

МИНИСТЕРСТВО СЕЛЬСКОГО ХОЗЯЙСТВА
РОССИЙСКОЙ ФЕДЕРАЦИИ
ФЕДЕРАЛЬНОЕ ГОСУДАРСТВЕННОЕ БЮДЖЕТНОЕ ОБРАЗОВАТЕЛЬНОЕ
УЧРЕЖДЕНИЕ ВЫСШЕГО ОБРАЗОВАНИЯ
«ПЕНЗЕНСКИЙ ГОСУДАРСТВЕННЫЙ АГРАРНЫЙ УНИВЕРСИТЕТ»

СОГЛАСОВАНО

Председатель методической
комиссии экономического факультета

 А.Е. Шпагина

«20» февраля 2023 г.

УТВЕРЖДАЮ

Декан
экономического факультета

 И.А. Бондин

«20» февраля 2023 г.

РАБОЧАЯ ПРОГРАММА ДИСЦИПЛИНЫ

Б1.О.14

АЛГОРИТМИЗАЦИЯ И ПРОГРАММИРОВАНИЕ

Направление подготовки
09.03.03 Прикладная информатика

Направленность (профиль) программы
Прикладная информатика в экономике

Квалификация
«БАКАЛАВР»

Форма обучения – очная

Пенза – 2023

Рабочая программа дисциплины «Алгоритмизация и программирование» составлена на основании федерального государственного образовательного стандарта высшего образования - бакалавриат по направлению подготовки 09.03.03 прикладная информатика, утверждённого приказом Минобрнауки РФ от 19 сентября 2017 года № 922.

Составитель рабочей программы:

доцент кафедры «Финансы и информатизация бизнеса»,

(уч. степень, ученое звание)



(подпись)

О.Н. Суханова
(инициалы, ф.)

Рецензент:

канд. техн. наук, доцент

(уч. степень, ученое звание)



(подпись)

А.В. Яшин
(инициалы, ф.)

Рабочая программа одобрена на заседании кафедры «Финансы и информатизация бизнеса»

«20» февраля 2023 года, протокол № 6.

Заведующий кафедрой:

канд. экон. наук, доцент

(уч. степень, ученое звание)



(подпись)

О.А. Тагирова
(инициалы, ф.)

Рабочая программа одобрена на заседании методической комиссии экономического факультета «20» февраля 2023 года, протокол № 7.

Председатель методической комиссии
экономического факультета



РЕЦЕНЗИЯ

на рабочую программу дисциплины
«Алгоритмизация и программирование»
для студентов, обучающихся по направлению подготовки
09.03.03 Прикладная информатика

В рецензируемой рабочей программе представлены учебно-методические материалы, необходимые для организации учебного процесса по дисциплине «Алгоритмизация и программирование» для студентов первого, второго курсов экономического факультета, обучающихся по направлению подготовки 09.03.03 Прикладная информатика.

Рабочая программа составлена в соответствии с требованиями ФГОС ВО по направлению подготовки 09.03.03 Прикладная информатика (профиль) программы «Прикладная информатика в экономике» (утвержден 19.09.2017 приказом Минобрнауки России № 922).

Программа содержит все структурные элементы, предусмотренные нормативными документами Пензенского ГАУ, рассмотрена и утверждена на заседании кафедры «Финансы и информатизация бизнеса» 20 февраля 2023 года, протокол № 6 и одобрена на заседании методической комиссии экономического факультета 20 февраля 2023 года, протокол № 7.

Предложения.

В целом рецензируемая рабочая программа удовлетворяет требованиям ФГОС ВО по направлению подготовки 09.03.03 Прикладная информатика, направленность (профиль) программы «Прикладная информатика в экономике», и нормативным документам Пензенского ГАУ и может быть использована в учебном процессе.

Рецензент:

канд. техн. наук, доцент



Яшин А.В.

Выписка из протокола № 7
заседания методической комиссии экономического факультета
от «20» февраля 2023 г.

Присутствовали члены методической комиссии:

Бондин И.А., Лаврина О.В.,
Бондина Н.Н., Шпагина И.Е.,
Позубенкова Э.И., Столярова О.А.,
Тагирова О.А., Сологуб Н.Н.

Повестка дня

Вопрос 2. Рассмотрение рабочей программы дисциплины «Алгоритмизация и программирование», подготовленной в соответствии с требованиями ФГОС ВО по направлению подготовки 09.03.03 Прикладная информатика (профиль) программы «Прикладная информатика в экономике» (утвержден 19.09.2017 приказом Минобрнауки России № 922).

Слушали: Шпагину И.Е., которая представила рабочую программу дисциплины «Алгоритмизация и программирование» для обучающихся по направлению подготовки 09.03.03 Прикладная информатика (профиль) программы «Прикладная информатика в экономике»

Выступили: Лаврина О.В., которая отметила, что рецензируемая рабочая программа дисциплины «Алгоритмизация и программирование» удовлетворяет требованиям ФГОС ВО по направлению подготовки 09.03.03 Прикладная информатика (профиль) программы «Прикладная информатика в экономике» и нормативным документам Пензенского ГАУ и может быть использована в учебном процессе.

Постановили: утвердить рабочую программу дисциплины «Алгоритмизация и программирование».

Председатель методической комиссии
экономического факультета,
канд. эконом. наук, доцент



Шпагина И.Е.

**Лист регистрации изменений и дополнений к рабочей программе
дисциплины «Алгоритмизация и программирование»**

№ п/п	Раздел	Изменения и дополнения	Дата, № протокола, виза зав. кафедрой	Дата, № протокола, виза председателя методической комиссии	С какой даты вводятся
1	10 Материально-техническая база, необходимая для осуществления образовательного процесса по дисциплине	Новая редакция таблицы 10.1 «Материально-техническое обеспечение дисциплины»	Протокол № 12 от 30.08.23 	Протокол № 9 от 30.08.2023 	01.09.2023

**Лист регистрации изменений и дополнений к рабочей программе
дисциплины «Алгоритмизация и программирование»**

№ п/п	Раздел	Изменения и дополнения	Дата, № протокола, виза зав. кафедрой	Дата, № протокола, виза пред- седателя методиче- ской ко- миссии	С какой даты вво- дятся
1	9 Учебно-методическое и информационное обеспечение дисциплины	Новая редакция таблиц 9.1.1 «Основная литература» и 9.1.2 «Дополнительная литература»	28.08.2024 протокол № 12 	28.08.2024 № 8 	01.09.2024
2	9 Учебно-методическое и информационное обеспечение дисциплины	Новая редакция таблицы 9.2.1 «Перечень информационных технологий (перечень современных профессиональных баз данных и информационных справочных систем, используемых при осуществлении образовательного процесса по дисциплине)»			
3	10 Материально-техническая база, необходимая для осуществления образовательного процесса по дисциплине	Новая редакция таблицы 10.1 «Материально-техническое обеспечение дисциплины»			

**Лист регистрации изменений и дополнений к рабочей программе
дисциплины «Алгоритмизация и программирование»**

№ п/п	Раздел	Изменения и дополнения	Дата, № протокола, виза зав. кафедрой	Дата, № протокола, виза пред- седателя методиче- ской ко- миссии	С какой даты вво- дятся
1	9 Учебно-методическое и информационное обеспечение дисциплины	Новая редакция таблиц 9.1.1 «Основная литература» и 9.1.2 «Дополнительная литература»	23.06.2025 протокол № 11 	29.08.2025 № 6 	01.09.2025
2	9 Учебно-методическое и информационное обеспечение дисциплины	Новая редакция таблицы 9.2.1 «Перечень информационных технологий (перечень современных профессиональных баз данных и информационных справочных систем, используемых при осуществлении образовательного процесса по дисциплине»			
3	10 Материально-техническая база, необходимая для осуществления образовательного процесса по дисциплине	Новая редакция таблицы 10.1 «Материально-техническое обеспечение дисциплины»			

