

**МИНИСТЕРСТВО НАУКИ И ВЫСШЕГО ОБРАЗОВАНИЯ  
РОССИЙСКОЙ ФЕДЕРАЦИИ**

**Федеральное государственное автономное  
образовательное учреждение высшего образования  
«Национальный исследовательский Нижегородский государственный университет  
им. Н.И. Лобачевского»**

**Дзержинский филиал ННГУ**

**УТВЕРЖДЕНО**

решением Ученого совета ННГУ

(протокол от «30» ноября 2022 г. № 13)

**Рабочая программа дисциплины  
АЛГОРИТМИЗАЦИЯ И ПРОГРАММИРОВАНИЕ**

Уровень высшего образования

**БАКАЛАВРИАТ**

Направление подготовки

**09.03.03 ПРИКЛАДНАЯ ИНФОРМАТИКА**

Направленность (профиль) образовательной программы

**ИТ-СЕРВИСЫ И ТЕХНОЛОГИИ ОБРАБОТКИ ДАННЫХ В ЭКОНОМИКЕ И  
ФИНАНСАХ**

*Год набора: 2023*

Квалификация

**БАКАЛАВР**

Форма обучения

**ОЧНАЯ, ОЧНО-ЗАОЧНАЯ**

Дзержинск  
2022 г.

## 1. Место дисциплины в структуре ОПОП

Дисциплина Б1.О.13 Алгоритмизация и программирование относится к обязательной части учебного плана ООП 09.03.03 Прикладная информатика.

Целью освоения дисциплины является формирование у студентов теоретических знаний и практических навыков по основам алгоритмизации и программированию в области экономических и управленческих информационных систем.

## 2. Планируемые результаты обучения по дисциплине, соотнесенные с планируемыми результатами освоения образовательной программы (компетенциями и индикаторами достижения компетенций)

Формируемые компетенции (код, содержание компетенции)	Планируемые результаты обучения по дисциплине (модулю), в соответствии с индикатором достижения компетенции		Наименование оценочного средства
	Индикатор достижения компетенции (код, содержание индикатора)	Результаты обучения по дисциплине	
ОПК-2 Способен понимать принципы работы современных информационных технологий и программных средств, в том числе отечественного производства, и использовать их при решении задач профессиональной деятельности	ОПК-2.1 Демонстрирует знание принципов работы современных информационных технологий и программных средств, в том числе отечественного производства	Знать: принципы работы современных информационных технологий и программных средств. Уметь: выбирать современные информационные технологии и программные средства, в том числе отечественного производства при решении прикладных задач. Владеть: современными подходами и стандартами, автоматизации задач профессиональной деятельности.	Собеседование, тестирование, отчет о выполнении лабораторных работ.
	ОПК-2.2 Демонстрирует умение применять современные информационные технологии и программные средства, в том числе отечественного производства, для решения задач профессиональной деятельности.	Знать: алгоритмические и программные средства в информационных технологиях. Уметь: применять современные программные средства для решения задач профессиональной деятельности. Владеть: навыками решения задач	Собеседование, тестирование, отчет о выполнении лабораторных работ.

Формируемые компетенции (код, содержание компетенции)	Планируемые результаты обучения по дисциплине (модулю), в соответствии с индикатором достижения компетенции		Наименование оценочного средства
	Индикатор достижения компетенции (код, содержание индикатора)	Результаты обучения по дисциплине	
		профессиональной деятельности с применением современных информационных технологий.	
	ОПК-2.3 Демонстрирует наличие практического опыта решения задач профессиональной деятельности с применением современных информационных технологий и программных средств, в том числе отечественного производства.	Знать: современные информационные технологии и программные средства. Уметь: выбирать современные информационные технологии и программные средства, в том числе отечественного производства при решении задач профессиональной деятельности. Владеть: навыками решения задач профессиональной деятельности с применением современных информационных технологий.	Отчет о выполнении лабораторных работ.
ОПК-3 Способен решать стандартные задачи профессиональной деятельности на основе информационной и библиографической культуры с применением информационно-коммуникационных технологий и с учетом основных требований информационной	ОПК-3.1 Демонстрирует знание принципов, методов и средств решения стандартных задач профессиональной деятельности на основе информационной и библиографической культуры с применением информационно-коммуникационных технологий и с учетом основных требований информационной безопасности	Знать: этапы и стадии подготовки и решения задач на компьютере. Уметь: решать стандартные задачи профессиональной деятельности с применением современных языков программирования. Владеть: навыками решения задач с учетом основных требований информационной безопасности.	Собеседование, тестирование, отчет о выполнении лабораторных работ.

Формируемые компетенции (код, содержание компетенции)	Планируемые результаты обучения по дисциплине (модулю), в соответствии с индикатором достижения компетенции		Наименование оценочного средства
	Индикатор достижения компетенции (код, содержание индикатора)	Результаты обучения по дисциплине	
безопасности	ОПК-3.2 Демонстрирует умение применять информационно- коммуникационные технологии решения стандартных задач профессиональной деятельности на основе информационной и библиографической культуры с учетом основных требований информационной безопасности	Знать: современные языки программирования. Уметь: применять высокоуровневые методы программирования для решения задач профессиональной деятельности с учетом основных требований информационной безопасности. Владеть: навыками разработки программ на языке программирования Python.	Собеседование, тестирование, отчет о выполнении лабораторных работ.
	ОПК-3.3 Имеет практический опыт решения стандартных задач профессиональной деятельности с соблюдением требований информационной безопасности	Знать: алгоритмически структуры. Уметь: разрабатывать алгоритмы для решения задач профессиональной деятельности. Владеть: навыками анализа алгоритма и его сложности, программирования, проверки (отладки) программы.	Отчет о выполнении лабораторных работ.
ОПК-4 Способен участвовать в разработке стандартов, норм и правил, а также технической документации, связанной с профессиональной деятельностью	ОПК-4.1 Демонстрирует знание основных стандартов, норм и правил оформления технической документации на различных стадиях проектирования и поддержки жизненного цикла информационных систем	Знать: обозначения элементов блок-схем. Уметь: представлять алгоритм в виде блок- схемы. Владеть: навыками разработки блок-схем алгоритмов в соответствии со стандартом.	Собеседование, тестирование, отчет о выполнении лабораторных работ.
	ОПК-4.2 Применяет стандарты, нормы и правила (в том числе установленные	Знать: требования к оформлению технической документации.	Собеседование, тестирование, отчет о выполнении

