------------

КМ1	Контрольные вопросы	УК-1-31;ОПК-7-31	<ol style="list-style-type: none">1. Какие основные этапы включает в себя решение задач на компьютере?2. Какие этапы компьютерного решения задач осуществляются без участия компьютера?3. Что называют математической моделью объекта или явления?4. Из каких последовательных действий состоит процесс разработки программы?5. Что называется алгоритмом?6. Перечислите основные свойства алгоритма7. Какими графическими символами принято изображать в схемах алгоритм?8. В чем отличие циклической структуры с предусловием от циклической структуры с постусловием?9. Приведите пример известных алгоритмов.
КМ2	Контрольные вопросы	УК-1-31;ОПК-7-31	<ol style="list-style-type: none">1. Чем тестирование программы отличается от её отладки?2. Можно ли с помощью тестирования доказать правильность программы?3. На какой стадии работы над программой вычисляются эталонные результаты тестов?4. Назовите основные этапы процесса тестирования.5. В чём заключается отличие синтаксических ошибок от семантических?6. О чём свидетельствует отсутствие сообщений машины о синтаксических ошибках?7. Какие разновидности ошибок транслятор не в состоянии обнаружить?

КМЗ	Экзамен	УК-1-В1;ОПК-7-31	<ol style="list-style-type: none"> 1. Понятие алгоритма. Свойства алгоритмов. Основные алгоритмические конструкции. 2. Этапы разработки алгоритмов. Алгоритмическая реализация информационных задач. Примеры алгоритмов. 3. Линейные алгоритмы. Разветвляющиеся алгоритмы. Циклические алгоритмы. Пример. 4. Встроенный тип str. Методы объекта str. 5. print() и форматирование вывода. 6. Работа с файловой системой средствами Python. 7. Работа с файлами. Методы open(), close(), read(), write(). 8. Встроенные типы последовательностей list, tuple, range и их методы. 9. Встроенный объект dict и его методы. 10. Встроенные типы чисел — int, float, complex. Машинное представление чисел с плавающей точкой и целых. 11. Множества. Встроенные типы set и frozenset. 12. Инструкции и синтаксис. Составные конструкции и обработка исключений 13. Инструкции if/else/elif, логические операторы и выражения сравнения 14. Циклы while и for в Python 15. Модульный подход к разработке программ. Функции. Назначение функций, виды функций, описание и определение функций. 16. Функции Python.. Основные понятия. Области видимости и пространство имен в Python. 17. Передача аргументов в функцию. Специальные режимы сопоставления аргументов. 18. Введение в объектно-ориентированное программирование. Принципы ООП. 19. Объекты. Динамическая типизация. Инкапсуляция. 20. Генерация объекта class. Новое пространство имен. Объект экземпляр класса. 21. Классы. Проектирование и разработка классов. Понятие библиотеки языка программирования. Библиотеки в Python. 22. Атрибуты класса. Атрибуты данных. Атрибуты-методы. Параметр self. Добавление атрибутов к классу во время исполнения программы. 23. Специальные методы и атрибуты классов. Методы __init__() и __del__() в Python. Декораторы функций и декораторы классов. 24. Абстрактные методы в Python. Классические классы и классы нового стиля. 25. Наследование. Базовый и производный класс. Построение производного класса. 26. Полиморфизм. Подмена методов в производном классе. Доступ к методам базового класса. 27. Обработка исключений. Инструкция try... except... else... finally. Объект Менеджер контекста и конструкция with...as. Классы встроенных исключений. Пользовательские исключения. Отладочные проверки assert и возбуждение исключения AssertionError.raise. Поиск ошибок программирования на стадиях разработки и тестирования. 28. Модули и пакеты. Библиотеки сторонних разработчиков (Django, Flask, Pygame, PyQt). 29. Модули и пакеты. Графический интерфейс. 30. Взаимодействие Python с базой данных SQLite.
-----	---------	------------------	---