**Лист регистрации изменений и дополнений к рабочей программе
дисциплины «Алгоритмизация и программирование»**

№ п/п	Раздел	Изменения и дополнения	Дата, № протокола, виза зав. кафедрой	Дата, № протокола, виза пред- седателя методиче- ской ко- миссии	С какой даты вво- дятся
1	9 Учебно-методическое и информационное обеспечение дисциплины	Новая редакция таблиц 9.1.1 «Основная литература» и 9.1.2 «Дополнительная литература»	22.06.2026 протокол № 10 	31.08.2026 № 4 	01.09.2026
2	9 Учебно-методическое и информационное обеспечение дисциплины	Новая редакция таблицы 9.2.1 «Перечень информационных технологий (перечень современных профессиональных баз данных и информационных справочных систем, используемых при осуществлении образовательного процесса по дисциплине)»			
3	10 Материально-техническая база, необходимая для осуществления образовательного процесса по дисциплине	Новая редакция таблицы 10.1 «Материально-техническое обеспечение дисциплины»			

1 ЦЕЛЬ И ЗАДАЧИ ДИСЦИПЛИНЫ

Цель дисциплины – изучение понятийного аппарата дисциплины, основных теоретических положений и методов, привитие навыков применения теоретических знаний для решения практических задач.

Задачи дисциплины:

- систематизация, формализация и расширение знаний по основам информатики, приобретенные в школе;
- привитие навыков алгоритмического мышления, культуры алгоритмизации и нисходящего структурного программирования;
- формирование теоретической базы и практических умений и навыков для решения задач на компьютере в императивных системах программирования;
- формирование основ современной культуры программирования.

2 ПЕРЕЧЕНЬ ПЛАНИРУЕМЫХ РЕЗУЛЬТАТОВ ОБУЧЕНИЯ ПО ДИСЦИПЛИНЕ, СООТНЕСЕННЫХ С ПЛАНИРУЕМЫМИ РЕЗУЛЬТАТАМИ ОСВОЕНИЯ ОБРАЗОВАТЕЛЬНОЙ ПРОГРАММЫ БАКАЛАВРИАТА

Дисциплина «Алгоритмизация и программирование» направлена на формирование следующих компетенций:

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-3: способен решать стандартные задачи профессиональной деятельности на основе информационной и библиографической культуры с применением информационно-коммуникационных технологий и с учетом основных требований информационной безопасности.

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

ПК-2: способен разрабатывать и адаптировать прикладное программное обеспечение.

Индикаторы и дескрипторы части соответствующей компетенции, формируемой в процессе изучения дисциплины «Алгоритмизация и программирование», оцениваются при помощи контрольных мероприятий, приведенных в таблице 2.1.

В результате изучения дисциплины «Алгоритмизация и программирование» обучающийся должен получить знания и навыки для успешного осво-

ения следующих трудовых функций и выполнения следующих трудовых действий:

Профессиональный стандарт ПС 06.001 «Программист», утвержденный приказом Министерства труда и социальной защиты Российской Федерации от 20 июля 2022 г. N 424н (зарегистрирован Министерством юстиции Российской Федерации 22 августа 2022 г., регистрационный № 69720).

Обобщенная трудовая функция – ОТФ D/3.4 «Разработка требований и проектирование программного обеспечения».

Трудовая функция – код D/01.6 ТФ 3.4.1 «Анализ возможностей реализации требований к компьютерному программному обеспечению».

Трудовые действия, необходимые умения и знания:

- оценка и согласование сроков выполнения поставленных задач
уметь:
- вырабатывать варианты реализации требований к компьютерному программному обеспечению
- проводить оценку и обоснование рекомендуемых решений
знать:
- возможности существующей программно-технической архитектуры
- возможности современных и перспективных средств разработки программных продуктов, технических средств.

Таблица 2.1 – Планируемые результаты обучения по дисциплине «Алгоритмизация и программирование», индикаторы достижения компетенций ОПК-2, ОПК-3, ОПК-7, ПК-2, перечень контрольных мероприятий

№ п п	Код индикатора достижения компетенции	Наименование индикатора достижения компетенции	Код планируемого результата обучения	Планируемые результаты обучения	Наименование контрольных мероприятий*
1	ИД-2 _{ОПК-2}	Использует современные информационные технологии и программные средства при решении профессиональных задач, в том числе отечественного про-	31 (ИД-2 _{ОПК-2})	Знать: программные средства, в том числе отечественного производства при решении задач профессиональной деятельности.	Зачет; экзамен; тестирование; собеседование.
			У1 (ИД-2 _{ОПК-2})	Уметь: выбирать современные программные средства, в том числе отечественного производ-	Зачет; экзамен; тестирование; курсовая работа.

		изводства		ства при решении задач профессиональной деятельности.	
			В1 (ИД-2 _{ОПК-2})	Владеть: навыками применения современных программных средств, в том числе отечественного производства, при решении задач профессиональной деятельности	Зачет; экзамен; тестирование; курсовая работа.
2	ИД-1 _{ОПК-3}	Способен применять информационно-коммуникационные технологии для решения стандартных задач профессиональной деятельности на основе информационной и библиографической культуры	31 (ИД-1 _{ОПК-3})	Знать: основные структуры данных и методы их обработки; различия между языками программирования высокого и низкого уровня	Зачет; экзамен; тестирование; собеседование.
			У1 (ИД-1 _{ОПК-3})	Уметь: программировать алгоритмы, используя средства языков высокого уровня.	Зачет; экзамен; тестирование; курсовая работа.
			В1 (ИД-1 _{ОПК-3})	Владеть: навыками в проведении анализа получаемых результатов и оформлении документации на программу	Зачет; экзамен; тестирование; курсовая работа.
3	ИД-1 _{ОПК-7}	Применяет языки программирования и работу с базами данных для автоматизации бизнес-процессов, реализацию прикладных задач различных классов, ведение баз данных и информационных хранилищ	31 (ИД-1 _{ОПК-7})	Знать: процесс подготовки и решения задач на ПЭВМ, основы алгоритмизации, приемы программирования на языке высокого уровня	Зачет; экзамен; тестирование; курсовая работа.
			У1 (ИД-1 _{ОПК-7})	Уметь: разрабатывать алгоритмы решения задач, выполнять разработку, тестирование и отладку программ на алгоритмических языках высокого уровня	Зачет; экзамен; тестирование; курсовая работа.
			В1 (ИД-1 _{ОПК-7})	Владеть: навыками разработки прикладных задач.	Зачет; экзамен; тестирование;

					курсовая работа.
4	ИД-2ОПК-7	Осуществляет программирование, отладку и тестирование прототипов программно-технических комплексов задач	31 (ИД-2ОПК-7)	Знать: современные информационные технологии и программные средства, в том числе отечественного производства при решении задач профессиональной деятельности.	Зачет; экзамен; тестирование; курсовая работа.
			У1 (ИД-2ОПК-7)	Уметь: выбирать современные информационные технологии и программные средства, в том числе отечественного производства при решении задач профессиональной деятельности	Зачет; экзамен; тестирование; курсовая работа.
			В1 (ИД-2ОПК-7)	Владеть: навыками применения современных информационных технологий и программных средств, в том числе отечественного производства, при решении задач профессиональной деятельности	Зачет; экзамен; тестирование; курсовая работа.
5	ИД-2ПК-2	Осуществляет организационное и технологическое обеспечение кодирования на языках программирования	31 (ИД-2ПК-2)	Знать: регламенты кодирования на языках программирования	Зачет; экзамен; тестирование; курсовая работа.
			У1 (ИД-2ПК-2)	Уметь: использовать инструменты и методы верификации структуры программного кода	Зачет; экзамен; тестирование; курсовая работа.
			В1 (ИД-2ПК-2)	Владеть: навыками обеспечения соответствия разработанного кода и про-	Зачет; экзамен; тестирование; курсовая ра-

				цесса кодирования на языках программирования принятым в организации или проекте стандартам и технологиям	бота.
--	--	--	--	--	-------

* Оценочные материалы для проведения текущего контроля успеваемости и промежуточной аттестации обучающихся по дисциплине, в т.ч. в форме заданий тестового типа, представлены в Приложении.