Формируемые компетенции (код, содержание компетенции)	Планируемые результаты обучения по дисциплине (модулю), в соответствии с индикатором достижения компетенции		Наименование оценочного средства
	Индикатор достижения компетенции (код, содержание индикатора)	Результаты обучения по дисциплине	
	самостоятельно) при оформлении технической документации на различных стадиях проектирования и поддержки жизненного цикла информационных систем	Уметь: ориентироваться в стандартах, нормах и правилах применяемых при оформлении технической документации. Владеть: оформлять документацию в соответствии с принятыми стандартами, нормами и правилами.	лабораторных работ.
	ОПК-4.3 Имеет практический опыт разработки технической документации на различных этапах проектирования и поддержки жизненного цикла информационной системы.	Знать: содержание основных разделов технической документации. Уметь: разрабатывать сопроводительную документацию на разных этапах разработки. Владеть: навыками составления документации.	Отчет о выполнении лабораторных работ.
ОПК-5 Способен устанавливать программное и аппаратное обеспечение для информационных и автоматизированных систем	ОПК-5.1 Демонстрирует знание основ системного администрирования и современных стандартов информационного взаимодействия систем	Знать: современные технологии реализации информационных систем, языки программирования. Уметь: решать задачи производственной и технологической деятельности на профессиональном уровне, включая: разработку алгоритмических и программных решений в области системного и прикладного программирования. Владеть: навыками разработки и отладки информационных систем.	Собеседование, тестирование, отчет о выполнении лабораторных работ.

Формируемые компетенции (код, содержание компетенции)	Планируемые результаты обучения по дисциплине (модулю), в соответствии с индикатором достижения компетенции		Наименование оценочного средства
	Индикатор достижения компетенции (код, содержание индикатора)	Результаты обучения по дисциплине	
	ОПК-5.2 Демонстрирует умение выполнять параметрическую настройку ИС.	Знать: основы параметрической настройки ИС, инсталляции программного обеспечения информационных систем. Уметь: учитывать особенности параметрической настройки ИС при разработке программного обеспечения. Владеть: навыками параметрической настройки ИС.	Собеседование, отчет о выполнении лабораторных работ.
	ОПК-5.3 Имеет практический опыт инсталляции программного и аппаратного обеспечения информационных систем.	Знать: принципы многопоточного программирования. Уметь: осуществлять инсталляцию программного обеспечения. Владеть: навыками инсталляции программного обеспечения.	Отчет о выполнении лабораторных работ.
ОПК-7 Способен разрабатывать алгоритмы и программы, пригодные для практического применения	ОПК-7.1 Демонстрирует знание основных языков программирования и работы с базами данных, операционных систем и оболочек, современных программных сред разработки информационных систем и технологий	Знать: языки структурного и объектно-ориентированного программирования, среды разработки ПО для данных языков. Уметь: применять языки структурного и объектно-ориентированного программирования. Владеть: навыками программирования на языках высокого уровня.	Собеседование, тестирование, отчет о выполнении лабораторных работ.

Формируемые компетенции (код, содержание компетенции)	Планируемые результаты обучения по дисциплине (модулю), в соответствии с индикатором достижения компетенции		Наименование оценочного средства
	Индикатор достижения компетенции (код, содержание индикатора)	Результаты обучения по дисциплине	
	ОПК-7.2 Применяет языки программирования и работы с базами данных, современные программные среды разработки информационных систем и технологий для автоматизации бизнес-процессов, решения прикладных задач различных классов, ведения баз данных и информационных хранилищ	Знать: основы применения современных языков программирования для автоматизации бизнес-процессов. Уметь: применять современные программные среды разработки информационных систем и технологий для автоматизации бизнес-процессов. Владеть: навыками решения прикладных задач различных классов с применением языков программирования.	Собеседование, отчет о выполнении лабораторных работ.
	ОПК-7.3 Имеет практический опыт программирования, отладки и тестирования прототипов программно-технических комплексов задач	Знать: основы программирования, отладки и тестирования программно-технических комплексов задач. Уметь: осуществлять программирование, отладку и тестирование прототипов программно-технических комплексов задач. Владеть: навыками программирования, отладки и тестирования прототипов программно-технических комплексов задач.	Отчет о выполнении лабораторных работ.
ПК-3 Способен вводить в эксплуатацию и осуществлять сопровождение ИС	ПК-3.1 Демонстрирует знание методологических и технических основ ввода ИС в эксплуатацию	Знать: методологические и технические основы ввода ИС в эксплуатацию.	Собеседование, тестирование, отчет о выполнении лабораторных работ.

<b>Формируемые компетенции</b> (код, содержание компетенции)	<b>Планируемые результаты обучения по дисциплине (модулю), в соответствии с индикатором достижения компетенции</b>		<b>Наименование оценочного средства</b>
	<b>Индикатор достижения компетенции</b> (код, содержание индикатора)	<b>Результаты обучения по дисциплине</b>	
на всех этапах ее жизненного цикла, включая ее презентацию и начальное обучение пользователей		Уметь: разрабатывать инструменты для ввода ИС в эксплуатацию. Владеть: навыками ввода ИС в эксплуатацию.	работ.
	ПК-3.2 Демонстрирует умение организовать репозиторий хранения данных о создании ИС, вводе ее в эксплуатацию и модификации в процессе жизненного цикла	Знать: основные этапы жизненного цикла ИС. Уметь: организовать репозиторий хранения данных о создании ИС, вводе ее в эксплуатацию и модификации. Владеть: навыками управления ИС на всех стадиях ЖЦ.	Собеседование, отчет о выполнении лабораторных работ.
	ПК-3.3 Имеет практический опыт инсталляции программного обеспечения ИС, его тестирования и начального обучения пользователей	Знать: основы инсталляции программного обеспечения ИС, его тестирование и начального обучение пользователей. Уметь: инсталлировать и тестировать разработанное ПО. Владеть: навыками начального обучения пользователей.	Отчет о выполнении лабораторных работ.

