

Документ подписан простой электронной подписью  
Информация о владельце:  
ФИО: Котова Лариса Анатольевна  
Должность: Директор филиала  
Дата подписания: 13.02.2016г.  
Уникальный программный ключ:  
10730ffe6b1ed0341744b6e9d97700b86e5c04a7  
«Национальный исследовательский технологический университет «МИСИС»

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

## Рабочая программа дисциплины (модуля)

# Алгоритмизация и программирование

Закреплена за подразделением

Кафедра математики и естествознания (Новотроицкий филиал)

Направление подготовки

09.03.03 Прикладная информатика

Профиль

Прикладная информатика в технических системах

Квалификация	<b>Бакалавр</b>		
Форма обучения	<b>очная</b>		
Общая трудоемкость	<b>8 ЗЕТ</b>		
Часов по учебному плану	288	Формы контроля в семестрах:	
в том числе:		экзамен 2	
аудиторные занятия	102	зачет 1	
самостоятельная работа	159		
часов на контроль	27		

### Распределение часов дисциплины по семестрам

Семестр (<Курс>.<Семестр на курсе>)	1 (1.1)		2 (1.2)		Итого
	Недель	19	18	34	
Вид занятий	УП	РП	УП	РП	УП
Лекции	17	17	17	17	34
Лабораторные	34	34	34	34	68
В том числе инт.	17	17	17	17	34
Итого ауд.	51	51	51	51	102
Контактная работа	51	51	51	51	102
Сам. работа	57	57	102	102	159
В том числе сам. работа в рамках ФОС		16		16	
Часы на контроль			27	27	27
Итого	108	108	180	180	288

Программу составил(и):  
кпп, Доцент, Абдулвелеева Р.Р.

Рабочая программа

**Алгоритмизация и программирование**

Разработана в соответствии с ОС ВО:

Самостоятельно устанавливаемый образовательный стандарт высшего образования - бакалавриат Федеральное государственное автономное образовательное учреждение высшего образования «Национальный исследовательский технологический университет «МИСИС» по направлению подготовки 09.03.03 Прикладная информатика (приказ от 05.03.2020 г. № 95 о.в.)

Составлена на основании учебного плана:

09.03.03 Прикладная информатика, 09.03.03\_25\_Прикладная информатика\_ПрПИвТСplx Прикладная информатика в технических системах, утвержденного Ученым советом ФГАОУ ВО НИТУ "МИСиС" в составе соответствующей ОПОП ВО 25.12.2024, протокол № 58

Утверждена в составе ОПОП ВО:

09.03.03 Прикладная информатика, Прикладная информатика в технических системах, утвержденной Ученым советом ФГАОУ ВО НИТУ "МИСиС" 25.12.2024, протокол № 58

Рабочая программа одобрена на заседании

**Кафедра математики и естествознания (Новотроицкий филиал)**

Протокол от 12.03.2025 г., №3

Руководитель подразделения Швалёва А.В.

**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:	Б1.0
2.1	<b>Требования к предварительной подготовке обучающегося:</b>
2.2	<b>Дисциплины (модули) и практики, для которых освоение данной дисциплины (модуля) необходимо как предшествующее:</b>
2.2.1	Дискретная математика
2.2.2	Информационные системы и технологии
2.2.3	Компьютерная графика
2.2.4	Теория вероятностей и математическая статистика
2.2.5	Теория систем и системный анализ
2.2.6	Технологии программирования
2.2.7	Языки и среды разработки интернет-приложений
2.2.8	Программная инженерия
2.2.9	Проектирование информационных систем
2.2.10	Проектирование систем SCADA
2.2.11	Моделирование metallургических процессов с использованием современных программных продуктов
2.2.12	Подготовка к процедуре защиты и защита выпускной квалификационной работы
2.2.13	Теоретическая механика
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