Задания тестового типа могут быть использованы при проведении диагностических процедур, в т.ч. диагностической работы, в рамках НОКО.

3 МЕСТО ДИСЦИПЛИНЫ В СТРУКТУРЕ ОБРАЗОВАТЕЛЬНОЙ ПРОГРАММЫ БАКАЛАВРИАТА

Учебная дисциплина «Алгоритмизация и программирование» включена в Блок Б1 «Дисциплины (модули)», обязательная часть (Б1.О.14).

Для успешного усвоения дисциплины необходимо, чтобы студент владел знаниями, умениями и навыками в объеме требований средней школы.

Дисциплина «Алгоритмизация и программирование» является основополагающим для изучения следующих дисциплин: «Программная инженерия», «Базы данных», «Проектирование информационных систем».

4 ОБЪЕМ И СТРУКТУРА ДИСЦИПЛИНЫ

Общая трудоемкость дисциплины составляет 8 зачетных единиц (288 часов).

Таблица 4.1 – Распределение общей трудоемкости дисциплины «Алгоритмизация и программирование» по формам и видам учебной работы

№ п/п	Форма и вид учебной работы	Условное обозначение по учебному плану	Трудоёмкость, ч/з.е.	
			Очная Форма обучения	
			1 курс 2 семестр	2 курс 3 семестр
1	Контактная работа – всего	Контакт часы	65,8/1,828	68,7/1,908
1.1	Лекции	Лек	32,0/0,888	32,0/0,888
1.2	Семинары, и практические занятия	Пр	-/-	-/-
1.3	Лабораторные работы	Лаб	32,0/0,888	32,0/0,888
1.4	Текущие консультации, руководство и консультации курсовых работ (курсовых проектов)	КТ	1,6/0,044	2,1/0,58
1.5	Сдача зачета (зачёта с оценкой), защита курсовой работы (курсового проекта)	КЗ	0,2/0,005	0,25/0,007
1.6	Предэкзаменационные консультации по дисциплине	КПЭ	-/-	2/0,055
1.7	Сдача экзамена	КЭ	-/-	0,35/0,009
2	Общий объем самостоятельной работы		42,2/1,172	77,65/2,157
2.1	Самостоятельная работа	СР	42,2/1,172	77,65/2,157
2.2	Контроль (самостоятельная подготовка к сдаче экзамена)	Контроль	-/-	33,65/0,934
Всего		По плану	108,0/3,0	180,0/5,0
			108,0/3,0	180,0/5,0

Форма промежуточной аттестации:

2 семестр– зачет.

3 семестр– курсовая работа, экзамен.

5 СОДЕРЖАНИЕ ДИСЦИПЛИНЫ

Таблица 5.1 – Наименование разделов дисциплины «Алгоритмизация и программирование» и их содержание

№ раздела	Наименование раздела дисциплины	Содержание раздела	Код планируемого результата обучения
1	2	3	4
1	Алгоритмизация процессов обработки данных	Понятие алгоритма и его свойства. Способы записи алгоритмов Основные алгоритмические конструкции Алгоритмы линейного, разветвляющегося и циклического процесса	31 (ИД-2ОПК-2) У1 (ИД-2ОПК-2) 31 (ИД-1ОПК-3) У1 (ИД-1ОПК-3) 31 (ИД-1ОПК-7) У1 (ИД-1ОПК-7) 31 (ИД-2ОПК-7) 31 (ИД-2ПК-2)
2	Введение в программирование. Управляющие операторы языка высокого уровня. Структуры данных	Программы и программное обеспечение. Методология разработки программных продуктов Понятие программы и программного обеспечения. Парадигмы программирования. Жизненный цикл программы. Языки программирования Языки программирования и их виды. Трансляторы и их виды. Системы программирования Структура программы и основные элементы языка Python. Элементы языка Python: алфавит, служебные слова, константы, переменные, имена, типы данных. Встроенные функции языка. Понятие выражений. Типы выражений. Структура программы на Python. Правила формирования и вычисления выражений. Операторы языка Python Виды операторов языка Python: простые, сложные (структурные), составные. Элементарный ввод и вывод на Python. Операторы при-	31 (ИД-2ОПК-2) У1 (ИД-2ОПК-2) В1 (ИД-2ОПК-2) 31 (ИД-1ОПК-3) У1 (ИД-1ОПК-3) В1 (ИД-1ОПК-3) 31 (ИД-1ОПК-7) У1 (ИД-1ОПК-7) В1 (ИД-1ОПК-7) 31 (ИД-2ОПК-7) У1 (ИД-2ОПК-7) В1 (ИД-2ОПК-7) 31 (ИД-2ПК-2) В1 (ИД-2ПК-2) У1 (ИД-2ПК-2)

		сваивания, условного и без- условного переходов. Опе- раторы условия. Оператор выбора. Операторы цикла. Типы данных языка Python Концепция типа данных. Простые (целые, веще- ственные, строковые, логи- ческие) и структурирован- ные (массив, множество, запись, файл) типы данных.	
3	Программирование базо- вых алгоритмов обработ- ки данных	Алгоритмы поиска данных Виды поиска, поисковый запрос. Поиск в неупорядо- ченном и в упорядоченном множестве данных. Оценка эффективности поиска. Алгоритмы упорядочивания данных Виды сортировки (внутрен- няя и внешняя). Основные методы упорядочивания данных. Оценка эффектив- ности сортировки. Алгоритмы обработки од- номерных массивов Объявление массива. Ини- циализация. Ввод и вывод одномерных массивов. Стандартные процедуры обработки массива. Удале- ние и вставка элементов в массив. Алгоритмы обработки дву- мерных массивов Объявление массива. Ини- циализация. Ввод и вывод двумерных массивов. Стан- дартные процедуры обра- ботки массива и элементов массива по условию.	31 (ИД-2опк-2) У1 (ИД-2опк-2) В1 (ИД-2опк-2) 31 (ИД-1опк-3) У1 (ИД-1опк-3) В1 (ИД-1опк-3) 31 (ИД-1опк-7) У1 (ИД-1опк-7) В1 (ИД-1опк-7) 31 (ИД-2опк-7) У1 (ИД-2опк-7) В1 (ИД-2опк-7) 31 (ИД-2пк-2) В1 (ИД-2пк-2) У1 (ИД-2пк-2)
4	Основы тестирования и отладки программ	Ошибки программ: понятие и виды Понятие ошибки в про- грамме и их виды. Понятие тестирования и от- ладки программ. Тестирование и отладка программ. Способы тести- рования и отладки про- граммного обеспечения.	В1 (ИД-2опк-2) В1 (ИД-1опк-3) В1 (ИД-1опк-7) В1 (ИД-2опк-7) В1 (ИД-2пк-2)