### 3. Структура и содержание дисциплины

#### 3.1 Трудоемкость дисциплины

	<b>очная форма обучения</b>	<b>очно-заочная форма обучения</b>
<b>Общая трудоемкость</b>	7 ЗЕТ	7 ЗЕТ
<b>Часов по учебному плану</b>	252	252
<b>в том числе</b>		



	очная форма обучения	очно-заочная форма обучения
<b>аудиторные занятия (контактная работа):</b>	115	66
- занятия лекционного типа	32	22
- занятия семинарского типа	16	8
- лабораторные занятия	64	32
- КСР	3	4
<b>самостоятельная работа</b>	101	150
<b>Промежуточная аттестация – зачёт, экзамен</b>	36	36

### 3.2. Содержание дисциплины

Наименование и краткое содержание разделов и тем дисциплины	Всего (часы)			в том числе												Самостоятельная работа обучающегося, часы		
				Контактная работа (работа во взаимодействии с преподавателем), часы														
				из них														
	Занятия лекционного типа			Занятия семинарского типа			Занятия лабораторного типа			Всего								
Очная	Очно-заочная	Заочная	Очная	Очно-заочная	Заочная	Очная	Очно-заочная	Заочная	Очная	Очно-заочная	Заочная	Очная	Очно-заочная	Заочная	Очная	Очно-заочная	Заочная	
Тема 1. Теоретические основы алгоритмизации и программирования.	16	14		4	4		2	1		2	1		8	6		8	8	
Тема 2. Введение в Python.	5	7		1	1								1	1		4	6	
Тема 3. Линейный алгоритм.	9	11		1	1		2	1		2	1		5	3		4	8	
Тема 4. Разветвляющийся алгоритм.	18	13		2	1		4	2		4	2		10	5		8	8	
Тема 5. Циклический алгоритм.	18	13		2	1		4	2		4	2		10	5		8	8	
Тема 6. Списки. Кортежи. Словари.	27	23		4	2		2	1		8	4		14	7		13	16	
Тема 7. Строки.	18	21		4	2		2	1		4	2		10	5		8	16	
Тема 8. Обработка вложенных последовательностей.	14	20		2	2					4	2		6	4		8	16	
Тема 9. Функции. Модули.	14	20		2	2					4	2		6	4		8	16	
Тема 10. Работа с файлами.	14	20		2	2					4	2		6	4		8	16	

Наименование и краткое содержание разделов и тем дисциплины	Всего (часы)			в том числе														
				Контактная работа (работа во взаимодействии с преподавателем), часы из них												Самостоятельная работа обучающегося, часы		
				Занятия лекционного типа			Занятия семинарского типа			Занятия лабораторного типа			Всего					
	Очная	Очно-заочная	Заочная	Очная	Очно-заочная	Заочная	Очная	Очно-заочная	Заочная	Очная	Очно-заочная	Заочная	Очная	Очно-заочная	Заочная	Очная	Очно-заочная	Заочная
Тема 11. Объектно-ориентированное программирование.	36	26		4	2					16	8		20	10		16	16	
Тема 12. Событийно-ориентированное программирование.	24	24		4	2					12	6		16	8		8	16	
КСР	3	4											3	4				
Промежуточная аттестация	36	36																
Итого	252	252		32	22		16	8		64	32		115	66		101	150	

Тема 1. Теоретические основы алгоритмизации и программирования

Алгоритм. Свойства алгоритма. Способы описания алгоритма. Назначение функциональных блоков. Основные этапы решения задач.

Алфавит языка Python. Идентификаторы и общие правила их написания.

Оператор присваивания. Арифметические выражения. Деление нацело. Остаток от деления. Типы данных. Функции приведения типов. Запись математических функций. Операции отношения.

Тема 2. Введение в Python.

Процесс создания проекта в Python. Методы ввода и вывода данных. Обработка исключений.

Тема 3. Линейный алгоритм.

Тема 4. Разветвляющийся алгоритм.

Сокращенный условный оператор if. Простой и составной условный оператор if ... else. Многочисленные ветвления if ... elif... else. Логические переменные. Логические операции И, ИЛИ, НЕ. Сложные условия. Алгоритмы поиска максимального и минимального элементов.

Тема 5. Циклический алгоритм.

Оператор цикла for. Оператор цикла while.

Тема 6. Списки. Кортежи. Словари.

Работа со списками. Объявление кортежей. Способы обработки кортежей. Работа со словарями.

Тема 7. Строки.

Основные понятия. Функции, методы работы со строками. Алгоритмы обработки строк.

Тема 8. Обработка вложенных последовательностей.

Формирование вложенных последовательностей. Алгоритмы обработки вложенных последовательностей.

Тема 9. Функции. Модули.

Создание пользовательских функций. Создание модулей.

Тема 10. Работа с файлами.

Чтение, запись информации в текстовый файл.

Тема 11. Объектно-ориентированное программирование.

Создание классов, конструкторов. Инкапсуляция. Создание свойств. Наследование.

Тема 12. Событийно-ориентированное программирование.

Создание форм и виджетов Кнопка. Текстовое поле. Надпись. Создание виджетов Флажок, Переключатель.

Практические занятия, лабораторные работы организуются, в том числе в форме практической подготовки, которая предусматривает участие обучающихся в выполнении отдельных элементов работ, связанных с будущей профессиональной деятельностью.

Практическая подготовка предусматривает: выполнение проекта по профилю профессиональной деятельности и направленности образовательной программы.

На проведение практических занятий (лабораторных работ) в форме практической подготовки отводится 6 часов.

Практическая подготовка направлена на формирование и развитие:

- практических навыков в соответствии с профилем ОП: формирование и анализ требований к информатизации и автоматизации прикладных процессов, формализация предметной области проекта; программирование приложений, создание прототипа информационной системы;

- компетенций – ОПК-7 способен разрабатывать алгоритмы и программы, пригодные для практического применения.

Текущий контроль успеваемости реализуется в рамках занятий практического и лабораторного типа.

#### **4. Учебно-методическое обеспечение самостоятельной работы обучающихся**

Основные виды самостоятельной работы студентов:

- работа с основной и дополнительной литературой;
- изучение категориального аппарата дисциплины;
- самостоятельное изучение тем дисциплины;
- подготовка к экзамену.

Работа с основной и дополнительной литературой

Изучение рекомендованной литературы следует начинать с учебников и учебных пособий, затем переходить к научным монографиям и материалам периодических изданий. Работа с литературой предусматривает конспектирование наиболее актуальных и познавательных материалов. Это не только мобилизует внимание, но и способствует более глубокому осмыслению материала, его лучшему запоминанию, а также позволяет студентам проводить систематизацию и сравнительный анализ изучаемой информации. Таким образом, конспектирование – одна из основных форм самостоятельного труда, которая требует от студента активно работать с учебной литературой и не ограничиваться конспектом лекций.

Студент должен уметь самостоятельно подбирать необходимую литературу для учебной и научной работы, уметь обращаться с предметными каталогами и библиографическим справочником библиотеки.

#### Изучение категориального аппарата дисциплины

Изучение и осмысление экономических категорий требует проработки лекционного материала, выполнения практических заданий, изучение словарей, энциклопедий, справочников.