**Владеть:**

УК-1-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	KM	Выполненные 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	ОПК-7-31	Л1.4 Л1.5 Л1.6Л2.1Л3. 2 Э1			
1.2	Линейные алгоритмы. Разветвляющиеся алгоритмы. Циклические алгоритмы /Лаб/	1	4	УК-1-У1	Л1.4 Л1.5 Л1.6Л2.1Л3. 2 Э1		KM1	P1
	<b>Раздел 2. Язык программирования Python</b>							
2.1	Знакомство со средой программирования. Основные базовые конструкции. Типы данных. Коллекции Python. /Лек/	1	13	ОПК-7-31	Л1.1 Л1.2 Л1.3 Л1.6 Л1.8Л2.1Л3. 1 Э1			
2.2	Составление программ линейной, разветвляющейся, циклической структуры /Лаб/	1	6	УК-1-У1	Л1.3 Л1.7 Л1.8Л2.1 Э1		KM2	P2
2.3	Типы данных. Множества. Строки. Индексация. Срезы. Списки. Кортежи. Преобразование коллекций. Методы списков и строк. /Лаб/	1	16	УК-1-31	Л1.7 Л1.8Л2.1 Э1		KM2	P3
2.4	Вложенные списки. Словари. Использование функций. /Лаб/	1	8	УК-1-31	Л1.2 Л1.5Л2.1 Э1			P4
2.5	Изучение материалов в электронной библиотеке и в среде Canvas. /Cр/	1	12	УК-1-31 УК-1-У1	Л1.1 Л1.3Л2.2Л3. 1 Э1			
2.6	Подготовка отчётов по лабораторным работам /Cр/	1	17	УК-1-У1	Э1			P1,P2,P3,P4
2.7	Подготовка к зачёту/Cр/	1	6	УК-1-В1	Л1.5Л2.1Л3. 1 Э1			
2.8	Введение в ООП. Принципы ООП. Классы. Проектирование и разработка классов. Понятие библиотеки языка программирования. Библиотеки Python. PyQT. Файлы, таблицы. /Лек/	2	17	ОПК-7-31	Л1.3 Л1.4 Л1.5 Л1.6Л2.1Л3. 1 Э1		KM5	
2.9	Объектно-ориентированное программирование в среде PyQt5 /Лаб/	2	18	УК-1-31	Л1.3 Л1.5 Л1.6Л2.1Л3. 1 Э1		KM5	P5
2.10	Изучение материалов в электронной библиотеке, в информационно-справочных системах; аналитическая обработка текста (конспектирование, реферирование) /Cр/	2	30	УК-1-31 УК-1-У1	Э1		KM5	

2.11	Подготовка и сдача лабораторных работ. /Cр/	2	42	УК-1-31 УК-1-У1 УК-1-В1	Э1		KM5	P5
2.12	Подготовка к экзамену /Cр/	2	14	УК-1-31 УК-1-У1 УК-1-В1	Э1		KM3, K M5	
2.13	Выполнение домашней работы /Cр/	1	6		Л1.1 Л1.4 Л1.5Л3.1 Э1		KM4	
2.14	Объектно ориентированное программирование. Библиотеки для получения и обработки данных /Лаб/	2	16	УК-1-31 УК-1-У1 УК-1-В1	Л1.4 Л1.5 Л1.6 Э1		KM5	P6
	<b>Раздел 3. Подготовка к контрольным мероприятиям и выполняемым работам</b>							
3.1	Объем часов самостоятельной работы на подготовку к КМ /Cр/	2	12					
3.2	Объем часов самостоятельной работы на подготовку к ВР /Cр/	2	4					
	<b>Раздел 4. Подготовка к контрольным мероприятиям и выполняемым работам</b>							
4.1	Объем часов самостоятельной работы на подготовку к КМ /Cр/	1	8					
4.2	Объем часов самостоятельной работы на подготовку к ВР /Cр/	1	8					

## 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
KM1	Контрольные вопросы	ОПК-7-31	<p>1. Какие основные этапы включает в себя решение задач на компьютере?</p> <p>2. Какие этапы компьютерного решения задач осуществляются без участия компьютера?</p> <p>3. Что называют математической моделью объекта или явления?</p> <p>4. Из каких последовательных действий состоит процесс разработки программы?</p> <p>5. Что называется алгоритмом?</p> <p>6. Перечислите основные свойства алгоритма</p> <p>7. Какими графическими символами принято изображать в схемах алгоритм?</p> <p>8. В чем отличие циклической структуры с предусловием от циклической структуры с постусловием?</p> <p>9. Приведите пример известных алгоритмов.</p>
KM2	Контрольные вопросы	УК-1-31	<p>1. Чем тестирование программы отличается от её отладки?</p> <p>2. Можно ли с помощью тестирования доказать правильность программы?</p> <p>3. На какой стадии работы над программой вычисляются эталонные результаты тестов?</p> <p>4. Назовите основные этапы процесса тестирования.</p> <p>5. В чём заключается отличие синтаксических ошибок от семантических?</p> <p>6. О чём свидетельствует отсутствие сообщений машины о синтаксических ошибках?</p> <p>7. Какие разновидности ошибок транслятор не в состоянии обнаружить?</p>