5.2 Наименование тем лекций и их объем в часах с указанием рассматриваемых вопросов и формы обучения

Таблица 5.2.1 – Наименование тем лекций и их объем в часах с указанием рассматриваемых вопросов (очная форма обучения)

№ п/п	№ раздела дисциплины	Тема лекции	Рассматриваемые вопросы	Время, ч
1	2	3	4	5
1	1	Понятие алгоритма и его свойства. Способы записи алгоритмов	Понятие алгоритма и его свойства. Способы записи алгоритмов.	2
2	1	Основные алгоритмические конструкции	Алгоритмы линейного, разветвляющегося и циклического процесса.	2
3	2	Программы и программное обеспечение. Методология разработки программных продуктов	Понятие программы и программного обеспечения. Парадигмы программирования. Жизненный цикл программы.	4
4	2	Языки программирования	Языки программирования и их виды. Трансляторы и их виды. Системы программирования.	4
5	2	Структура программы и основные элементы языка Python	Элементы языка Python: алфавит, служебные слова, константы, переменные, имена, типы данных. Встроенные функции языка. Понятие выражений. Типы выражений. Структура программы на Python. Правила формирования и вычисления выражений.	6
6	2	Операторы языка Python	Виды операторов языка Python: простые, сложные (структурные), составные. Элементарный ввод и вывод на Python. Операторы присваивания, условного и безусловного переходов. Операторы условия. Оператор выбора. Операторы цикла.	8
7	2	Типы данных языка Python	Концепция типа данных. Простые (целые, вещественные, строковые, логические) и структурированные (массив, множество, запись, файл) типы данных.	6
Итого (2 семестр)				32

1	2	3	4	5
8	3	Алгоритмы поиска данных	Виды поиска, поисковый запрос. Поиск в неупорядоченном и в упорядоченном множестве данных. Оценка эффективности поиска.	4
9	3	Алгоритмы упорядочивания данных	Виды сортировки (внутренняя и внешняя). Основные методы упорядочивания данных. Оценка эффективности сортировки.	4
10	3	Алгоритмы обработки одномерных массивов	Объявление массива. Инициализация. Ввод и вывод одномерных массивов. Стандартные процедуры обработки массива. Удаление и вставка элементов в массив.	8
11	3	Алгоритмы обработки двумерных массивов	Объявление массива. Инициализация. Ввод и вывод двумерных массивов. Стандартные процедуры обработки массива и элементов массива по условию.	12
12	4	Ошибки программ: понятие и виды	Понятие ошибки в программе и их виды.	2
13	4	Понятие тестирования и отладки программ.	Тестирование и отладка программ. Способы тестирования и отладки программного обеспечения.	2
Итого (3 семестр)				32

5.3 Наименование тем лабораторных работ, их объем в часах и содержание (очная форма обучения)

Таблица 5.3.1 – Наименование тем лабораторных работ, их объем в часах и содержание (очная форма обучения)

№ п/п	№ раздела дисциплины	Тема работы	Время, ч
1	2	3	4
1	1	Понятие алгоритма и его свойства. Способы за- писи алгоритмов. Основные алгоритмические конструкции.	4
2	2	Языки программирования. Структура программы и основные элементы языка Python.	8
3	2	Операторы языка Python.	10
4	2	Типы данных языка Python.	10
Итого (2 семестр)			32
5	3	Алгоритмы поиска данных. Алгоритмы упорядо- чивания данных.	8
6	3	Алгоритмы обработки одномерных массивов.	10
7	3	Алгоритмы обработки двумерных массивов.	12
8	4	Понятие тестирования и отладки программ.	2
Итого (3 семестр)			32

5.4 Наименование тем практических занятий, их объем в часах и содер- жание

*Таблица 5.4.1 –Наименование тем практических занятий, их объем
и содержание (очная форма обучения)*

№ п/п	№ раздела дисципли- ны	Тема занятия	Время, ч.
1	2	3	4
1	3	<i>Массивы. Алгоритмы обработки массивов. Организа- ция массивов на Python.</i> Использование списков для ввода и вывода массивов. Использование списков для обработки массивов. Практические задания.	6

2	3	<i>Многомерные массивы.</i> Создание массивов numpy. Некоторые свойства массивов numpy. Генерация массивов случайных чисел. Некоторые методы многомерных массивов. Практические задания.	8
3	3	<i>Подпрограммы (функции пользователя) Python.</i> Понятие подпрограммы. Применение функций. Локальные и глобальные переменные. Практические задания.	8
Итого			22

5.5 Распределение трудоёмкости самостоятельной работы по видам работ (с указанием формы обучения)

*Таблица 5.5.1 – Распределение трудоёмкости самостоятельной работы
по видам работ (очная форма обучения 2 семестр)*

№	Вид работы	Время, ч
1	2	3
1	Подготовка к выполнению лабораторных работ и их защите	10,2
2	Выполнение задачи (практическое задание)	10
3	Изучение отдельных тем и вопросов (табл. 6.1.1)	22
Итого		42,2

Таблица 5.5.2 – Распределение трудоёмкости самостоятельной работы по видам работ (очная форма обучения 3 семестр)

№	Вид работы	Время, ч
1	2	3
1	Подготовка к выполнению лабораторных работ и их защите	20,4
2	Самостоятельное изучение материала по темам дисциплины и выполнение заданий	22
3	Выполнение курсовой работы	35,25
Итого		77,65

6 ПЕРЕЧЕНЬ УЧЕБНО-МЕТОДИЧЕСКОГО ОБЕСПЕЧЕНИЯ ДЛЯ САМОСТОЯТЕЛЬНОЙ РАБОТЫ ОБУЧАЮЩИХСЯ ПО ДИСЦИПЛИНЕ

Перечень учебно-методического обеспечения для самостоятельной работы обучающегося приведены в таблице 6.1.1.

Таблица 6.1.1– Тема, задания, вопросы и перечень учебно-методического обеспечения для самостоятельного изучения (очная форма обучения)

№ п/п	№ раздела дисциплины	Тема, вопросы, задание, планируемые результаты обучения	Время, ч	Рекомендуемая литература
1	1, 2	Алгоритмы и структуры данных Алгоритмы. Определение, основные свойства. Временная сложность алгоритмов: время выполнения в худшем случае, в среднем, в лучшем случае. Асимптотическая нотация: верхние оценки временной сложности, точные оценки, нижние оценки. Классификация алгоритмов по временной сложности. Структуры данных. Классификация. Элементарные данные: данные числовых типов, данные символьного типа, данные логического типа. Массивы. Множества. Комбинированный тип – записи. Файловый тип. Ссылочный тип. Переменные с указателями. Линейные структуры данных. Цепочки, очередь, стек, дек, линейные списки. Нелинейные структуры данных. Деревья. Характеристика древовидной структуры данных. Операции над сбалансированными деревьями. Общие операции над деревьями. <i>Осваиваемые компетенции: ОПК-2, ОПК-3, ОПК-7</i>	14	См табл.9.1.1, 9.1.2 раздела 9 Учебно-методическое и информационное обеспечение дисциплины
2	3	Алгоритмы сортировки Внутренняя и внешняя сортировки. Устойчивость метода сортировки. Сортировка массивов. Методы сортировки массивов. Сортировка с помощью прямого выбора. Сортировка с помощью прямой вставки. Сортировка с помощью прямого обмена. Улучшенные методы сортировки. Сортировка Шелла (включений с уменьшающимися расстояниями). Быстрая сортировка. Сравнение методов сортировки массивов. <i>Осваиваемые компетенции: ОПК-2, ОПК-3, ОПК-7, ПК-2</i>	14	См табл.9.1.1, 9.1.2 раздела 9 Учебно-методическое и информационное обеспечение дисциплины