Индивидуальная самостоятельная работа студента направлена на овладение и грамотное применение экономической терминологии в области компьютерного моделирования.

#### Самостоятельное изучение тем дисциплины

Особое место отводится самостоятельной проработке студентами отдельных разделов и тем изучаемой дисциплины. Такой подход вырабатывает у студентов инициативу, стремление к увеличению объема знаний, умений и навыков, всестороннего овладения способами и приемами профессиональной деятельности.

Изучение вопросов определенной темы направлено на более глубокое усвоение основных категорий экономической теории, понимание экономических процессов, происходящих в обществе, совершенствование навыка анализа теоретического и эмпирического материала.

#### Подготовка к экзамену

Промежуточная аттестация студентов по дисциплине проходит в виде экзамена и предусматривает оценку. Условием успешного прохождения промежуточной аттестации является систематическая работа студента в течение семестра. В этом случае подготовка к экзамену является систематизацией всех полученных знаний по данной дисциплине.

Рекомендуется внимательно изучить перечень вопросов к экзамену, а также использовать в процессе обучения программу, учебно-методический комплекс, другие методические материалы.

Желательно спланировать трехкратный просмотр материала перед экзаменом. Во-первых, внимательное чтение с осмыслением, подчеркиванием и составлением краткого плана ответа. Во-вторых, повторная проработка наиболее сложных вопросов. В-третьих, быстрый просмотр материала или планов ответов для его систематизации в памяти.

Контрольные вопросы и задания для проведения текущего контроля и промежуточной аттестации по итогам освоения дисциплины приведены в п. 5.2.

Для обеспечения самостоятельной работы обучающихся используется электронный курс Алгоритмизация и программирование <https://e-learning.unn.ru/course/view.php?id=11158>, созданный в системе электронного обучения ННГУ – <https://e-learning.unn.ru/>.

### **5. Фонд оценочных средств для промежуточной аттестации по дисциплине (модулю), Включающий**

### 5.1.Описание шкал оценивания результатов обучения по дисциплине

Уровень сформированности компетенций (индикатор достижения компетенций)	Шкала оценивания сформированности компетенций						
	плохо	неудовлетворительно	удовлетворительно	хорошо	очень хорошо	отлично	превосходно
	не зачтено		зачтено				
<u>Знания</u>	Отсутствие знаний теоретического материала. Невозможность оценить полноту знаний вследствие отказа обучающегося от ответа	Уровень знаний ниже минимальных требований. Имели место грубые ошибки.	Минимально допустимый уровень знаний. Допущено много негрубых ошибок.	Уровень знаний в объеме, соответствующем программе подготовки. Допущено несколько негрубых ошибок	Уровень знаний в объеме, соответствующем программе подготовки. Допущено несколько незначительных ошибок	Уровень знаний в объеме, соответствующем программе подготовки, без ошибок.	Уровень знаний в объеме, превышающем программу подготовки.
<u>Умения</u>	Отсутствие минимальных умений. Невозможность оценить наличие умений вследствие отказа обучающегося от ответа	При решении стандартных задач не продемонстрированы основные умения. Имели место грубые ошибки.	Продemonстрированы основные умения. Решены типовые задачи с негрубыми ошибками. Выполнены все задания, но не в полном объеме.	Продemonстрированы все основные умения. Решены все основные задачи с негрубыми ошибками. Выполнены все задания, в полном объеме, но некоторые с недочетами.	Продemonстрированы все основные умения. Решены все основные задачи. Выполнены все задания, в полном объеме, но некоторые с недочетами.	Продemonстрированы все основные умения, решены все основные задачи с отдельными незначительными недочетами, выполнены все задания в полном объеме.	Продemonстрированы все основные умения,. Решены все основные задачи. Выполнены все задания, в полном объеме без недочетов
<u>Навыки</u>	Отсутствие владения материалом. Невозможность оценить наличие навыков вследствие отказа обучающегося от ответа	При решении стандартных задач не продемонстрированы базовые навыки. Имели место грубые ошибки.	Имеется минимальный набор навыков для решения стандартных задач с некоторыми недочетами	Продemonстрированы базовые навыки при решении стандартных задач с некоторыми недочетами	Продemonстрированы базовые навыки при решении стандартных задач без ошибок и недочетов.	Продemonстрированы навыки при решении нестандартных задач без ошибок и недочетов.	Продemonстрирован творческий подход к решению нестандартных задач

### Шкала оценки при промежуточной аттестации

Оценка		Уровень подготовки
зачтено	Превосходно	Все компетенции (части компетенций), на формирование которых направлена дисциплина, сформированы на уровне не ниже «превосходно»

Оценка		Уровень подготовки
	Отлично	Все компетенции (части компетенций), на формирование которых направлена дисциплина, сформированы на уровне не ниже «отлично», при этом хотя бы одна компетенция сформирована на уровне «отлично»
	Очень хорошо	Все компетенции (части компетенций), на формирование которых направлена дисциплина, сформированы на уровне не ниже «очень хорошо», при этом хотя бы одна компетенция сформирована на уровне «очень хорошо»
	Хорошо	Все компетенции (части компетенций), на формирование которых направлена дисциплина, сформированы на уровне не ниже «хорошо», при этом хотя бы одна компетенция сформирована на уровне «хорошо»
	Удовлетворительно	Все компетенции (части компетенций), на формирование которых направлена дисциплина, сформированы на уровне не ниже «удовлетворительно», при этом хотя бы одна компетенция сформирована на уровне «удовлетворительно»
не зачтено	Неудовлетворительно	Хотя бы одна компетенция сформирована на уровне «неудовлетворительно», ни одна из компетенций не сформирована на уровне «плохо»
	Плохо	Хотя бы одна компетенция сформирована на уровне «плохо»

## 5.2. Типовые контрольные задания или иные материалы, необходимые для оценки результатов обучения.

### 5.2.1 Контрольные вопросы

Вопрос	Код компетенции
1. Дать определение понятию алгоритм.	ОПК-7
2. Назовите и дайте характеристику основным свойствам алгоритма.	ОПК-7
3. Назовите способы описания алгоритма.	ОПК-3
4. Назовите основные этапы решения задач.	ОПК-4
5. Что такое идентификаторы? Назовите общие правила написания идентификаторов.	ОПК-5
6. Назовите функции приведения типов данных.	ПК-3
7. Назовите методы ввода и вывода данных в Python.	ОПК-5
8. Что такое линейный алгоритм?	ОПК-7
9. Изобразите блок-схему сокращенного условного оператора.	ОПК-3
10. Изобразите блок-схему простого условного оператора.	ОПК-3
11. Изобразите блок-схему многозначного ветвления.	ОПК-3
12. Алгоритмы поиска максимального и минимального элементов.	ОПК-2
13. Оператор цикла for.	ПК-3
14. Оператор цикла while.	ПК-3
15. Что такое списки?	ПК-3
16. Что такое кортежи?	ПК-3
17. Что такое словари?	ПК-3
18. Строки. Основные параметры.	ПК-3
19. Алгоритмы обработки строк.	ПК-3