КМ4	Контрольная работа	УК-1-31;ОПК-7-31	<p>Понятие алгоритма. Свойства алгоритмов. Основные алгоритмические конструкции.</p> <p>Этапы разработки алгоритмов. Алгоритмическая реализация информационных задач. Примеры алгоритмов.</p> <p>Язык программирования Python. Его место среди языков программирования. Сферы применения.</p> <p>Условный оператор. Примеры применения.</p> <p>Линейные алгоритмы. Разветвляющиеся алгоритмы. Циклические алгоритмы. Пример.</p> <p>Встроенный тип str. Методы объекта str. print() и форматирование вывода.</p> <p>Встроенные типы последовательностей list, tuple, range и их методы.</p> <p>Коллекции в Python.</p> <p>Словари. Особенности. Методы словарей.</p> <p>Множества. Особенности. Методы.</p> <p>Алгоритм сортировки в Python. Примеры использования.</p> <p>Списки. Особенности. Методы списков</p> <p>Списочные выражения. Вложенные списки. Массивы.</p> <p>Кортежи. Особенности. Примеры использования</p> <p>задачи по вариантам из хэнд бука</p> <ol style="list-style-type: none"> 1) 2.3 - G, S 2) 2.4 - G, T 3) 2.2 - J; 3.1 - I 4) 2.2 - K, R 5) 2.4 - C, N 6) 3.1 - F, I 7) 2.1 - L, O 8) 3.1 - G, R 9) 3.1 - E, K 10) 3.2 - C; 2.3 - I 11) 3.2 - B; 2.4 - K 12) 2.2 - N, 2.3 - D 13) 3.1 - C, L 14) 3.1 - G; 2.4 - T 15) 3.1 - H; 2.4 - H
-----	--------------------	------------------	--

КМ5	Диагностические задания	УК-1-31;ОПК-7-31	<p>1. Кроссплатформенное программное обеспечение – это программное обеспечение, работающее:</p> <ol style="list-style-type: none"> 1) на одной аппаратной платформе и/или операционной системе; 2) на двух аппаратных платформах и/или операционных системах; 3) более чем на одной аппаратной платформе и/или операционной системе; 4) более чем на двух аппаратных платформах и/или операционных системах. <p>2. Процесс создания приложения можно разделить на следующие этапы:</p> <ol style="list-style-type: none"> 5) написание программного кода, описание свойств элементов; 6) формирование окна программы; 7) отладка программы; 8) тестирование; 9) разработка справочной системы. <p>3. Укажите компонент, который служит для размещения текста на форме:</p> <ol style="list-style-type: none"> 1) Label; 2) Button; 3) Edit; 4) Image; 5) GroupBox. <p>4. Компонент, который представляет собой текстовое окно, в которое можно вводить текст или числа во время выполнения программы размещения текста на форме называется ...</p> <ol style="list-style-type: none"> 1) Label; 2) Button; 3) Edit; 4) Image; 5) OpenFileDialog. <p>5. Компонент, который используется для организации выбора из не-скольких взаимоисключающих возможностей (при этом допускается выбор только одного варианта из нескольких) называется ...</p> <ol style="list-style-type: none"> 1) GroupBox; 2) RadioButton; 3) ListBox; 4) ComboBox; <p>6. Компонент, который может выполнять некоторый код через регулярные промежутки времени. Размещенный на форме, он остается невидимым во время работы приложения называется ...</p> <ol style="list-style-type: none"> 1) GroupBox; 2) RadioButton; 3) ListBox; 4) ComboBox; 5) Timer. <p>7. Вещественное число – это ...</p> <ol style="list-style-type: none"> 1) тип данных; 2) идентификатор; 3) служебное слово; 4) свойство переменных. <p>8. Язык высокого уровня - это</p> <ol style="list-style-type: none"> 1) Реализация языка Ассемблера 2) Язык программирования, наиболее приближенный к человеческому языку 3) Машинный язык 4) Мнемоническое представление машинного языка <p>9. Компонент, представляющий сочетание выпадающего списка и однострочного текстового поля, которое позволяет пользователю ввести значение вручную или выбрать из списка:</p> <ol style="list-style-type: none"> 1) GroupBox; 2) RadioButton; 3) ListBox; 4) ComboBox;
-----	-------------------------	------------------	---