3	3	Алгоритмы поиска Классификация основных методов поиска. Последовательный и индекснопоследовательный поиск. Бинарный поиск. Эффективность указанных методов поиска. Алгоритмы оптимизации поиска. Вероятностный подход. Алгоритмы перестановки в начало и транспозиции. Поиск по дереву. Вставка в дерево бинарного поиска. Удаление из дерева бинарного поиска. Эффективность алгоритма поиска по бинарному дереву. <i>Осваиваемые компетенции: ОПК-2, ОПК-3, ОПК-7, ПК-2</i>	16	См табл.9.1.1, 9.1.2 раздела 9 Учебно-методическое и информационное обеспечение дисциплины
Итого			44	

7 ОБРАЗОВАТЕЛЬНЫЕ ТЕХНОЛОГИИ

Для реализации предусмотренных видов учебной работы в качестве образовательных технологий в преподавании дисциплины «Алгоритмизация и программирование» используются традиционная и модульно-компетентностная технологии.

Таблица 7.1.1 – Образовательные технологии, обеспечивающие развитие у обучающихся навыков командной работы, межличностной коммуникации, принятия решений, лидерских качеств (очная форма обучения)

№раз дела	Вид за- нятия (лек, пр, лаб)	Используемые технологии и рассматриваемые вопросы, планируемые результаты обу- чения	Время, ч
1	2	3	4
1	лаб	<i>Технологии:</i> работа в малых группах, дискуссия, тестирование. <i>Вопросы:</i> понятие алгоритма и его свойства, способы записи алгоритмов, основные алгоритмические конструкции. 31 (ИД-1ОПК-3), В1 (ИД-1ОПК-3), У1 (ИД-1ОПК-3)	4
2	лаб	<i>Технологии:</i> работа в малых группах, дискуссия, тестирование. <i>Вопросы:</i> языки программирования, структура программы и основные элементы языка Python. 31 (ИД-2ОПК-2), В1 (ИД-2ОПК-2), У1 (ИД-2ОПК-2) 31 (ИД-2ПК-2), В1 (ИД-2ПК-2), У1 (ИД-2ПК-2)	8
2	лаб	<i>Технологии:</i> работа в малых группах, дискуссия. <i>Вопросы:</i> операторы языка Python. 31 (ИД-1ОПК-7), В1 (ИД-1ОПК-7), У1 (ИД-1ОПК-7)	10
2	лаб	<i>Технологии:</i> работа в малых группах, дискуссия. <i>Вопросы:</i> Типы данных языка Python. 31 (ИД-1ОПК-3), В1 (ИД-1ОПК-3), У1 (ИД-1ОПК-3)	10
3	лаб	<i>Технологии:</i> работа в малых группах, дискуссия. <i>Вопросы:</i> алгоритмы поиска данных, алгоритмы упорядочивания данных. 31 (ИД-1ОПК-7), В1 (ИД-1ОПК-7), У1 (ИД-1ОПК-7)	8
3	лаб	<i>Технологии:</i> работа в малых группах, дискуссия. <i>Вопросы:</i> алгоритмы обработки одномерных массивов. 31 (ИД-1ОПК-3), В1 (ИД-1ОПК-3), У1 (ИД-1ОПК-3) 31 (ИД-1ОПК-7), В1 (ИД-1ОПК-7), У1 (ИД-1ОПК-7)	10
3	лаб	<i>Технологии:</i> работа в малых группах, дискуссия. <i>Вопросы:</i> алгоритмы обработки двумерных массивов. 31 (ИД-1ОПК-3), В1 (ИД-1ОПК-3), У1 (ИД-1ОПК-3) 31 (ИД-1ОПК-7), В1 (ИД-1ОПК-7), У1 (ИД-1ОПК-7)	12
4	лаб	<i>Технологии:</i> работа в малых группах, дискуссия. <i>Вопросы:</i> понятие тестирования и отладки программ. 31 (ИД-2ОПК-7), В1 (ИД-2ОПК-7), У1 (ИД-2ОПК-7) 31 (ИД-2ПК-2), В1 (ИД-2ПК-2), У1 (ИД-2ПК-2)	2
Итого			64

8 ОЦЕНОЧНЫЕ МАТЕРИАЛЫ ПО ДИСЦИПЛИНЕ «АЛГОРИТМИЗАЦИЯ И ПРОГРАММИРОВАНИЕ»

Оценочные материалы для проведения текущего контроля успеваемости и промежуточной аттестации обучающихся по дисциплине, в т.ч. в форме заданий тестового типа, представлены в Приложении.

Задания тестового типа могут быть использованы при проведении диагностических процедур, в т.ч. диагностической работы, в рамках НОКО.

9 УЧЕБНО-МЕТОДИЧЕСКОЕ И ИНФОРМАЦИОННОЕ ОБЕСПЕЧЕНИЕ ДИСЦИПЛИНЫ

9.1 Перечень основной и дополнительной учебной литературы, ресурсов информационно-телекоммуникационной сети «Интернет» необходимых для освоения дисциплины «Алгоритмизация и программирование»

Таблица 9.1.1– Основная литература

(редакция от 31.08.2026, вводится в действие с 01.09.2026)

№ п/п	Наименование	Количество, экз.	
		Всего	В расчете на 100 обучаю- щихся
1	<i>Трофимов, В. В.</i> Алгоритмизация и программирование : учебник для вузов / В. В. Трофимов, Т. А. Макаrchук, Т. А. Павловская. — 5-е изд. — Москва : Издательство Юрайт, 2026. — 121 с. — (Высшее образование). — ISBN 978-5-534-21895-4. — Текст : электронный // Образовательная платформа Юрайт [сайт]. — URL: https://urait.ru/bcode/582397		

Таблица 9.1.2– Дополнительная литература

(редакция от 31.08.2026, вводится в действие с 01.09.2026)

№ п/п	Наименование	Количество, экз.	
		Всего	В расчете на 100 обучаю- щихся
2	<i>Федоров, Д. Ю.</i> Программирование на Python : учебник для вузов / Д. Ю. Федоров. — 6-е изд., перераб. и доп. — Москва : Издательство Юрайт, 2026. — 166 с. — (Высшее образование). — ISBN 978-5-534-22181-7. — Текст : электронный // Образовательная платформа Юрайт [сайт]. — URL: https://urait.ru/bcode/600874		
3	<i>Чернышев, С. А.</i> Основы программирования на Python : учебник для вузов / С. А. Чернышев. — 2-е изд., перераб. и доп. — Москва : Издательство Юрайт, 2026. — 349 с. — (Высшее образование). — ISBN 978-5-534-17139-6. — Текст : электронный // Образовательная платформа Юрайт [сайт]. — URL: https://urait.ru/bcode/588667		

Таблица 9.1.3 – Собственные методические издания кафедры по дисциплине

№ п/п	Наименование	Количество, экз.	
		всего	в расчете на 100 обучаю- щихся

Таблица 9.1.4 – перечень ресурсов информационно-телекоммуникационной сети «Интернет»

№ п/п	Наименование	Условия до- ступа
1	Федеральный центр информационно-образовательный ресурсов // электронный ресурс / http://fcior.edu.ru/	свободный
2	Единое окно доступа к образовательным ресурсам // электронный ресурс / http://window.edu.ru /	свободный
3	Электронно-библиотечная система издательства «лань» // элек- тронный ресурс http://e.lanbook.com/	по договору
4	Информационно-коммуникационные технологии в образовании // электронный ресурс / http://ict.edu.ru/	свободный
5	Электронная библиотека книг «bukoteka.ru» // электронный ресурс / http://bukoteka.ru/	свободный