KM3	Экзамен	УК-1-В1	<ol style="list-style-type: none"> <li>1. Понятие алгоритма. Свойства алгоритмов. Основные алгоритмические конструкции.</li> <li>2. Этапы разработки алгоритмов. Алгоритмическая реализация информационных задач. Примеры алгоритмов.</li> <li>3. Линейные алгоритмы. Разветвляющиеся алгоритмы. Циклические алгоритмы. Пример.</li> <li>4. Встроенный тип str. Методы объекта str.</li> <li>5. print() и форматирование вывода.</li> <li>6. Работа с файловой системой средствами Python.</li> <li>7. Работа с файлами. Методы open(), close(), read(), write().</li> <li>8. Встроенные типы последовательностей list, tuple, range и их методы.</li> <li>9. Встроенный объект dict и его методы.</li> <li>10. Встроенные типы чисел — int, float, complex. Машинное представление чисел с плавающей точкой и целых.</li> <li>11. Множества. Встроенные типы set и frozenset.</li> <li>12. Инструкции и синтаксис. Составные конструкции и обработка исключений</li> <li>13. Инструкции if/else/elif, логические операторы и выражения сравнения</li> <li>14. Циклы while и for в Python</li> <li>15. Модульный подход к разработке программ. Функции. Назначение функций, виды функций, описание и определение функций.</li> <li>16. Функции Python.. Основные понятия. Области видимости и пространство имен в Python.</li> <li>17. Передача аргументов в функцию. Специальные режимы сопоставления аргументов.</li> <li>18. Введение в объектно-ориентированное программирование. Принципы ООП.</li> <li>19. Объекты. Динамическая типизация. Инкапсуляция.</li> <li>20. Генерация объекта class. Новое пространство имен. Объект экземпляр класса.</li> <li>21. Классы. Проектирование и разработка классов. Понятие библиотеки языка программирования. Библиотеки в Python.</li> <li>22. Атрибуты класса. Атрибуты данных. Атрибуты-методы. Параметр self. Добавление атрибутов к классу во время исполнения программы.</li> <li>23. Специальные методы и атрибуты классов. Методы __init__() и __del__() в Python. Декораторы функций и декораторы классов.</li> <li>24. Абстрактные методы в Python. Классические классы и классы нового стиля.</li> <li>25. Наследование. Базовый и производный класс. Построение производного класса.</li> <li>26. Полиморфизм. Подмена методов в производном классе. Доступ к методам базового класса.</li> <li>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.</li> <li>28. Модули и пакеты. Библиотеки сторонних разработчиков (Django, Flask, Pygame, PyQt).</li> <li>29. Модули и пакеты. Графический интерфейс.</li> <li>30. Взаимодействие Python с базой данных SQLite.</li> </ol>
-----	---------	---------	---

KM4	Домашняя работа	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. Сфера 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>Словарии. 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 хэндбука</p> <p>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</p>
-----	-----------------	----------	---