<p>5) Timer.</p> <p>10. ShowMessage() осуществляет;</p> <ol style="list-style-type: none"> 1) Вывод сообщения 2) Результат компилирования 3) Вызов окна сообщения ошибок <p>11. Результатом выполнения операции $5 // 2$ является число...</p> <p>Запишите число: ____</p> <p>12. Результатом выполнения операции $5 \% 2$ является число...</p> <p>Запишите число: ____</p> <p>13. В языке Python объектами являются:</p> <ol style="list-style-type: none"> 1) экземпляры классов и переменные 2) экземпляры классов, переменные и функции 3) экземпляры классов, классы и переменные 4) все типы данных <p>14. В результате выполнения программы будет выведено _____</p> <p>Дана программа:</p> <pre> from math import pi class Circle: def __init__(self, radius): self.radius = radius def perimeter(self): return 2 * pi * self.radius class Square: def __init__(self, side): self.side = side def area(self): return self.side * self.side def print_shape_info(shape): print("Perimeter = {}".format(shape.perimeter())) square = Square(10) circle = Circle(10) print_shape_info(circle) print_shape_info(square) </pre> <p>15. Верно ли утверждение: метод <code>__init__</code> выполняется при создании каждого нового экземпляра класса и инициализирует свойства нового экземпляра</p> <p>- Да - Нет</p> <p>16. Вставьте пропущенное слово _____ описывает модель объекта, его свойства и поведение.</p> <p>17. Вставьте пропущенное слово _____ - действие, которое объект может выполнять над самим собой или другими объектами.</p> <p>18. Даны объекты 1, 2, 3, "abc", [10, 20, 30]. Пропишите через запятую классы этих объектов _____</p> <p>19. Чтобы узнать, к какому классу относится тот или иной объект, можно воспользоваться функцией _____</p> <p>20. У методов всегда есть хотя бы один аргумент, и первый по счету аргумент должен называться _____</p> <p>21. Особое значение метода _____ заключается в том, что, если такой метод в классе определен, интерпретатор автоматически вызывает его при создании каждого экземпляра этого класса для инициализации экземпляра.</p> <p>22. Укажите результат выполнения программы:</p> <pre> class Foo: def foo(self): print ('foo') del self def __del__(self): print ('del') obj = Foo() obj.foo() </pre> <p>Варианты ответа:</p> <ol style="list-style-type: none"> 1) foo
--

2) del
3) del foo
4) foo del

23. Укажите результат выполнения программы:
x=0
class Foo:
count=x
def __init__(self):
self.count+=1
def __del__(self):
self.count+=1
obj = Foo()
print (obj.count)
Варианты ответа:
1) 0
2) 1
3) 2
4) пустая строка
5) скрипт не будет выполнен, т.к. код содержит ошибки

24 . Укажите результат выполнения скрипта:
class Foo:
count=0
def __init__(self):
self.count+=1
obj = Foo()
print (obj.count)
Варианты ответа:
1) 0
2) 1
3) пустая строка
4) скрипт не будет выполнен, т.к. код содержит ошибки

25. Укажите результат выполнения скрипта:
class Foo:
def foo(self):
print ('1')
def __init__(self):
print ('2')
x = Foo()
Варианты ответа:
1) 1
2) 2
3) 2;1
4) пустая строка
5) скрипт не будет выполнен, т.к. код содержит ошибки

26. Укажите результат выполнения программы:
class Foo:
def __init__(self):
print ('construct')
del self
obj = Foo()
if obj: print ('exist')
Варианты ответа:
1) construct
2) exist
3) construct exist
4) скрипт не будет выполнен, т.к. код содержит ошибки

27. Укажите результат выполнения программы:
class Foo:
def __init__(self):
print ('constructor', end=' ')
self.__del__(self)
def __del__(self):
print ('destructor', end=' ')
obj = Foo()
Варианты ответа:
1) constructor
2) destructor
3) constructor destructor

			4) скрипт не будет выполнен, т.к. код содержит ошибки
КМ6	Зачёт	УК-1-31	<ol style="list-style-type: none"> 1. Понятие алгоритма. Свойства алгоритмов. Основные алгоритмические конструкции. 2. Этапы разработки алгоритмов. Алгоритмическая реализация информационных задач. Примеры алгоритмов. 3. Язык программирования Python. Его место среди языков программирования. Сферы применения. 4. Условный оператор. Примеры применения. 5. Линейные алгоритмы. Разветвляющиеся алгоритмы. Циклические алгоритмы. Пример. 6. Встроенный тип str. Методы объекта str. 7. print() и форматирование вывода. 8. Встроенные типы последовательностей list, tuple, range и их методы. 9. Коллекции в Python. 10. Словари. Особенности. Методы словарей. 11. Множества. Особенности. Методы. 12. Алгоритм сортировки в Python. Примеры использования. 13. Списки. Особенности. Методы списков 14. Списочные выражения. Вложенные списки. Массивы. 15. Кортежи. Особенности. Примеры использования

5.2. Перечень работ, выполняемых по дисциплине (Курсовая работа, Курсовой проект, РГР, Реферат, ЛР, ПР и т.п.)