9.2. Перечень информационных технологий, используемых при осуществлении образовательного процесса по дисциплине «Алгоритмизация и программирование», включая перечень программного обеспечения и информационных справочных систем

Таблица 9.2.1 – Перечень информационных технологий (перечень современных профессиональных баз данных и информационных справочных систем, используемых при осуществлении образовательного процесса по дисциплине «Алгоритмизация и программирование»

(редакция от 31.08.2026, вводится в действие с 01.09.2026)

№ П/п	Наименование	Условия доступа
1	2	3
1	Электронная библиотека полнотекстовых документов Пензенского ГАУ	https://pgau.ru/strukturnye-podrazdeleniya/nauchnaya-biblioteka/elektronnaya-biblioteka-pgau.html (доступ с любого компьютера локальной сети университета по IP-адресам; с личных ПК, мобильных устройств по коллективному или индивидуальному аутентификатору (логин/пароль), через личный кабинет; возможность регистрации для удаленной работы по IP) Помещения для самостоятельной работы: ауд. №5202, №1237
2	Образовательная платформа Юрайт. Для вузов и ссузов	https://urait.ru/ (доступ с любого компьютера локальной сети университета по IP-адресам; с личных ПК, мобильных устройств по индивидуальному аутентификатору (логин/пароль), через личный кабинет) Помещения для самостоятельной работы: ауд. №5202, №1237
3	Электронно-библиотечная система «Национальный цифровой ресурс «Рукопт»	https://lib.rucont.ru/search (доступ с любого компьютера локальной сети университета по IP-адресам; с личных ПК, мобильных устройств по коллективному или индивидуальному аутентификатору (логин / пароль); возможность регистрации для удаленной работы по IP) Помещения для самостоятельной работы: ауд. №5202, №1237
4	Электронно-библиотечная система издательства «Лань»	https://e.lanbook.com/ (доступ с любого компьютера локальной сети университета по IP-адресам; с личных ПК, мобильных устройств через личный кабинет по индивидуальному аутентификатору (логин/пароль); возможность удаленной регистрации и работы) Помещения для самостоятельной работы: ауд. №5202, №1237
5	Elibrary.ru – научная электронная библиотека	https://elibrary.ru/defaultx.asp (доступны поиск, просмотр и загрузка полнотекстовых лицензионных материалов через интернет (в том числе по электронной почте) по IP адресам университета без ограничения коли-

		чества пользователей; неограниченный доступ с личных компьютеров для библиографического поиска, просмотра оглавления журналов) Помещения для самостоятельной работы: ауд. №5202, №1237
6	Научная электронная библиотека «КИБЕРЛЕНИНКА»	https://cyberleninka.ru/ (доступ свободный) Помещения для самостоятельной работы: ауд. №5202, №1237
7	Центр цифровой трансформации в сфере АПК	https://cctmcx.ru/ (доступ свободный) Помещения для самостоятельной работы: ауд. №5202, №1237
8	Национальная платформа открытого образования -	https://npoed.ru/ (доступ свободный) Помещения для самостоятельной работы: ауд. №5202, №1237
9	Справочно-правовая система «КОНСУЛЬТАНТ+»	https://www.consultant.ru/ (в залах университета (ауд. 1237, 5202) без пароля)

10 МАТЕРИАЛЬНО-ТЕХНИЧЕСКАЯ БАЗА, НЕОБХОДИМАЯ ДЛЯ ОСУЩЕСТВЛЕНИЯ ОБРАЗОВАТЕЛЬНОГО ПРОЦЕССА ПО ДИСЦИПЛИНЕ

Таблица 10.1 – Материально-техническое обеспечение дисциплины
«Алгоритмизация и программирование»

(редакция от 31.08.2026, вводится в действие с 01.09.2026)

№ п/п	Наименование учебных аудиторий и помещений для самостоятельной работы	Перечень оборудования и технических средств обучения, наличие возможности подключения к сети «Интернет. Перечень лицензионного и свободно распространяе- мого программного обеспечения, в т.ч. оте- чественного производства. Реквизиты под- тверждающего документа
1	2	3
1	Учебная аудитория для проведения учебных занятий, 440014, Пензенская область, г. Пенза, ул. Ботаническая, д. 30; аудитория 1228	https://pgau.ru/sveden/objects/pracvo2p https://pgau.ru/sveden/objects/cabvo2p
2	Учебная аудитория для проведения учебных занятий, 440014, Пензенская область, г. Пенза, ул. Ботаническая, д. 30; аудитория 1102	https://pgau.ru/sveden/objects/pracvo2p https://pgau.ru/sveden/objects/cabvo2p
3	Учебная аудитория для проведения учебных занятий, 440014, Пензенская область, г. Пенза, ул. Ботаническая, д. 30; аудитория 1107	https://pgau.ru/sveden/objects/pracvo2p https://pgau.ru/sveden/objects/cabvo2p
4	Помещение для самостоятельной работы 440014, Пензенская область, г. Пенза, ул. Ботаническая, д. 30; аудитория 1237	https://pgau.ru/sveden/objects/pracvo2p https://pgau.ru/sveden/objects/cabvo2p
5	Помещение для самостоятельной работы 440014, Пензенская область, г. Пенза, ул. Ботаническая, д. 30; аудитория 5202	https://pgau.ru/sveden/objects/pracvo2p https://pgau.ru/sveden/objects/cabvo2p

11 МЕТОДИЧЕСКИЕ УКАЗАНИЯ ДЛЯ ОБУЧАЮЩЕГОСЯ ПО ОСВОЕНИЮ ДИСЦИПЛИНЫ

11.1 Методические советы по планированию и организации времени, необходимого для изучения дисциплины

Планирование времени на самостоятельную работу, необходимого на изучение настоящей дисциплины, студентам лучше всего осуществлять на весь семестр, предусматривая при этом регулярное повторение пройденного материала. Материал, изученный на лекциях, необходимо регулярно дополнять сведениями из литературных источников, представленных в рабочей программе. По каждой из тем для самостоятельного изучения, приведенных в рабочей программе дисциплины следует сначала изучить рекомендованную литературу. При необходимости следует составить краткий конспект основных положений, терминов, сведений, требующих запоминания и являющихся основополагающими в этой теме и для освоения последующих тем курса.

Регулярно отводить время для повторения пройденного материала, проверяя свои знания, умения и навыки по контрольным вопросам.

Рабочей программой дисциплины предусмотрена самостоятельная работа. Самостоятельная работа проводится с целью углубления знаний по дисциплине и предусматривает:

- изучение рекомендованной литературы и усвоение теоретического материала дисциплины;
- выполнение самостоятельных работ;
- подготовку к сдаче экзамена.

Для расширения знаний по дисциплине проводить поиск в различных системах и использовать материалы сайтов, рекомендованных преподавателем на лекциях и практических занятиях.

11.2 Методические рекомендации по использованию материалов рабочей программы

Рабочая программа представляет собой целостную систему, направленную на эффективное усвоение дисциплины в виду современных требований высшего образования. Структура и содержание РП позволяет сформировать необходимые общекультурные и профессиональные компетенции, предъявляемые к бакалавру техники технологии для успешного решения инженерных задач в своей практической деятельности.

При использовании РП необходимо ознакомиться со структурой и содержанием РП. Материалы, входящие в РП позволяют студенту иметь полное представление об объеме и предъявляемых требованиях к изучению дисциплины.