KM5	Диагностические задания	УК-1-31	<p>1. Кроссплатформенное программное обеспечение – это программное обеспечение, работающее:</p> <ol style="list-style-type: none"> <li>1) на одной аппаратной платформе и/или операционной системе;</li> <li>2) на двух аппаратных платформах и/или операционных системах;</li> <li>3) более чем на одной аппаратной платформе и/или операционной системе;</li> <li>4) более чем на двух аппаратных платформах и/или операционных системах.</li> </ol> <p>2. Процесс создания приложения можно разделить на следующие этапы:</p> <ol style="list-style-type: none"> <li>5) написание программного кода, описание свойств элементов;</li> <li>6) формирование окна программы;</li> <li>7) отладка программы;</li> <li>8) тестирование;</li> <li>9) разработка справочной системы.</li> </ol> <p>3. Укажите компонент, который служит для размещения текста на форме:</p> <ol style="list-style-type: none"> <li>1) Label;</li> <li>2) Button;</li> <li>3) Edit;</li> <li>4) Image;</li> <li>5) GroupBox.</li> </ol> <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"> <li>1) Label;</li> <li>2) Button;</li> <li>3) Edit;</li> <li>4) Image;</li> <li>5) OpenDialog.</li> </ol> <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"> <li>1) GroupBox;</li> <li>2) RadioButton;</li> <li>3) ListBox;</li> <li>4) ComboBox;</li> </ol> <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"> <li>1) GroupBox;</li> <li>2) RadioButton;</li> <li>3) ListBox;</li> <li>4) ComboBox;</li> <li>5) Timer.</li> </ol> <p>7. Вещественное число – это ...</p> <ol style="list-style-type: none"> <li>1) тип данных;</li> <li>2) идентификатор;</li> <li>3) служебное слово;</li> <li>4) свойство переменных.</li> </ol> <p>8. Язык высокого уровня - это</p> <ol style="list-style-type: none"> <li>1) Реализация языка Ассемблера</li> <li>2) Язык программирования, наиболее приближенный к человеческому языку</li> <li>3) Машинный язык</li> </ol>
-----	-------------------------	---------	--

		<p>4) Мнемоническое представление машинного языка</p> <p>9. Компонент, представляющий сочетание выпадающего списка и односторочного текстового поля, которое позволяет пользователю ввести значение вручную или выбрать из списка:</p> <ol style="list-style-type: none"> <li>1) GroupBox;</li> <li>2) RadioButton;</li> <li>3) ListBox;</li> <li>4) ComboBox;</li> <li>5) Timer.</li> </ol> <p>10. ShowMessage() осуществляет;</p> <ol style="list-style-type: none"> <li>1) Вывод сообщения</li> <li>2) Результат компилирования</li> <li>3) Вызов окна сообщения ошибок</li> </ol> <p>11. Результатом выполнения операции <math>5 // 2</math> является число... Запишите число: _____</p> <p>12. Результатом выполнения операции <math>5 \% 2</math> является число... Запишите число: _____</p> <p>13. В языке Python объектами являются:</p> <ol style="list-style-type: none"> <li>1) экземпляры классов и переменные</li> <li>2) экземпляры классов, переменные и функции</li> <li>3) экземпляры классов, классы и переменные</li> <li>4) все типы данных</li> </ol> <p>14. В результате выполнения программы будет выведено</p> <hr/> <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"> <li>1) foo</li> <li>2) del</li> <li>3) del foo</li> <li>4) foo del</li> </ol> <p>23. Укажите результат выполнения программы:</p> <pre>x=0 class Foo:     count=x     def __init__(self):         self.count+=1     def __del__(self):         self.count+=1 obj = Foo() print (obj.count)</pre> <p>Варианты ответа:</p> <ol style="list-style-type: none"> <li>1) 0</li> <li>2) 1</li> <li>3) 2</li> <li>4) пустая строка</li> <li>5) скрипт не будет выполнен, т.к. код содержит ошибки</li> </ol> <p>24. Укажите результат выполнения скрипта:</p> <pre>class Foo:     count=0     def __init__(self):         self.count+=1 obj = Foo() print (obj.count)</pre> <p>Варианты ответа:</p> <ol style="list-style-type: none"> <li>1) 0</li> </ol>
--	--	--

- 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=' ')  
    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

KM6	Зачёт	ОПК-7-31	<ol style="list-style-type: none"> <li>1. Понятие алгоритма. Свойства алгоритмов. Основные алгоритмические конструкции.</li> <li>2. Этапы разработки алгоритмов. Алгоритмическая реализация информационных задач. Примеры алгоритмов.</li> <li>3. Язык программирования Python. Его место среди языков программирования. Сфера применения.</li> <li>4. Условный оператор. Примеры применения.</li> <li>5. Линейные алгоритмы. Разветвляющиеся алгоритмы. Циклические алгоритмы. Пример.</li> <li>6. Встроенный тип str. Методы объекта str.</li> <li>7. print() и форматирование вывода.</li> <li>8. Встроенные типы последовательностей list, tuple, range и их методы.</li> <li>9. Коллекции в Python.</li> <li>10. Словари. Особенности. Методы словарей.</li> <li>11. Множества. Особенности. Методы.</li> <li>12. Алгоритм сортировки в Python. Примеры использования.</li> <li>13. Списки. Особенности. Методы списков</li> <li>14. Списочные выражения. Вложенные списки. Массивы.</li> <li>15. Кортежи. Особенности. Примеры использования</li> </ol>
-----	-------	----------	---