Код работы	Название работы	Проверяемые индикаторы компетенций	Содержание работы
P1	Лабораторная работа 1	УК-1-У1;УК-1-В1	<p>Цель: научиться работать с алгоритмическим представлением данных.</p> <p>1. Что получится в результате выполнения шагов следующих блок - схем.</p> <p>Предлагаются блок-схемы.</p> <p>Результат можно проверить с использованием Python/</p>
P2	Лабораторная работа 2	УК-1-У1;УК-1-В1	<p>Базовые конструкции Python</p> <ol style="list-style-type: none"> 1. Ввод и вывод данных. Операции с числами, строками. Форматирование 2. Условный оператор 3. Циклы 4. Вложенные циклы <p>20 задач из хэндбука</p>
P3	Лабораторная работа 3	УК-1-У1;ОПК-7-У1	<p>Коллекции и работа с памятью</p> <ol style="list-style-type: none"> 1. Строки, кортежи, списки 2. Множества, словари 3. Списочные выражения. Модель памяти для типов языка Python 4. Встроенные возможности по работе с коллекциями
P4	Лабораторная работа 4	УК-1-В1;ОПК-7-У1	<p>Функции и их особенности в Python</p> <ol style="list-style-type: none"> 1. Функции. Области видимости. Передача параметров в функции 2. Позиционные и именованные аргументы. Функции высших порядков. Лямбда-функции 3. Рекурсия. Декораторы. Генераторы

5.3. Оценочные материалы, используемые для экзамена (билеты, тесты и т.п.)

Формой промежуточной аттестации по дисциплине является в первом семестре - зачет, во втором - экзамен.
В первом семестре экзамен не предусмотрен

Во втором семестре проводится экзамен

Ниже представлен образец билета для экзамена, проводимого в устной форме.

МИНИСТЕРСТВО НАУКИ И ВЫСШЕГО ОБРАЗОВАНИЯ РОССИЙСКОЙ ФЕДЕРАЦИИ
Федеральное государственное автономное образовательное учреждение высшего образования
НАЦИОНАЛЬНЫЙ ИССЛЕДОВАТЕЛЬСКИЙ ТЕХНОЛОГИЧЕСКИЙ УНИВЕРСИТЕТ
«МИСиС»
НОВОТРОИЦКИЙ ФИЛИАЛ

Кафедра математических и естественно – научных дисциплин

БИЛЕТ К ЭКЗАМЕНУ № 0

Дисциплина: «Алгоритмизация и программирование»

Направление: 09.03.03 "Прикладная информатика_ПрПИВТС"

Форма обучения: очная

Форма проведения экзамена: устная

1. Понятие алгоритма. Свойства алгоритмов. Основные алгоритмические конструкции.
2. Циклы while и for в Python

Составил к.п.н, доцент: _____ А.В. Леднов

Зав. кафедрой МиЕ: _____ А.В. Швалёва

«01» сентября 2026 г.

Дистанционно экзамен проводится в LMS Canvas. Экзаменационный тест содержит 30 заданий. На решение отводится 30 минут. Разрешенные попытки - две. Зачитывается наилучший результат. Образец заданий для экзамена, проводимого дистанционно в LMS Canvas

1.К изменяемым относятся:

Списки,
Множества,
Словари.

2.Представителями неизменяемых типов являются:

Строки,
Кортежи,
Числа.

3.Выберите верные утверждения о языке Python

Выберите по крайней мере один ответ:

Является компилируемым языком
Является интерпретируемым языком
Является языком высокого уровня
Является объектно-ориентированным языком
Среди приведенных вариантов нет правильного

4.Что из нижеперечисленного является объектами?

Выберите по крайней мере один ответ:

Генератор
Итератор
Декоратор
Менеджер контекста
Среди приведенных вариантов нет правильного

5.Какие из указанных типов являются неизменяемыми (immutable)?

Выберите по крайней мере один ответ:

Кортеж (tuple)
Список (list)
Словарь (dict)
Множество (set)
Среди приведенных вариантов нет правильного

6. Что выведет следующий фрагмент программы:

```
numbers = [2, 5, 4, 3, 1]
hello = 'Hello, world!'
numbers.sort()
hello.upper()
print(numbers[2:])
print(hello[0])
```

7. Что выведет фрагмент программы:

```
name = 'Пауза'
print(f"Меня зовут {name}. Друзья зовут меня {name[:1] + 'o' + name[3:]}.")
print(f"А автоматический набор текста зовёт меня так: {'П' + name[1:]}.")
```

8. Может ли список (list) быть элементом множества (set)?

Выберите один ответ:

Да
Нет

9. Что выведет блок -схема?

