

Документ подписан простой электронной подписью.
 Информация о владельце:
 ФИО: Котова Лариса Анатольевна
 Должность: Директор филиала
 Дата подписания: 01.06.2026 19:25:47
 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	
	уп	рп	уп	рп		
Неделя	19		18			
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		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		8		
Часы на контроль			27	27	27	27
Итого	108	108	180	180	288	288

Программу составил(и):

к.т.н., Доцент, Леднов Анатолий Викторович

Рабочая программа дисциплины

Алгоритмизация и программирование

Составлен на основании учебного плана:

09.03.03_24_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	Программная инженерия	
2.2.9	Теория систем и системный анализ	
2.2.10	Компьютерная графика	
2.2.11	Проектирование информационных систем	
2.2.12	Проектирование систем SCADA	
2.2.13	Моделирование металлургических процессов с использованием современных программных продуктов	
2.2.14	Электротехника, электроника и схемотехника	

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. 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-У1 ОПК-7-В1	Л1.3Л2.1Л3. 1 Э1 Э2 Э3 Э4			
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-У1 ОПК-7-В1	Л1.2Л2.1Л3. 1 Э1 Э2 Э3 Э4		КМ1	Р1
	Раздел 2. Раздел 2. Язык программирования Python							
2.1	Знакомство со средой программирования. Основные базовые конструкции. Типы данных. Коллекции Python. /Лек/	1	13	УК-1-31 УК-1-У1 УК-1-В1 ОПК-7-31 ОПК-7-У1 ОПК-7-В1	Л1.1Л2.1Л3. 1 Э1 Э2 Э3 Э4			
2.2	Составление программ линейной, разветвляющейся, циклической структуры /Лаб/	1	6	УК-1-31 УК-1-У1 УК-1-В1 ОПК-7-31 ОПК-7-У1 ОПК-7-В1	Л1.1 Л1.2 Л1.3Л2.1Л3. 1 Э1 Э2 Э3 Э4		КМ2	Р2
2.3	Типы данных. Множества. Строки. Индексация. Срезы. Списки. Кортежи. Преобразование коллекций. Методы списков и строк. /Лаб/	1	10	УК-1-31 УК-1-У1 УК-1-В1 ОПК-7-31 ОПК-7-У1 ОПК-7-В1	Л1.1 Л1.2 Л1.3Л2.1Л3. 1 Э1 Э2 Э3 Э4		КМ2	Р3
2.4	Вложенные списки. Словари. Использование функций. /Лаб/	1	8	УК-1-31 УК-1-У1 УК-1-В1 ОПК-7-31 ОПК-7-У1 ОПК-7-В1	Л1.1 Л1.2 Л1.3Л2.1Л3. 1 Э1 Э2 Э3 Э4			Р4
2.5	Изучение материалов в электронной библиотеке и в среде Canvas /Ср/	1	10	УК-1-31 УК-1-У1 УК-1-В1 ОПК-7-31 ОПК-7-У1 ОПК-7-В1	Л1.1 Л1.2 Л1.3Л2.1Л3. 1 Э1 Э2 Э3 Э4			
2.6	Подготовка отчётов по лабораторным работам /Ср/	1	17	УК-1-31 УК-1-У1 УК-1-В1 ОПК-7-31 ОПК-7-У1 ОПК-7-В1	Л1.1 Л1.2 Л1.3Л2.1Л3. 1 Э1 Э2 Э3 Э4			Р1,Р2,Р3,Р4