**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-1-31	<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-1-31	<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	<p>Типы данных. Множества. Строки. Индексация. Срезы.</p> <p>Списки. Кортежи. Преобразование коллекций. Методы списков и строк.</p> <p><a href="https://academy.yandex.ru/handbook/python/article/stroki-kortezhi-spiski">https://academy.yandex.ru/handbook/python/article/stroki-kortezhi-spiski</a></p>
P4	Лабораторная работа 4	УК-1-В1	<p>Вложенные списки. Словари. Использование функций.</p> <p><a href="https://academy.yandex.ru/handbook/python/article/funkcii-oblasti-vidimosti-peredacha-parametrov-v-funkcii">https://academy.yandex.ru/handbook/python/article/funkcii-oblasti-vidimosti-peredacha-parametrov-v-funkcii</a></p>
P5	Лабораторная работа	УК-1-В1;УК-1-У1;УК-1-31	Объектно-ориентированное программирование
P6	Лабораторная работа ООП	УК-1-31;УК-1-У1;УК-1-В1	<p><a href="https://academy.yandex.ru/handbook/python/article/obuektnaya-model-pyton-klassy-polya-i-metody">https://academy.yandex.ru/handbook/python/article/obuektnaya-model-pyton-klassy-polya-i-metody</a></p> <p><a href="https://academy.yandex.ru/handbook/python/article/volshebnye-metody-pereopredelenie-metodov-nasledovanie">https://academy.yandex.ru/handbook/python/article/volshebnye-metody-pereopredelenie-metodov-nasledovanie</a></p> <p><a href="https://academy.yandex.ru/handbook/python/article/model-isklyuchenijs-pyton-try-except-else-finally-moduli">https://academy.yandex.ru/handbook/python/article/model-isklyuchenijs-pyton-try-except-else-finally-moduli</a></p> <p><a href="https://academy.yandex.ru/handbook/python/article/moduli-math-i-numpy">https://academy.yandex.ru/handbook/python/article/moduli-math-i-numpy</a></p> <p><a href="https://academy.yandex.ru/handbook/python/article/modul-pandas">https://academy.yandex.ru/handbook/python/article/modul-pandas</a></p> <p><a href="https://academy.yandex.ru/handbook/python/article/modul-requests">https://academy.yandex.ru/handbook/python/article/modul-requests</a></p>

**5.3. Оценочные материалы, используемые для экзамена (описание билетов, тестов 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: \_\_\_\_\_ 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 = 'Рауза'
print(f"Меня зовут {name}. Друзья зовут меня {name[:1] + 'о' + 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»: `poprav = 'тише едишь – дольше будишь'`

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

- 1 `replace('O', 'A')`
- 2 `replace('иш', 'еш')`
- 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. Использованы тестовые данные и приведены результаты работы программы (файла)

7. Высокое качество оформления работы с использованием правил оформления текста в текстовом редакторе

8. Использованы 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, эл. адрес
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,
Л1.4	Гниденко И.Г.	Технологии и методы программирования: учеб.пособие		М.: Юрайт, 2019,
Л1.5	Е.Д. Агафонов, Г.В. Вашенко	Прикладное программирование: учебное пособие		, 2015, <a href="http://biblioclub.ru/index.php?page=book&amp;id=435640">http://biblioclub.ru/index.php? page=book&amp;id=435640</a>
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, <a href="https://biblioclub.ru/index.php?page=book&amp;id=83542">https://biblioclub.ru/index.php? page=book&amp;id=83542</a>

#### 6.1.2. Дополнительная литература

Авторы, составители	Заглавие	Библиотека	Издательство, год, эл. адрес
---------------------	----------	------------	------------------------------

	Авторы, составители	Заглавие	Библиотека	Издательство, год, эл. адрес
Л2.1	Ю.А.Аляев, О.А.Козлов	Алгоритмизация и языки программирования Pascal, C++, Visual Basic : Учебно- справочное пособие		М. : Финансы и статистика, 2002,
Л2.2	И. А. Нагаева, И. А. Кузнецов	Алгоритмизация и программирование: практикум: учебное пособие		Москва:Берлин : Директ-Медиа, , 2019, <a href="https://biblioclub.ru/index.php?page=book&amp;id=570287">https://biblioclub.ru/index.php? page=book&amp;id=570287</a>