10.В какой последовательности нужно записать цепочку вызовов методов в нижеприведенной программе, чтобы на экран была выведена строка «ТИШЕ ЕДЕШЬ – ДАЛЬШЕ БУДЕШЬ»: `progravn = ' тише едишь – дольше будишь '`

```
print(proverb.....)
```

- 1 `replace('O', 'A')`
- 2 `replace('ишь', 'ешь')`

3 lower()
4 upper()
5 strip()

5.4. Методика оценки освоения дисциплины (модуля, практики. НИР)

Критерии оценки ответов на зачёте, проводимом в дистанционной форме в электронном курсе

90 ≤ Процент верных ответов ≤ 100 - отлично

75 ≤ Процент верных ответов < 90 - хорошо

60 ≤ Процент верных ответов < 75 – удовлетворительно

Критерии оценки выполнения курсового проекта:

1. Теоретические сведения изложены в достаточном объеме, четко и последовательно
2. Используются выводы (позиции, мнения и др.) известных ученых, профессионалов
3. Исследуются и сравниваются разные подходы, методики, приводятся собственные суждения и выводы
4. Приведено описание предметной области в полном объеме
5. Разработана модель информационной системы
6. Разработана модель процесса системы.
7. Текст написан грамотно, стилистически выдержан
8. Текст оформлен в соответствии с требованиями

Проект оценивается на отлично, если:

теоретические сведения изложены в достаточном объеме, четко и последовательно, использованы выводы (позиции, мнения и др.) известных ученых, профессионалов, исследуются и сравниваются разные подходы, методики, приводятся собственные суждения и выводы, имеются примеры, даются ссылки на источники, текст написан грамотно, стилистически выдержан и оформлен в соответствии с требованиями.

задача поставлена верно, предметная область описана точно в полном объеме.

верно разработаны модель информационной системы, модель процесса системы, в полном объеме, приведены пояснения и рисунки.

В целом по работе: расставлены ссылки на источники, текст написан грамотно, стилистически выдержан, оформлен в соответствии с требованиями.

Выполнение проекта оценивается как хорошее, если она соответствует всем критериям, перечисленным выше, но в работе отсутствует описание и сравнения разных подходов, методик и т.д. с последующим формированием собственных выводов на данный счет. Модель информационной системы, модель процесса системы разработаны полностью, но имеются неточности, отсутствуют рисунки и пояснения.

В целом по работе: расставлены ссылки на источники, текст написан грамотно, стилистически выдержан, оформлен в соответствии с требованиями.

Выполнение работы оценивается как удовлетворительное, если она соответствует всем критериям, перечисленным выше, но в работе отсутствуют описание и сравнения разных подходов, методик и т.д. с последующим формированием собственных выводов на данный счет. Модель информационной системы, модель процесса системы разработаны не полностью, имеются неточности, отсутствуют рисунки и пояснения. Отсутствует описание предметной области.

Если работа допущена до защиты с оценкой «отлично», в процессе защиты студент хорошо владеет материалом, не использует при этом опорных конспектов и т.д., с легкостью отвечает на любой вопрос по курсовому проекту, то в этом случае студенту за выполнение курсового проекта ставится оценка «отлично», которая и проставляется в зачетную книжку и в ведомость.

В процессе защиты оценка повышаться не может, т.е. если студент допущен до защиты с оценкой «хорошо», «отлично» он уже в любом случае не сможет получить, а вот «удовлетворительно» может – если при защите возникают определенные трудности с ориентацией в материале, ответами на вопросы по курсовому проекту.

Если студент совершенно не владеет материалом курсового проекта, то получает «неудовлетворительно».

Если курсовой проект не соответствует критериям выполнения курсового проекта, то оценивается неудовлетворительно и до защиты не допускается.

Критерии оценки защиты лабораторных работ:

При оценке результатов защиты отчетов по лабораторным работам используется бинарная система, которая предусматривает следующие результаты и критерии оценивания:

- "Зачтено" Выполнены все задания лабораторной работы, студент ответил на все контрольные вопросы;
- "Не зачтено" Студент не выполнил или выполнил неправильно задания лабораторной работы, студент ответил на контрольные вопросы с ошибками или не ответил на контрольные вопросы.