2.7	Подготовка к зачёту /Ср/	2	18	УК-1-31 УК-1-У1 УК-1-В1 ОПК-7-31 ОПК-7-У1 ОПК-7-В1	Л1.1 Л1.2 Л1.3Л2.1Л3. 1 Э1 Э2 Э3 Э4			
2.8	Введение в ООП. Принципы ООП. Классы. Проектирование и разработка классов. Понятие библиотеки языка программирования. Библиотеки Python. PyQT. Файлы, таблицы /Лек/	2	17	УК-1-31 УК-1-У1 УК-1-В1 ОПК-7-31 ОПК-7-У1 ОПК-7-В1	Л1.1 Л1.2 Л1.3Л2.1Л3. 1 Э1 Э2 Э3 Э4		КМ5	
2.9	Объектно-ориентированное программирование в среде PyQt5 /Лаб/	2	18	УК-1-31 УК-1-У1 УК-1-В1 ОПК-7-31 ОПК-7-У1 ОПК-7-В1	Л1.1 Л1.2 Л1.3Л2.1Л3. 1 Э1 Э2 Э3 Э4		КМ5	Р5
2.10	Изучение материалов в электронной библиотеке, в информационно справочных системах; аналитическая обработка текста (конспектирование, реферирование) /Ср/	2	28	УК-1-31 УК-1-У1 УК-1-В1 ОПК-7-31 ОПК-7-У1 ОПК-7-В1	Л1.1 Л1.2 Л1.3Л2.1Л3. 1 Э1 Э2 Э3 Э4		КМ5	
2.11	Подготовка и сдача лабораторных работ. /Ср/	2	20	УК-1-31 УК-1-У1 УК-1-В1 ОПК-7-31 ОПК-7-У1 ОПК-7-В1	Л1.1 Л1.2 Л1.3Л2.1Л3. 1 Э1 Э2 Э3 Э4		КМ5	Р5
2.12	Подготовка к экзамену /Ср/	2	12	УК-1-31 УК-1-У1 УК-1-В1 ОПК-7-31 ОПК-7-У1 ОПК-7-В1	Л1.1 Л1.2 Л1.3Л2.1Л3. 1 Э1 Э2 Э3 Э4		КМ3,К М5	
2.13	Выполнение домашней работы /Лаб/	1	6	УК-1-31 УК-1-У1 УК-1-В1 ОПК-7-31 ОПК-7-У1 ОПК-7-В1	Л1.1 Л1.2 Л1.3Л2.1Л3. 1 Э1 Э2 Э3 Э4		КМ4	
2.14	Объектно ориентированное программирование. Библиотеки для получения и обработки данных /Лаб/	2	16	УК-1-31 УК-1-У1 УК-1-В1 ОПК-7-31 ОПК-7-У1 ОПК-7-В1	Л1.1 Л1.2 Л1.3Л2.1Л3. 1 Э1 Э2 Э3 Э4		КМ5	Р6
	Раздел 3. Раздел 3. Подготовка к контрольным мероприятиям и выполняемым работам							
3.1	Объем часов самостоятельной работы на подготовку к КМ /Ср/	2	8	УК-1-31 УК-1-У1 УК-1-В1 ОПК-7-31 ОПК-7-У1 ОПК-7-В1	Л1.1 Л1.2 Л1.3Л3.1 Э1 Э2 Э3 Э4			
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-У1 ОПК-7-В1	Л1.1 Л1.2 Л1.3Л2.1Л3. 1 Э1 Э2 Э3 Э4			
	Раздел 4. Подготовка к контрольным мероприятиям и выполняемым работам							

4.1	Объем часов самостоятельной работы на подготовку к КМ /Ср/	2	4	ОПК-7-31 ОПК-7-У1 ОПК-7-В1	Э1 Э2 Э3 Э4			
4.2	Объем часов самостоятельной работы на подготовку к ВР /Ср/	2	4	ОПК-7-31 ОПК-7-У1 ОПК-7-В1	Э1 Э2 Э3 Э4			
Раздел 5. Раздел 4. Подготовка к контрольным мероприятиям и выполняемым работам								
5.1	Объем часов самостоятельной работы на подготовку к КМ /Ср/	1	8	ОПК-7-31 ОПК-7-У1 ОПК-7-В1	Л1.1 Л1.2 Л1.3Л2.1Л3. 1 Э1 Э2 Э3 Э4			
5.2	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-У1 ОПК-7-В1	Л1.1 Л1.2 Л1.3Л2.1Л3. 1 Э1 Э2 Э3 Э4			
Раздел 6. Подготовка к контрольным мероприятиям и выполняемым работам								
6.1	Объем часов самостоятельной работы на подготовку к КМ /Ср/	1	6	ОПК-7-31 ОПК-7-У1 ОПК-7-В1	Э1 Э2 Э3 Э4			
6.2	Объем часов самостоятельной работы на подготовку к ВР /Ср/	1	8	ОПК-7-31 ОПК-7-У1 ОПК-7-В1	Э1 Э2 Э3 Э4			

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	<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 алгоритма 7. 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	<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-1-31	<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 <code>__init__()</code> и <code>__del__()</code> 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	Домашняя работа	УК-1-31	<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	<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	<p>1. Понятие алгоритма. Свойства алгоритмов. Основные алгоритмические конструкции.</p> <p>2. Этапы разработки алгоритмов. Алгоритмическая реализация информационных задач. Примеры алгоритмов.</p> <p>3. Язык программирования Python. Его место среди языков программирования. Сферы применения.</p> <p>4. Условный оператор. Примеры применения.</p> <p>5. Линейные алгоритмы. Разветвляющиеся алгоритмы. Циклические алгоритмы. Пример.</p> <p>6. Встроенный тип str. Методы объекта str.</p> <p>7. print() и форматирование вывода.</p> <p>8. Встроенные типы последовательностей list, tuple, range и их методы.</p> <p>9. Коллекции в Python.</p> <p>10. Словари. Особенности. Методы словарей.</p> <p>11. Множества. Особенности. Методы.</p> <p>12. Алгоритм сортировки в Python. Примеры использования.</p> <p>13. Списки. Особенности. Методы списков</p> <p>14. Списочные выражения. Вложенные списки. Массивы.</p> <p>15. Кортежи. Особенности. Примеры использования</p>