### 6.1.3. Методические разработки

	Авторы, составители	Заглавие	Библиотека	Издательство, год, эл. адрес
Л3.1	Е.Н. Гусева, И.Ю. Ефимова, Р.И. Коробков и др..	Информатика : учебное пособие		Москва : Издательство «Флинта», 2016., <a href="http://biblioclub.ru/index.php?page=book&amp;id=83542">http://biblioclub.ru/index.php? page=book&amp;id=83542</a>
Л3.2	Абдулвелеева Р.Р., Абдулвелеева Р.Р.	Объектно-ориентированное программирование в среде Lazarus: Лабораторный практикум		НФ НИТУ МИСиС, 2020, <a href="http://elibrary.misis.ru">http://elibrary.misis.ru</a> ; <a href="http://www.nf.misis.ru">www.nf.misis.ru</a>

### 6.2. Перечень ресурсов информационно-телекоммуникационной сети «Интернет»

Э1	Хендбуки Академии Яндекса. Бесплатные онлайн-учебники для тех, кто хочет освоить ключевые ИТ-дисциплины.	<a href="https://academy.yandex.ru/handbook">https://academy.yandex.ru/handbook</a>
----	--	---

### 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	Wing 101
П.6	PyCharm

### 6.4. Перечень информационных справочных систем и профессиональных баз данных

7. МАТЕРИАЛЬНО-ТЕХНИЧЕСКОЕ ОБЕСПЕЧЕНИЕ		
Ауд.	Назначение	Оснащение
123	Учебная лаборатория (компьютерный класс) Кабинет курсового и дипломного проектирования, самостоятельной работы обучающихся	Комплект учебной мебели на 12 мест для обучающихся, 12 стационарных компьютеров для студентов, 1 стационарный компьютер для преподавателя (у всех выход в интернет), проектор, экран, коммутатор, веб камера, доска-флипчарт магн.-маркерная передвижная, доступ к ЭИОС Университета МИСиС через личный кабинет на платформе LMS Canvas и Moodle, лицензионные программы MS Office, MS Teams, антивирус Dr.Web.
123	Учебная лаборатория (компьютерный класс) Кабинет курсового и дипломного проектирования, самостоятельной работы обучающихся	Комплект учебной мебели на 12 мест для обучающихся, 12 стационарных компьютеров для студентов, 1 стационарный компьютер для преподавателя (у всех выход в интернет), проектор, экран, коммутатор, веб камера, доска-флипчарт магн.-маркерная передвижная, доступ к ЭИОС Университета МИСиС через личный кабинет на платформе LMS Canvas и Moodle, лицензионные программы MS Office, MS Teams, антивирус Dr.Web.
121	Учебная аудитория для занятий лекционного типа, практических занятий	Комплект учебной мебели на 56 мест для обучающихся, 1 стационарный компьютер для преподавателя (выход в интернет), проектор, экран настенный, колонки, доска аудиторная меловая, веб камера Logitech, лицензионные программы MS Office, MS Teams, антивирус Dr.Web.

### 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 заченной 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 прослеживать 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/>.

Электронный образовательный ресурс позволяет использовать специальный контент и элементы электронного обучения и дистанционных образовательных технологий. Электронный курс используется преимущественно для асинхронного взаимодействия между участниками образовательного процесса посредством сети «Интернет». Чтобы эффективно использовать возможности электронный курс, а соответственно и успешно освоить дисциплину, нужно:

- 1) зарегистрироваться на курс.
- 2) ознакомиться с содержанием курса, вопросами для самостоятельной подготовки, условиями допуска к аттестации, формой промежуточной аттестации (зачет/экзамен), критериями оценивания и др.;
- 3) изучать учебные материалы, размещенные преподавателем. В т.ч. пользоваться литературой, рекомендованной преподавателем, переходя по ссылкам;
- 4) ознакомиться с содержанием задания к письменной работе, сроками сдачи, критериями оценки. В установленные сроки

выполнить работу(ы), подгрузить здесь же для проверки. Удобно называть файл работы следующим образом (название предмета (сокращенно), группа, ФИО, дата актуализации (при повторном размещении)).

Работа, подгружаемая для проверки, должна:

- 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; 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 обратится к материалу и заново его просмотреть.