Вопрос	Код компетенции
20. Что такое функции?	ОПК-2
21. Что такое модули?	ОПК-2
22. Чтение, запись информации в текстовый файл.	ОПК-5
23. Создание классов. Конструктор классов.	ПК-3
24. Инкапсуляция. Наследование.	ОПК-3

### 5.2.2. Типовые тестовые задания для оценки сформированности компетенции

#### ОПК-2

1. Последовательность команд, понятных транслятору (компилятору и или интерпретатору), которые могут быть переведены в машинный код и исполнены на компьютере – это ...

- А) Программа
- Б) Алгоритм
- В) Структура данных

2. Комплекс программных средств, используемых программистами для разработки программного обеспечения – это ...

- А) Интегрированная среда разработки
- Б) Язык программирования
- В) Оператор

3. Программа, преобразующая исходный текст программы на языке программирования в машинный язык вычислительной системы, на которой эта программ должна выполняться – это ...

- А) Транслятор
- Б) Интерпретатор
- В) Компилятор

4. Программа, преобразующая текст, написанный на алгоритмическом языке, в программу, состоящую из машинных команд – это ...

- А) Транслятор
- Б) Интерпретатор
- В) Компилятор

5. Самостоятельная часть программы, которая разрабатывается независимо от других частей и затем вызывается по имени – это...

- А) Подпрограмма
- Б) Программа
- В) Оператор

#### ОПК-3

6. Цикл ... осуществляет поворотное выполнение сформированного кода на основе переменной или счетчика.

- А) for
- Б) while
- В) Цикл с постусловием

7. Цикл ... используется в сценарии, когда заранее неизвестно количество итераций. Блок операторов в цикле ... выполняется до тех пор, пока будет выполняться условие.

- А) for
- Б) while
- В) Цикл с постусловием

8. [] – это...

- А) Список
- Б) Словарь

- В) Кортеж
9. Метод, позволяющий осуществить добавление элемента в конец списка
- А) append()
  - Б) insert()
  - В) remove()
10. Встроенная функция, предназначенная для получения количества элементов списка – это...
- А) len()
  - Б) max()
  - В) filter(функция, последовательность)

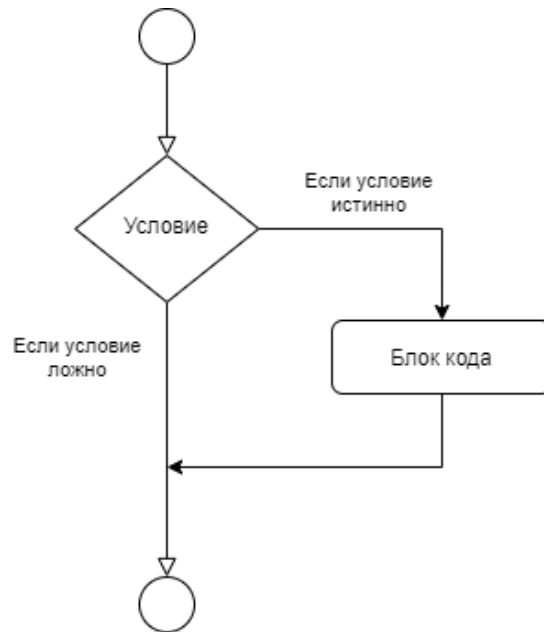
#### ОПК-4

11. Последовательность элементов, которые разделены между собой запятой и заключены в круглые скобки. Неизменяемый упорядоченный тип данных.
- А) Список
  - Б) Словарь
  - В) Кортеж
12. Изменяемый неупорядоченный набор пар «ключ : значение» – это...
- А) Список
  - Б) Словарь
  - В) Кортеж
13. Метод для работы со строкой, позволяющий перевести символы нижнего регистра в верхний, а верхнего – в нижний.
- А) swapcase()
  - Б) upper()
  - В) lower()
14. Метод для работы со строкой, позволяющий осуществить поиск подстроки в строке
- А) find()
  - Б) count()
  - В) replace()
15. Метод для работы со строкой, позволяющий определить число вхождений подстроки в строку
- А) find()
  - Б) count()
  - В) replace()

#### ОПК-5

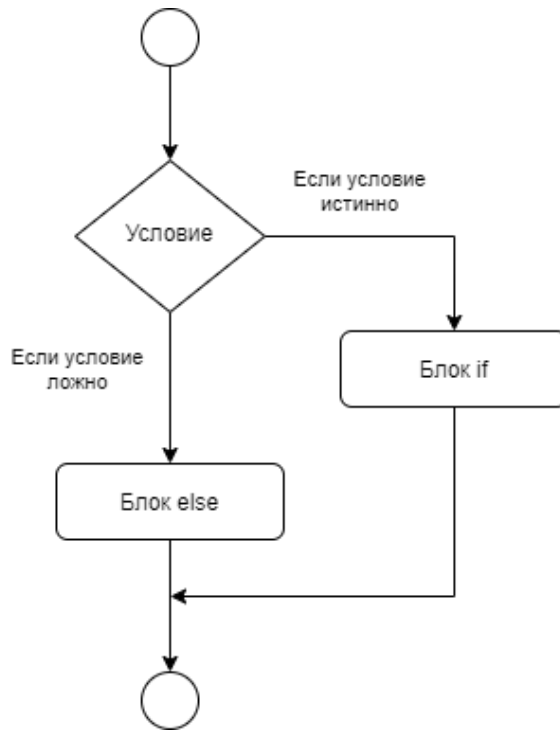
16. Дополнительные функции, сгруппированные по темам, хранящиеся в отдельном файле – это ...
- А) Модуль
  - Б) Функция
  - В) Подпрограмма
17. Представленной блок-схеме соответствует следующая форма записи условного оператора