11.3 Методические советы по подготовке к промежуточной аттестации

При подготовке к промежуточной аттестации необходимо проработать лекции, имеющиеся учебно-методические материалы и другую рекомендованную литературу. Если самостоятельно не удалось разобраться в материале, сформулируйте вопросы и обратитесь за помощью к преподавателю на консультации.

Для самоконтроля необходимо ответить на имеющиеся тесты и вопросы к экзамену.

11.4 Методические советы по работе с тестовым материалом дисциплины

При работе над тестовыми заданиями необходимо ответить на тестовые вопросы и свериться с правильными ответами.

В случае недостаточности знаний, по какой-либо теме, необходимо проработать лекционный материал по этой теме, а также рекомендованную литературу.

Если по некоторым вопросам возникли затруднения, следует их законспектировать и обратиться к преподавателю на консультации за разъяснением.

Для наглядности выводов и обобщений можно привести графики, диаграммы и схемы.

12 СЛОВАРЬ ТЕРМИНОВ

Алгоритм — это точное предписание, определяющее вычислительный процесс, ведущий от входных данных к требуемому результату.

Алгоритмизация — описание очередности выполнения различных операций, необходимых для решения той или иной задачи в форме алгоритма.

Алгоритмические языки — это специальное средство, предназначенное для записи алгоритмов в аналитическом виде.

Блок-схема алгоритма — это такое графическое представление алгоритма, когда отдельные действия (или команды) представляются в виде геометрических фигур — блоков. Внутри блоков указывается информация о действиях, подлежащих выполнению. Связь между блоками изображают с помощью линий, называемых линиями связи, обозначающих передачу управления.

Глобальные переменные — переменные, объявленные вне функций.

Индекс — порядковый номер элемента.

Итерация — циклическая управляющая структура, которая содержит композицию и ветвление. Она предназначена для организации повторяющихся процессов обработки последовательности значений данных.

Комментарий — это пояснительный текст, который можно записать в любом месте программы, где разрешен пробел. Текст комментария может содержать любые комбинации латинских и русских букв, цифр и других символов языка.

Линейный алгоритм — это алгоритм, в котором блоки выполняются последовательно сверху вниз от начала до конца.

Массив — упорядоченная структура, предназначенная для хранения однотипных данных.

Одномерный массив — это последовательность ячеек, расположенных в одну линию.

Оператор присваивания — это основной оператор любого языка программирования, позволяющий поместить определенное значение в необходимую переменную.

Процедура — именованная последовательность инструкций, реализующая некоторое действие.

Псевдокод — система обозначений и правил, предназначенная для единообразной записи алгоритмов. Занимает промежуточное место между естественным и формальным языками.

Рекурсия – это способ организации процесса вычисления, когда алгоритм обращается сам к себе.

Сортировка – процесс расположения элементов массива в порядке убывания (возрастания) из начальных значений.

Строка – это последовательность символов кодовой таблицы ЭВМ.

Условие – вопрос, имеющий два варианта ответа: да или нет.

Цикл – многократно повторяемые участки вычислительного процесса.

ПРИЛОЖЕНИЕ

МИНИСТЕРСТВО СЕЛЬСКОГО ХОЗЯЙСТВА
РОССИЙСКОЙ ФЕДЕРАЦИИ
Федеральное государственное бюджетное образовательное
учреждение высшего образования
«Пензенский государственный аграрный
университет»

ОЦЕНОЧНЫЕ МАТЕРИАЛЫ ПО ДИСЦИПЛИНЕ

АЛГОРИТМИЗАЦИЯ И ПРОГРАММИРОВАНИЕ

Направление подготовки
09.03.03 ПРИКЛАДНАЯ ИНФОРМАТИКА

Направленность (профиль) программы
Прикладная информатика в экономике

Квалификация
«Бакалавр»

Форма обучения – очная

Пенза – 2023

1 Перечень компетенций с указанием этапов их формирования и индикаторов достижения

Таблица 1.1 – Перечень компетенций с указанием этапов их формирования и индикаторов достижения по дисциплине «Алгоритмизация и программирование»

№ пп	Код и наименование компетенции	Код Индикатора Достижения компетенции	Наименование индикатора достижения компетенции	Код планируемого результата обучения	Этапы формирования компетенции
1	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	ИД-2 _{ОПК-2}	Использует современные информационные технологии и программные средства при решении профессиональных задач, в том числе отечественного производства	31 (ИД-2 _{ОПК-2})	Знать: программные средства, в том числе отечественного производства при решении задач профессиональной деятельности
2				У1 (ИД-2 _{ОПК-2})	Уметь: выбирать современные программные средства, в том числе отечественного производства при решении задач профессиональной деятельности
3				В1 (ИД-2 _{ОПК-2})	Владеть: навыками применения современных программных средств, в том числе отечественного производства, при решении задач профессиональной деятельности.
4	ОПК-3 - способен решать стандартные задачи профессиональной деятельности на основе информационной и библиографической культуры с применением информационно-коммуникационных технологий и с учетом основных требований ин-	ИД-1 _{ОПК-3}	Способен применять информационно-коммуникационные технологии для решения стандартных задач профессиональной деятельности на основе информационной и библиографической культуры	31 (ИД-1 _{ОПК-3})	Знать: основные структуры данных и методы их обработки; различия между языками программирования высокого и низкого уровня
5				У1 (ИД-1 _{ОПК-3})	Уметь: программировать алгоритмы, используя средства языков высокого уровня
6				В1 (ИД-1 _{ОПК-3})	Владеть навыками в проведении анализа получаемых результатов и оформлении документации на

	формационной безопасности.				программу
7	ОПК-7 - способен разрабатывать алгоритмы и программы, пригодные для практического применения.	ИД-1 _{ОПК-7}	Применяет языки программирования и работу с базами данных для автоматизации бизнес-процессов, реализацию прикладных задач различных классов, ведение баз данных и информационных хранилищ	31 (ИД-1 _{ОПК-7})	Знать: процесс подготовки и решения задач на ПЭВМ, основы алгоритмизации, приемы программирования на языке высокого уровня
8				У1 (ИД-1 _{ОПК-7})	Уметь: разрабатывать алгоритмы решения задач, выполнять разработку, тестирование и отладку программ на алгоритмических языках высокого уровня
9				В1 (ИД-1 _{ОПК-7})	Владеть навыками разработки прикладных задач.
10		ИД-2 _{ОПК-7}	Осуществляет программирование, отладку и тестирование прототипов программно-технических комплексов задач	31 (ИД-2 _{ОПК-7})	Знать: современные информационные технологии и программные средства, в том числе отечественного производства при решении задач профессиональной деятельности.
11				У1 (ИД-2 _{ОПК-7})	Уметь: выбирать современные информационные технологии и программные средства, в том числе отечественного производства при решении задач профессиональной деятельности
12				В1 (ИД-2 _{ОПК-7})	Владеть навыками применения современных информационных технологий и программных средств, в том числе отечественного производства, при решении задач профессиональной деятельности
13	ПК-2 - способен разрабатывать и адаптировать прикладное про-	ИД-2 _{ПК-2}	Осуществляет организационное и технологическое обеспечение кодирования на языках про-	31 (ИД-2 _{ПК-2})	Знать: - регламенты кодирования на языках программирования
14				У1 (ИД-2 _{ПК-2})	Уметь: выбирать со-