6. УЧЕБНО-МЕТОДИЧЕСКОЕ И ИНФОРМАЦИОННОЕ ОБЕСПЕЧЕНИЕ

6.1. Рекомендуемая литература

6.1.1. Основная литература

	Авторы, составители	Заглавие	Библиотека	Издательство, год
Л1.1	Под ред. С.В.Симоновича	Информатика. Базовый курс: Учебник		СПб.: Питер, 2015
Л1.2	Острейковский В.А.	Информатика: Учебник		М.: Высш. шк., 2000
Л1.3	С.В. Симонович.	Информатика. Базовый курс: учебник		СПб.: Питер, 2015

	Авторы, составители	Заглавие	Библиотека	Издательство, год
Л1.4	Гниденко И.Г.	Технологии и методы программирования: учеб.пособие		М.: Юрайт, 2019
Л1.5	Е.Д. Агафонов, Г.В. Ващенко	Прикладное программирование: учебное пособие		, 2015
Л1.6	Гагарина Л.Г. и др.	Технология разработки программного обеспечения: учебное пособие		НИЦ ИНФРА-М, 2020
Л1.7	Симонович	Информатика		,
Л1.8	И. Ю. Ефимова, Р. И. Коробков.	Информатика: учебное пособие для вузов		5-е изд., стер. – Москва : ФЛИНТА,, 2021

6.1.2. Дополнительная литература

	Авторы, составители	Заглавие	Библиотека	Издательство, год
Л2.1	Ю.А.Аляев, О.А.Козлов	Алгоритмизация и языки программирования Pascal, C++, Visual Basic : Учебно-справочное пособие		М. : Финансы и статистика, 2002

6.1.3. Методические разработки

	Авторы, составители	Заглавие	Библиотека	Издательство, год
Л3.1	Е.Н. Гусева, И.Ю. Ефимова, Р.И. Коробков и др..	Информатика : учебное пособие		Москва : Издательство «Флинта», 2016.

6.2. Перечень ресурсов информационно-телекоммуникационной сети «Интернет»

Э1	Хендбуки Академии Яндекса. Бесплатные онлайн-учебники для тех, кто хочет освоить ключевые IT-дисциплины.	https://academy.yandex.ru/handbook
Э2	Научная электронная библиотека eLIBRARY	https://www.elibrary.ru
Э3	LMS Moodle	https://lms.misis.ru
Э4	НФ НИТУ МИСИС	http://nf.misis.ru
Э5	Университетская библиотека ONLINE	https://biblioclub.ru/

6.3 Перечень программного обеспечения

П.1	Браузер Google Chrome
П.2	Microsoft Teams
П.3	Python
П.4	Антивирус Dr Web Suite
П.5	PyCharm

6.4. Перечень информационных справочных систем и профессиональных баз данных

7. МАТЕРИАЛЬНО-ТЕХНИЧЕСКОЕ ОБЕСПЕЧЕНИЕ

Ауд.	Назначение	Вид	Оснащение
123	Учебная лаборатория (компьютерный класс) Кабинет курсового и дипломного проектирования, самостоятельной работы обучающихся	Лаб	14 шт. - Системный блок; 14 шт. - Монитор LCD LG21,5; 1 шт. - Экран настенный 200x200; 1 шт. - Проектор ACER X118DLP 3600; 1 шт. - Подвес для проектора; 1 шт. - Коммутатор D-Link; 1 шт. - Доска ученическая; 27 шт. - Столы ученические; 52 шт. - Стулья; 4 шт. - Жалюзи.

121	Учебная аудитория для занятий лекционного типа, практических занятий	Лек	14 шт. - Системный блок Intel Core; 14 шт. - Монитор LCD; 1 шт. - Экран настенный Seven Media 240x240; 1 шт. - Проектор ACER P5206; 1 шт. - Подвес для проектора; 1 шт. - Веб камера Logitech; 1 шт. - Доска ученическая; 27 шт. - Столы ученические; 52 шт. - Стулья; 4 шт. - Жалюзи.
113	Учебная лаборатория (компьютерный класс)	Лаб	13 шт. - Компьютер в сборе; 1 шт. - Проектор универсальный Vivitek DH278; 1 шт. - Экран настенный 150x200; 1 шт. - Коммутатор D-Link 16 порт.; 1 шт. - Подвес для проектора; 1 шт. - Веб камера Logitech; 1 шт. - Кондиционер ; 13 шт. - Стол компьютерный; 2 шт. - Стол преподавательский; 7 шт. - Стулья; 12 шт. - Кресло; 1 шт. - Шкаф книжный; 12 шт. - Рулонные шторы; 1 шт. - Ученическая доска;

8. МЕТОДИЧЕСКИЕ УКАЗАНИЯ ДЛЯ ОБУЧАЮЩИХСЯ

Алгоритмизация и программирование – одна из важных дисциплин, без применения теоретических основ и практических умений которой не обходиться ни одно из ИТ компаний и информационных отделов предприятий. Сложность ее освоения во многом определяется значительным объемом материала, большим числом специфических терминов и понятий, взаимосвязанностью с изученными школьными предметами как информатика и информационно-коммуникационные технологии, математика, физика и другими.