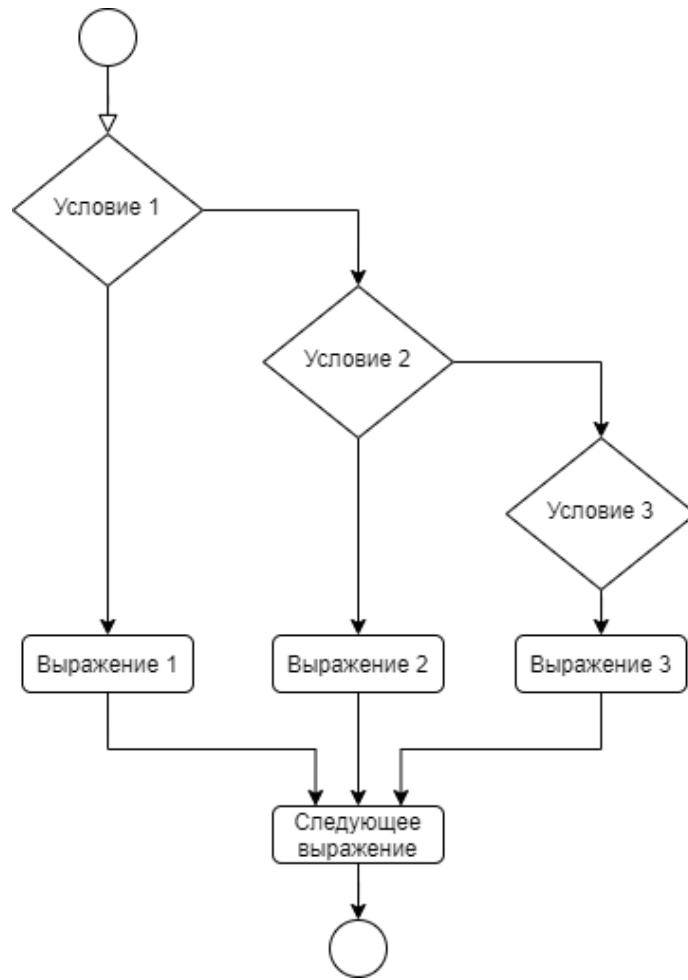
- A) if <логическое выражение>:  
     <действия, выполняемые, когда логическое выражение принимает значение True >
- Б) if <логическое выражение>:  
     <действия, выполняемые, когда логическое выражение принимает значение True >  
   else:  
     <действия, выполняемые, когда логическое выражение принимает значение False >
- В) if <логическое выражение>:  
     <действия, выполняемые, если логическое выражение принимает значение True >  
   elif <второе логическое выражение>:  
     <действия, выполняемые, если второе логическое выражение принимает значение True >  
   elif <третье логическое выражение>:  
     <действия, выполняемые, если третье логическое выражение принимает значение True >  
   ...  
   else:  
     <действия, выполняемые, если ни одно из логических выражений не принимает значение True >

18. Представленной блок-схеме соответствует следующая форма записи условного оператора



- A) if <логическое выражение>:  
     <действия, выполняемые, когда логическое выражение принимает значение True >
- Б) if <логическое выражение>:  
     <действия, выполняемые, когда логическое выражение принимает значение True >  
   else:  
     <действия, выполняемые, когда логическое выражение принимает значение False >
- В) if <логическое выражение>:  
     <действия, выполняемые, если логическое выражение принимает значение True >  
   elif <второе логическое выражение>:  
     <действия, выполняемые, если второе логическое выражение принимает значение True >  
   elif <третье логическое выражение>:  
     <действия, выполняемые, если третье логическое выражение принимает значение True >  
   ...  
   else:  
     <действия, выполняемые, если ни одно из логических выражений не принимает значение True >

19. Представленной блок-схеме соответствует следующая форма записи условного оператора



- A) if <логическое выражение>:  
     <действия, выполняемые, когда логическое  
     выражение принимает значение True >
- Б) if <логическое выражение>:  
     <действия, выполняемые, когда логическое  
     выражение принимает значение True >  
   else:  
     <действия, выполняемые, когда логическое  
     выражение принимает значение False >
- В) if <логическое выражение>:  
     <действия, выполняемые, если логическое  
     выражение принимает значение True >  
   elif <второе логическое выражение>:  
     <действия, выполняемые, если второе логическое  
     выражение принимает значение True >  
   elif <третье логическое выражение>:  
     <действия, выполняемые, если третье логическое  
     выражение принимает значение True >  
   ...  
   else:  
     <действия, выполняемые, если ни одно из  
     логических выражений не принимает значение True >

## ОПК-7

20. Целочисленное деление обозначается

- A) `x / y`
  - Б) `x // y`
  - В) `x % y`
21. Остаток от целочисленного деления обозначается
- A) `x / y`
  - Б) `x // y`
  - В) `x % y`
22. Проверка на вхождение
- A) `x in s`
  - Б) `s[i]`
  - В) `s[i:j:k]`
23. Получение среза
- A) `x in s`
  - Б) `s[i]`
  - В) `s[i:j:k]`
24. Обращение к элементу по индексу
- A) `x in s`
  - Б) `s[i]`
  - В) `s[i:j:k]`

### ПК-3

25. Метод объекта `dict`, который возвращает объект, содержащий все ключи словаря и поддерживающий итерации и все операции над множествами – это ...
- A) `d.keys()`
  - Б) `d.values()`
  - В) `d.items()`
26. Метод объекта `dict`, который возвращает объект, содержащий все значения словаря – это ...
- A) `d.keys()`
  - Б) `d.values()`
  - В) `d.items()`
27. Метод объекта `dict`, который возвращает объект, содержащий кортежи (ключ, значение) – это ...
- A) `d.keys()`
  - Б) `d.values()`
  - В) `d.items()`
28. Метод для работы со строкой, позволяющий заменить все символы строки соответствующими строчными буквами
- A) `swapcase()`
  - Б) `upper()`
  - В) `lower()`
29. Метод для работы со строкой, позволяющий осуществить поиск подстроки в строке
- A) `find()`
  - Б) `count()`
  - В) `replace()`
30. Инструкция, которая создаст функцию, принимающую указанный список аргументов и возвращающую результат вычисления выражения – это...
- A) лямбда-функция
  - Б) функция `map`
  - В) функция `filter`

### 5.2.3. Типовые задания для лабораторных работ для оценки сформированности компетенции

#### ОПК-2

### 1. Линейный алгоритм.

– Даны действительные числа А, В, С. Найти максимальное и минимальное из этих чисел.

– Для заданных значений вычислить результат:

$$\frac{\sqrt{|x|} \ln(x^2)}{-\frac{5}{4}x + e^{x/2}}$$

ОПК-3

### 2. Разветвляющийся алгоритм.

– Можно ли из бревна, имеющего диаметр поперечного сечения D, выпилить квадратный брус шириной А?