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>Составление программ линейной, разветвляющейся, циклической структуры</p> <p>Базовые конструкции Python</p> <p>1. Ввод и вывод данных. Операции с числами, строками.</p> <p>Форматирование</p> <p>2. Условный оператор</p> <p>3. Циклы</p> <p>4. Вложенные циклы</p> <p>20 задач из хэндбука</p>
P3	Лабораторная работа 3	УК-1-У1;УК-1-В1	<p>Типы данных. Множества. Строки. Индексация. Срезы.</p> <p>Списки. Кортежи. Преобразование коллекций. Методы списков и строк.</p> <p>https://academy.yandex.ru/handbook/python/article/stroki-kortezhispiski</p>
P4	Лабораторная работа 4	УК-1-У1;УК-1-В1	<p>Вложенные списки. Словари. Использование функций.</p> <p>https://academy.yandex.ru/handbook/python/article/funkcii-oblastivimidosti-peredacha-parametrov-v-funkcii</p>
P5	Лабораторная работа 5	УК-1-У1;УК-1-В1	Объектно-ориентированное программирование
P6	Лабораторная работа ООП	УК-1-У1;УК-1-В1	https://education.yandex.ru/handbook/python/article/obuektnaya-model-python-klassy-polya-i-metody

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

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: _____ Р.Р. Абдулвелеева

Зав. кафедрой МиЕ: _____ А.В. Швалёва

«31» августа 2022 г.

Дистанционно экзамен проводится в LMS . Экзаменационный тест содержит 30 заданий. На решение отводится 30 минут.

Разрешенные попытки - две. Зачитывается наилучший результат.

Образец заданий для экзамена, проводимого дистанционно в LMS

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»: `prograw = ' 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)

1. Контрольные вопросы КМ1, КМ2

2. Лабораторная работа 1

3. Лабораторная работа 2

4. Лабораторная работа 3

5. Лабораторная работа 4

6. Вопросы к экзамену КМ-3

7. Задания контрольной работы КМ-4

Критерии оценки ответов на экзамене, проводимом в устной форме

- оценка «отлично» выставляется студенту, если четко сформулирован ответ на вопрос билета, ясно излагаются основные понятия и теоретические основы; логически соединены в единое повествование термины, понятия, теоретические обобщения, относящиеся к раскрываемой теме; если без ошибок выполнено практическое задание;
- оценка «хорошо» выставляется студенту, если частично сформулирован ответ на вопрос билета, излагаются основные понятия и теоретические основы; недостаточно логично соединены в единое повествование термины, понятия, теоретические обобщения, относящиеся к раскрываемой теме; если без ошибок выполнено практическое задание;
- оценка «удовлетворительно» выставляется студенту, если отсутствует четко сформулированный ответ на поставленный вопрос и ясное изложение темы; отсутствует логическое соединение в единое повествование теоретические обобщения; ответ формулируется на примерах бытового уровня; практическое задание выполнено с недочетами.

Критерии оценки ответов на экзамене, проводимом в дистанционной форме в LMS

90 ≤ Процент верных ответов ≤ 100 - отлично

75 ≤ Процент верных ответов < 90 - хорошо

60 ≤ Процент верных ответов < 75 – удовлетворительно

Критерии оценки выполнения домашней работы:

1. Теоретические сведения изложены в достаточном объеме, четко и последовательно
2. Текст в электронном или в печатном виде оформлен строго по требованиям.
3. Используются собственные примеры
4. Имеются скриншоты и листинги примеров
5. Проведено описание процесса работы используемых функций, формул, операторов и обоснование их применения
6. Используются тестовые данные и приведены результаты работы программы (файла)
7. Высокое качество оформления работы с использованием правил оформления текста в текстовом редакторе
8. Используются и указаны источники литературы
9. Текст написан грамотно, стилистически выдержан

Работа оценивается по следующим отметкам:

Отметка «отлично» выставляется студенту, если:

- даны исчерпывающие и обоснованные ответы на все поставленные вопросы, правильно и рационально выполнены практические задания;
- студент самостоятельно и правильно решил практические задачи, уверенно, логично, последовательно и аргументировано излагал свое решение, используя соответствующую терминологию;
- в ответах выделялось главное, все теоретические положения умело увязывались с требованиями условия задания;
- письменные ответы были четкими и краткими, а мысли излагались в логической последовательности;
- показано умение самостоятельно анализировать факты, события, явления, процессы в их взаимосвязи и диалектическом развитии.