Для успешного усвоения теоретического материала необходимо регулярно посещать лекции, активно работать на лабораторных работах, дополнять и расширять лекционный материал используя источники интернет и электронные библиотечные ресурсы, значительное внимание уделять самостоятельному изучению дисциплины.

Успешному освоению курса также поможет ведение терминологического словаря, что позволит быстрее усваивать теоретический материал, грамотно строить свою речь при устных и письменных ответах.

Программа дисциплины включает лекционные и лабораторные занятия, выполнение 2-х компьютерных диагностических срезов.

Домашняя работа отличается значительными затратами времени и требует от студента знаний лекционного материала, и практических умений полученных на лабораторных занятиях. В связи с этим, при планировании своей самостоятельной работы следует учитывать, что пропуск лекционных и лабораторных занятий и невнимательное отношение к изучению методических указаний существенно осложнит выполнение домашней работы.

Подготовка к выполнению домашней работы заключается в изучении соответствующих методических указаний и стандартов по оформлению работ. Оформленная в соответствии со стандартами домашняя работа сдается на кафедру.

Работа считается выполненной, если она зачтена преподавателем, ведущим занятия.

Задание на выполнение домашней работы выдается на кафедре, срок сдачи на проверку – за 2 недели до экзаменационной сессии. Контролируемые разделы дисциплины 2-4 указаны в содержании. Консультации по вопросам, связанным с выполнением домашней работы, проводятся по согласованию с преподавателем, ведущим дисциплину, в соответствии с расписанием вывешиваемым на стенде кафедры и на сайте НФ НИТУ МИСиС.

Большое значение для успешного освоения дисциплины имеют лабораторные занятия. В соответствии с учебным планом по курсу предусмотрено 34 часов практических занятий. Задания к ним разработаны в соответствии со стандартом.

Лабораторные работы проходят в компьютерных классах и предполагают использование специального программного обеспечения, методических пособий и разработок с подробным изложением материала к занятию размещенные на сайте и в библиотеке. Выполнение практических заданий лабораторных работ связана со значительными затратами аудиторного времени, для их полноценного выполнения требуется предварительная проработка теоретического материала, что следует делать дома до начала лабораторной работы. Также следует заранее подготовить письменные ответы на контрольные вопросы по каждой лабораторной работе и подготовиться к собеседованию по ним. В связи с этим, при планировании своей учебной работы вам следует учитывать, что пропуск лабораторного занятия связан со сложностями их выполнения, своевременной отчетности по заданиям и собеседованию по контрольным вопросам.

Структура лабораторного занятия предполагает осуществление следующих видов работ:

- решение задач на компьютере;
- разбор заданий;
- отчет и контроль знаний.

Проведение лабораторных работ требует активного участия, высокого уровня организованности и самостоятельности студентов группы.

Подготовка к выполнению лабораторной работы заключается в составлении теоретического отчета по контрольным вопросам к лабораторной работе. После выполнения лабораторной работы оформляется отчет по результатам практической работы. Работа считается полностью зачтенной после ее защиты.

По индивидуальным самостоятельным заданиям, выдаваемым в начале занятия, необходимо провести самостоятельную разработку в компьютерной программе и сделать выводы по полученным результатам: протестировать программу и проанализировать полученные данные на их соответствие реальным величинам.

При работе над заданиями необходимо проследить взаимосвязь с вашим профилем подготовки, необходимо постоянно расширять свой кругозор, в чем большую помощь может оказать анализ источников в сети интернет и научно-методическая литература.

Итогом изучения дисциплины является в 1 семестре зачет, а во 2 - экзамен. Для подготовки к итоговой аттестации следует ознакомиться со списком вопросов. В качестве основы для подготовки рекомендуется лекционный материал и проработанные задания лабораторных работ. Для подготовки к вопросам, которые не нашли своего отражения в лекциях надлежит воспользоваться рекомендуемой литературой.

Оценка на экзамене выставляется исходя из оценок, полученных в результате диагностики в компьютерной программе (или устного ответа), оценки за домашнее задание и оценок полученных при защите лабораторных работ.

При освоении курса следует внимательно изучать основную и дополнительную литературу, анализировать рекомендованный материал. Список рекомендуемых источников содержится в рабочей программе курса.