– Имеются две ёмкости: кубическая с ребром А, цилиндрическая с высотой Н и радиусом основания R. Определить, поместится ли жидкость объёма М в первую ёмкость, во вторую, в обе.

– Заданы числа k, l, m (целые) и x. вычислить величины y(x) и z(y):

$$y(x) = \begin{cases} \frac{\sin|x|}{\sqrt{x^2+1}}, & \text{если } m = \min(k, l) \\ |x| \ln(1+x), & \text{если } m = \max(k, l) \\ x^3 + x + 10^{-2}, & \text{если } m \in [\min(k, l), \max(k, l)] \\ -1, & \text{в остальных случаях} \end{cases}$$

$$z(y) = y^4 - y^2 + 5$$

ОПК-7

### 3. Циклический алгоритм.

– Начав тренировки, спортсмен в 1-ый день пробежал X км. Каждый день он увеличивал дневную норму на 10 % нормы предыдущего дня. Какой суммарный путь пробежит спортсмен за 1 неделю? ( X = 10 )

– Найти сумму всех 4-значных чисел, кратных k.

– Напишите программу, используя цикл с переменной (for), которая считывает одно число k, после чего для каждого из чисел от 0 до k (включительно) выводит фразу: «Корень числа [такого-то] равен [тому-то], а куб равен [тому-то]».

– Напишите программу, которая по двум данным натуральным числам а и b, не превосходящим 30 000, подсчитывает:

– сумму натуральных чисел, уменьшенных на заданное число d, на отрезке [a, b] (включая концы отрезка);

– произведение натуральных чисел, уменьшенных на заданное число d, на отрезке [a, b];

– сумму натуральных чисел, увеличенных в заданное число раз d, на отрезке [a, b];

– среднее арифметическое натуральных чисел, увеличенных на заданное число d, на отрезке [a, b];

– среднее арифметическое натуральных чисел, уменьшенных на заданное число d, на отрезке [a, b];

Программа получает на вход три натуральных числа а, b и d. Требуется осуществить проверку входных данных на корректность. Результаты вычислений необходимо вывести на экран.

ПК-3

### 4. Списки. Кортежи. Словари.

– Пользователь вводит свою электронную почту в формате name@mail.\*. Почта должна относиться к почтовому серверу yandex.ru, gmail.com, mail.ru. Создайте программу, которая записывает данные об электронной почте в файл, предусмотрев обработку исключений при неправильном вводе почтового сервера.

– Дан одномерный массив числовых значений, насчитывающий N элементов. Поменять местами элементы, стоящие на чётных и нечётных местах.

– Дан одномерный массив числовых значений, насчитывающий N элементов. Вставить группу из M новых элементов, начиная с позиции K.

– Дан одномерный массив числовых значений, насчитывающий N элементов. Исключить все нулевые элементы.

#### ОПК-5

##### 5. Строки.

– Создать программу, если дана строка из слов, разделенных пробелами. Переставить слова в обратном порядке и вывести результат – новую строку.

– С клавиатуры вводится символьная строка. Определите, сколько в ней цифр.

– С клавиатуры вводится предложение, в котором каждые два слова разделены ровно пробелом. Определите, сколько слов в этом предложении.

– Дана строка, содержащая английский текст. Найти количество слов, начинающихся с буквы b.

– Удалить из текста символы «\_» и подсчитать длину сформированного текста.

– Напишите программу, в которой пользователь вводит одну строку, а выводится та же строка, но каждый символ повторяется дважды.

– Найти в строке указанную подстроку и заменить ее на новую. Строку, ее подстроку для замены и новую подстроку вводит пользователь.

– Вводится строка слов, разделенных пробелами. Найти самое длинное слово и вывести его на экран. Предусмотреть случай, когда самых длинных слов может быть несколько.

– Определить, является ли введенное слово идентификатором, т.е. начинается ли оно с английской буквы в любом регистре или знака подчеркивания и не содержит других символов, кроме букв английского алфавита (в любом регистре), цифр и знака подчеркивания.

#### ОПК-4

##### 6. Функции. Модули.

– Напишите функцию, которая удаляет из переданной ей символьной строки все лишние пробелы, заменяя повторяющиеся пробелы на один пробел.

– Составьте лямбда-функцию, которая вычисляет налог на доход физических лиц (НДФЛ) в размере 13% от начисленной заработной платы, которую вводит пользователь.

– Составьте модуль, который содержит функции вычисления чистой заработной платы (заработная плата на руки после вычета налога 13%), а также начисления на фонд оплаты труда (ФОТ) (определяется так, чтобы после вычета 30% получалась начисленная заработная плата, от которой вычитается 13% и остается заработная плата на руки). В основной программе пользователь вводит число сотрудников, для каждого вводит начисленную заработную плату. Программа выводит для каждого сотрудника заработную плату на руки, а также начисления на ФОТ и сумму общего размера ФОТ.

##### 6. Объектно-ориентированное программирование.

Создать класс с полями, указанными рисунок 1.

Реализовать в классе методы:

– конструктор по умолчанию;

– деструктор для освобождения памяти (с сообщением об уничтожении объекта);

– функции обработки данных, указанные в задании 1;

– функцию формирования строки информации об объекте.

Создать проект для демонстрации работы: сформировать объекты со значениями-константами и с введенными с клавиатуры значениями полей объекта. В основной ветке программы создайте три объекта класса. Вывести результаты работы на экран.

Класс-родитель и его поля	Функция-метод 1 обработки данных	Функция-метод 2 обработки данных
Дата (три числа): день, месяц, год	Определить, является ли год високосным (кратным 4)	Увеличить дату на 5 дней

Рисунок 1 – Вариант задания

#### 5.2.4. Вопросы для собеседования

Вопрос	Код компетенции
1. Типы данных.	ОПК-7
2. Назовите и дайте характеристику основным свойствам алгоритма.	ОПК-7
3. Назовите способы описания алгоритма.	ОПК-3
4. Назовите основные этапы решения задач.	ОПК-4
5. Обработка исключений. Регулярные выражения.	ОПК-5
6. Назовите функции приведения типов данных.	ПК-3
7. Функции для вывода данных в Python.	ОПК-5
8. Что такое линейный алгоритм?	ОПК-7
9. Программирование логики программы: условные операторы.	ОПК-3
10. Программирование логики программы: циклы в Python.	ОПК-3
11. Изобразите блок-схему многозначного ветвления.	ОПК-3
12. Статический метод	ОПК-2
13. Операции в Python	ПК-3
14. Переменные: именованное переменных, присваивание значения переменным, удаление переменных.	ПК-3
15. Списки.	ПК-3
16. Кортежи.	ПК-3
17. Словари.	ПК-3
18. Строки.	ПК-3
19. Алгоритмы обработки строк.	ПК-3
20. Функции.	ОПК-2
21. Модули	ОПК-2
22. Файлы.	ОПК-5
23. Объектно-ориентированное программирование (ООП). Классы.	ПК-3
24. Метод в объектно-ориентированном программировании	ОПК-3