2. Отметка «хорошо» выставляется студенту, если:

- даны полные, достаточно обоснованные ответы на поставленные вопросы, правильно выполнены практические задания;
- студент самостоятельно и в основном правильно решил практические задачи, уверенно, логично, последовательно и аргументировано излагал решение, используя соответствующую терминологию;
- в ответах не всегда выделялось главное, отдельные положения недостаточно увязывались с требованиями условия задания, при решении практических задач не всегда использовались рациональные методы решения;
- ответы в основном были краткими, но не всегда четкими.

3. Оценка «удовлетворительно» выставляется студенту, если:

- даны в основном правильные ответы на все задания, но без должной глубины и обоснования, при выполнении практических заданий студент использовал прежний опыт и не применял новые методики выполнения заданий;
- студент в основном решил практические задачи, допустил несущественные ошибки, слабо аргументировал решение, почти не использовал соответствующую терминологию;
- при ответах не выделялось главное;
- письменные ответы были многословными, нечеткими и без должной логической последовательности.

4. Оценка «неудовлетворительно» выставляется студенту, если:

- студент не усвоил значительную часть учебного материала, письменный ответ не обоснован, скопирован, нет анализа решения задачи или не выполнил практические задания;
- студент не решил практическую задачу;
- испытывает трудности в практическом применении знаний;
- не может аргументировать научные положения;
- не формулирует выводов и обобщений.

Критерии оценки на зачете

Оценка «зачтено» выставляется студенту, если:

- 1 выполнены все лабораторные работы, которые содержат функционирующие приложения, исполняющее все условия предложенного задания; по лабораторным работам составлены отчеты;
- 2 сданы ответы на контрольные вопросы по каждой лабораторной работе в письменном и устном виде;
- 3 выполнена и своевременно сдана домашняя работа на отметку "отлично", "хорошо", "удовлетворительно";
- 4 устный ответ на зачете содержит исчерпывающие и обоснованные ответы на все поставленные вопросы, используется соответствующая терминология; в ответах выделялось главное; ответы были четкими и краткими, а мысли излагались в логической последовательности; показано умение самостоятельно анализировать факты, события, явления, процессы в их взаимосвязи и диалектическом развитии.

Оценка «не зачтено» выставляется студенту, если:

1 не выполнены предыдущие критерии 1-3;
2 студент не усвоил значительную часть учебного материала и допускает существенные ошибки и неточности при рассмотрении теоретических вопросов; испытывает трудности в практическом применении знаний; не может аргументировать научные положения; не формулирует выводов и обобщений

6. УЧЕБНО-МЕТОДИЧЕСКОЕ И ИНФОРМАЦИОННОЕ ОБЕСПЕЧЕНИЕ

6.1. Рекомендуемая литература

6.1.1. Основная литература

	Авторы, составители	Заглавие	Библиотека	Издательство, год
Л1.1	С.В. Симонович.	Информатика. Базовый курс: учебник		СПб.: Питер, 2015
Л1.2	Е.Д. Агафонов, Г.В. Ващенко	Прикладное программирование: учебное пособие		, 2015
Л1.3	Гагарина Л.Г. и др.	Технология разработки программного обеспечения: учебное пособие		НИЦ ИНФРА-М, 2020

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	НФ НИТУ МИСИС	http://nf.misis.ru
Э2	Университетская библиотека ONLINE	https://biblioclub.ru/
Э3	LMS Moodle	https://lms.misis.ru
Э4	Научная электронная библиотека eLIBRARY	https://www.elibrary.ru

6.3 Перечень программного обеспечения

6.4. Перечень информационных справочных систем и профессиональных баз данных

7. МАТЕРИАЛЬНО-ТЕХНИЧЕСКОЕ ОБЕСПЕЧЕНИЕ

Ауд.	Назначение	Вид	Оснащение
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;

114	Учебная аудитория для занятий лекционного типа, практических занятий	Лек	1 шт. - Компьютер в сборе; 1 шт. - Проектор Acer X118 DLP 3600Lm; 1 шт. - Экран Lumien Eco Picture 200x200 см; 19 шт. - Рулонные шторы; 4 шт. - Шкаф книжный; 26 шт. - Стол студенческий; 46 шт. - Стул; 1 шт. - Стол преподавательский.
123	Учебная лаборатория (компьютерный класс) Кабинет курсового и дипломного проектирования, самостоятельной работы обучающихся	Ср	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.

8. МЕТОДИЧЕСКИЕ УКАЗАНИЯ ДЛЯ ОБУЧАЮЩИХСЯ

--