Ознакомиться с программой курса и получить задания к практическим занятиям можно на кафедре МиЕ или на сайте <http://www.nfmisis.net/>.

Освоение дисциплины предполагает как проведение традиционных аудиторных занятий, так и работу в электронной информационно-образовательной среде НИТУ «МИСиС» (ЭИОС), частью которой непосредственно предназначенной для осуществления образовательного процесса является Электронный образовательный ресурс LMS Moodle. Он доступен по URL адресу <https://lms.misis.ru/> и позволяет использовать специальный контент и элементы электронного обучения и дистанционных образовательных технологий.

LMS Moodle используется преимущественно для асинхронного взаимодействия между участниками образовательного процесса посредством сети «Интернет».

Чтобы эффективно использовать возможности LMS Moodle и освоить дисциплину, нужно:

- 1) зарегистрироваться на курс. Для этого нужно перейти по ссылке ... Логин и пароль совпадает с логином и паролем от личного кабинета НИТУ МИСиС;
- 2) в опции «В начало» ознакомиться с содержанием курса, вопросами для самостоятельной подготовки, условиями допуска к аттестации, формой промежуточной аттестации (зачет/экзамен), критериями оценивания и др.;
- 3) в опции «Модули», заходя в соответствующие разделы изучать учебные материалы, размещенные преподавателем. В т.ч. пользоваться литературой, рекомендованной преподавателем, переходя по ссылкам;
- 4) в опции «Библиотека» возможно подбирать для выполнения письменных работ (контрольные, домашние работы, курсовые работы/проекты) литературу, размещенную в ЭБС НИТУ «МИСИС»;
- 5) в опции «Задания» нужно ознакомиться с содержанием задания к письменной работе, сроками сдачи, критериями оценки. В установленные сроки выполнить работу(ы), подгрузить их для проверки. Удобно называть файл работы следующим образом (название предмета (сокращенно), группа, ФИО, дата актуализации (при повторном размещении)). Например, Информатика_Иванов_И.И._БМТ-19_20.04.2020. Если работа содержит рисунки, формулы, то с целью сохранения форматирования ее нужно подгружать в pdf формате.

Работа, должна быть загружена за 10 дней до начала сессии для проверки, и:

- содержать все структурные элементы: титульный лист, введение, основную часть, заключение, список источников, приложения (при необходимости);
- быть оформлена в соответствии с требованиями.

Преподаватель в течение установленного срока (не более десяти дней) проверяет работу и размещает в комментариях к заданию рецензию. В ней он указывает как положительные стороны работы, так замечания. При наличии в рецензии замечаний и рекомендаций, нужно внести поправки в работу, подгрузить ее заново для повторной проверки. При этом важно следить за сроками, в течение которых должно быть выполнено задание. При нарушении сроков, указанных преподавателем возможность подгрузить работу остается, но система выводит сообщение о нарушении сроков. По окончании семестра подгрузить работу не получится;

- 6) в опции «Тесты» пройти тестовые задания, освоив соответствующий материал, размещенный в рубрике «Модули»;
- 7) в опции «Оценки» отслеживать свою успеваемость;
- 8) в опции «Объявления» читать объявления, размещаемые преподавателем, давать обратную связь;
- 9) в опции «Обсуждения» создавать обсуждения и участвовать в них (обсуждаются общие моменты, вызывающие вопросы у большинства группы). Данная рубрика также может быть использована для взаимной проверки;
- 10) проявлять регулярную активность на курсе.

Преимущественно для синхронного взаимодействия между участниками образовательного процесса посредством сети «Интернет» используется Microsoft Teams (MS Teams). Чтобы полноценно использовать его возможности нужно установить приложение MS Teams на персональный компьютер и телефон. Старостам нужно создать группу в MS Teams. Участие в группе позволяет:

- слушать лекции;
- работать на практических занятиях;
- быть на связи с преподавателем, задавая ему вопросы или отвечая на его вопросы в общем чате группы в рабочее время с 9.00 до 17.00;
- осуществлять совместную работу над документами (вкладка «Файлы»).

При проведении занятий в дистанционном синхронном формате нужно всегда работать с включенной камерой.

Исключение – если преподаватель попросит отключить камеры и микрофоны в связи с большими помехами. На аватарках должны быть исключительно деловые фото.

При проведении лекционно-практических занятий ведется запись. Это дает возможность просмотра занятия в случае невозможности присутствия на нем или при необходимости вновь обратиться к материалу и заново его просмотреть.