#### 6. Учебно-методическое и информационное обеспечение дисциплины

##### Основная литература

1. Андреева, О.В. Основы алгоритмизации и программирования на языке PYTHON: учебник / О.В. Андреева, О.И. Ремизова. — Москва: МИСИС, 2022. — 149 с. — ISBN 978-5-907560-22-2. — Текст: электронный // Лань: электронно-библиотечная система. —

URL: <https://e.lanbook.com/book/263552> (дата обращения: 29.10.2022). — Режим доступа: для авториз. пользователей.

2. Косицин, Д. Ю. Язык программирования Python : учебно-методическое пособие / Д. Ю. Косицин. — Минск : БГУ, 2019. — 136 с. — ISBN 978-985-566-746-0. — Текст : электронный // Лань : электронно-библиотечная система. — URL: <https://e.lanbook.com/book/180546> (дата обращения: 29.10.2022). — Режим доступа: для авториз. пользователей.

#### **Дополнительная литература**

3. Полупанов, Д. В. Программирование в Python 3 : учебное пособие / Д. В. Полупанов, С. Р. Абдюшева, А. М. Ефимов. — Уфа : БашГУ, 2020. — 164 с. — ISBN 978-5-7477-5230-6. — Текст : электронный // Лань : электронно-библиотечная система. — URL: <https://e.lanbook.com/book/179915> (дата обращения: 29.10.2022). — Режим доступа: для авториз. пользователей.
4. Забелин, А. А. Реализация алгоритмов вычислительной математики на языке Python : учебное пособие / А. А. Забелин. — Чита : ЗабГУ, 2020. — 130 с. — ISBN 978-5-9293-2575-5. — Текст : электронный // Лань : электронно-библиотечная система. — URL: <https://e.lanbook.com/book/173632> (дата обращения: 29.10.2022). — Режим доступа: для авториз. пользователей.
5. Янцев, В. В. Web-программирование на Python : учебное пособие для вузов / В. В. Янцев. — Санкт-Петербург : Лань, 2022. — 180 с. — ISBN 978-5-8114-9461-3. — Текст : электронный // Лань : электронно-библиотечная система. — URL: <https://e.lanbook.com/book/233264> (дата обращения: 29.10.2022). — Режим доступа: для авториз. пользователей.
6. Широбокова, С. Н. Программирование на языке Python для лабораторных занятий : учебное пособие / С. Н. Широбокова, А. А. Кацупеев, А. В. Сулыз. — Новочеркасск : ЮРГПУ, 2020. — 104 с. — ISBN 978-5-9997-0725-3. — Текст : электронный // Лань : электронно-библиотечная система. — URL: <https://e.lanbook.com/book/180938> (дата обращения: 29.10.2022). — Режим доступа: для авториз. пользователей.

#### **Программное обеспечение и Интернет-ресурсы**

1. Python. URL: <https://www.python.org/>
2. Python Tutorials. URL: <https://pythonspot.com/all-tutorials/>
3. Full Stack Python. URL: <https://www.fullstackpython.com/>
4. Python для всех. URL: <https://www.py4e.com/lessons>
5. Online Python. URL: <https://www.online-python.com/>
6. Программное обеспечение: менеджер пакетов Anaconda, Spyder (open-source IDE для Python).

### **7. Материально-техническое обеспечение дисциплины**

Помещения представляют собой учебные аудитории для проведения учебных занятий, предусмотренных программой, оснащенные оборудованием и техническими средствами обучения: компьютерного класса, имеющего компьютеры, объединенные сетью с выходом в Интернет, презентационная техника (телевизор, компьютер/ноутбук).

Помещения для самостоятельной работы обучающихся оснащены компьютерной техникой с возможностью подключения к сети "Интернет" и обеспечены доступом в электронную информационно-образовательную среду.

#### **Специальные условия организации обучения по дисциплине для инвалидов и лиц с ограниченными возможностями здоровья**

Организация обучения по дисциплине инвалидов и лиц с ограниченными возможностями здоровья осуществляется с учетом особенностей психофизического развития,



индивидуальных возможностей и состояния здоровья при наличии таких обучающихся путем создания специальных условий для получения образования.

Профессорско-преподавательский состав знакомится с психолого-физиологическими особенностями обучающихся инвалидов и лиц с ограниченными возможностями здоровья, индивидуальными программами реабилитации инвалидов (при наличии).

В соответствии с Методическими рекомендациями по организации образовательного процесса для обучения инвалидов и лиц с ограниченными возможностями здоровья в образовательных организациях высшего образования, в том числе оснащенности образовательного процесса, утв. Минобрнауки РФ 08.04.2014 АК-44/05вн при изучении дисциплины предполагается использование социально-активных и рефлексивных методов обучения, технологий социокультурной реабилитации с целью оказания помощи в установлении полноценных межличностных отношений с другими студентами, создании комфортного психологического климата в студенческой группе.

При освоении дисциплины используются различные сочетания видов учебной работы с методами и формами активизации познавательной деятельности обучающихся для достижения запланированных результатов обучения и формирования компетенций. Форма проведения промежуточной аттестации для обучающихся-инвалидов и лиц с ограниченными возможностями здоровья устанавливается с учетом индивидуальных психофизиологических особенностей. По личной просьбе обучающегося с ограниченными возможностями здоровья, изложенной в форме письменного заявления, по дисциплине предусматриваются:

- замена устного ответа на письменный ответ при сдаче зачета или экзамена;
- увеличение продолжительности времени на подготовку к ответу на зачете или экзамене;
- при подведении результатов промежуточной аттестации студентов выставляется максимальное количество баллов за посещаемость аудиторных занятий.

Программа составлена в соответствии с требованиями ФГОС ВО/ОС ННГУ по направлению 09.03.03 Прикладная информатика (приказ №349-ОД от 21.06.2021).

**Автор(ы):** к.т.н. Кечкина Н.И.

**Рецензент:**

Программа одобрена Методической комиссией Дзержинского филиала ННГУ от 10.11.2022 года, протокол № 12.