	граммное обеспечение.		граммирования		временные информационные технологии и программные средства, в том числе отечественного производства при решении задач профессиональной деятельности
15				В1 (ИД-2 ПК-2)	Владеть навыками обеспечения соответствия разработанного кода и процесса кодирования на языках программирования принятым в организации или проекте стандартам и технологиям

2 Оценочные материалы по дисциплине «Алгоритмизация и программирование»

2.1 Оценочные материалы тестового типа

Таблица 2.1 - Задания тестового типа

№ п/п	Текст задания	Варианты ответов	Правильный ответ	Код компетенции	Семестр
1. Задание закрытого типа на установление соответствия					
Инструкция (сценарий выполнения):					
1. Внимательно прочитать текст задания и понять, что в качестве ответа ожидаются пары элементов.					
2. Внимательно прочитать оба списка: список 1 — вопросы, утверждения, факты, понятия и т.д.; список 2 — утверждения, свойства объектов и т.д.					
3. Сопоставить элементы списка 1 с элементами списка 2, сформировать пары элементов.					
4. Записать попарно буквы и цифры (в зависимости от задания) вариантов ответа (например, А1 или Б4)					
1.	Установите соответствие типов алгоритмов с их характеристиками А. Алгоритм поиска Б. Алгоритм сортировки В. Алгоритм вычисления Г. Алгоритм обработки данных	1. упорядочивает элементы в определенном порядке (например, по возрастанию) 2. находит определенный элемент в массиве или списке 3. выполняет математические операции, такие как сложение, умножение и вычисление факториала 4. обрабатывает и преобразует данные (например, фильтрация или агрегирование)	А2Б1В3Г4	ОПК-2	2
2.	Установите соответствие типов алгоритмов с их характеристиками А. Линейный алгоритм Б. Разветвляющийся алгоритм В. Циклический алгоритм Г. Рекурсивный алгоритм	1. выполняется последовательно, шаг за шагом 2. выполняет одни и те же действия несколько раз до тех пор, пока не будет выполнено определенное условие 3. вызывает сам себя для решения подзадачи 4. принимает решение на основе условий, выполняя разные действия в зависимости от результата	А1Б4В2Г3	ОПК-2	2

3.	Установите соответствие типов алгоритмов с их характеристиками А. Детерминированный алгоритм Б. Непредсказуемый алгоритм В. Стохастический алгоритм Г. Эвристический алгоритм	1. всегда дает один и тот же результат при одинаковых входных данных 2. использует случайные элементы для принятия решений 3. находит решение, основываясь на приближенных методах 4. может давать разные результаты в зависимости от условий и случайных факторов	A1Б4В2Г3	ОПК-2	2
4.	Установите соответствие алгоритмов по области применения с их характеристиками А. Алгоритмы оптимизации Б. Алгоритмы шифрования В. Алгоритмы машинного обучения Г. Алгоритм с обратной связью	1. используются для нахождения наилучшего решения среди множества вариантов 2. обучаются на данных для предсказания или классификации 3. преобразуют данные для защиты информации 4. корректирует свои действия на основе полученной информации	A1Б3В2Г4	ОПК-2	2
5.	Установите соответствие между операциями и их обозначениями в языке Python А. Логическое И, логическое умножение Б. Логическое ИЛИ, логическое сложение В. Операция "не равно" обозначается Г. Отрицание, логическое НЕ	1. not 2. and 3. or 4. !=	A2Б3В4Г1	ОПК-3	2
6.	Установите соответствие математических операций языка Python А. // Б. % В. * Г. /	1. умножение 2. деление 3. целочисленное деление 4. остаток от деления	A3Б4В1Г2	ОПК-3	2
7.	Установите соответствие элементов языка Python и их характеристиками А. Служебные слова Б. Величины В. Переменные Г. Имя переменной (идентификатор)	1. любая отличная от служебных слов последовательность латинских букв, цифр и символа подчеркивания "_", не может начинаться с цифры 2. в программе представлены в виде констант и переменных 3. величины, которые могут изменять свое значение при выполнении программы и каждая из них имеет имя, тип и значение 4. цепочки символов, имеющие фиксированное смысловое значение	A4Б2В3Г1	ОПК-3	2

8.	Установите соответствие типов данных и их обозначений в языке Python А. Целочисленный Б. Вещественный В. Строковый Г. Логический	1. bool 2. str 3. int 4. float	A3B4B2Г1	ОПК-3	2
9.	Установите соответствие встроенных функций Python для работы со списками А. len(list) Б. sorted(list, [key]) В. min(list) Г. max(list)	1. возвращает длину списка 2. возвращает наибольший элемент списка 3. возвращает отсортированный список 4. возвращает наименьший элемент списка	A1Б3B4Г2	ОПК-7	3
10.	Установите соответствие между типами последовательностей и их определением А. List (список) Б. Tuple (кортеж) В. Set (множество) Г. Dict (словарь)	1. неупорядоченная изменяемая последовательность; одинаковые элементы удаляются 2. неупорядоченная изменяемая последовательность, состоящая из пар ключ, значение; ключи не дублируются 3. последовательность, которая упорядочена, но не изменяемая; допускаются одинаковые элементы 4. упорядоченная последовательность, которую можно изменять; допускаются одинаковые элементы	A4Б3B1Г2	ОПК-7	3
11.	Установите соответствие между функциями библиотеки math и их математической записью А. math.pow(x,a) Б. math.abs(x) В. math.exp(x) Г. math.sqrt(x)	1. e^x 2. $\sqrt{x}$ 3. x^a 4. $ x $	A3Б4B1Г2	ОПК-7	3
12.	Установите соответствие между алгоритмами и их описанием А. Находит наибольший общий делитель двух чисел Б. Сравнивает соседние элементы и меняет их местами для сортировки В. Находит кратчайший путь в графе между двумя вершинами Г. Ищет элемент в отсортированном массиве, деля его пополам на каждой итерации	1. алгоритм сортировки пузырьком 2. алгоритм Дейкстры 3. алгоритм Евклида 4. алгоритм бинарного поиска	A3Б1B2Г4	ОПК-7	3

13.	Установите соответствие встроенных функций Python для работы со словарями А. Удаление элемента словаря и возвращение значения по ключу Б. Удаление и возвращение случайного элемента словаря В. Удаление всех элементов из списка Г. Создание нового отсортированного словаря	1. .sorted() 2. .pop() 3. .popitem() 4. .clear()	A2Б3В4Г1	ПК-2	3
14.	Установите соответствие встроенных функций Python для работы со словарями А. Удаление элемента по ключу Б. Длина словаря В. Наименьшее значение элемента из словаря Г. Суммирование значений элементов, если они все числовые	1. len() 2. sum() 3. min() 4. del	A4Б1В3Г2	ПК-2	3
15.	Установите соответствие встроенных функций Python для работы со множествами А. Возвращает множество, являющееся объединением множеств А и В Б. Добавляет в множество А все элементы из множества В В. Возвращает множество, являющееся пересечением множеств А и В Г. Оставляет в множестве А только те элементы, которые есть в множестве В	1. A.union(B) 2. A.intersection(B) 3. A.intersection_update(B) 4. A.update(B)	A1Б4В2Г3	ПК-2	3
16.	Установите соответствие встроенных функций Python для работы со списками А. Добавляет элемент в конец последовательности Б. Считает количество элементов, равных х В. Добавляет к концу последовательности последовательность s Г. Меняет порядок элементов s на обратный	1. reverse(s) 2. extend(s) 3. count(x) 4. append(x)	A4Б3В2Г1	ПК-2	3

2. Задание закрытого типа на установление последовательности

Инструкция (сценарий выполнения):

1. Внимательно прочитать текст задания и понять, что в качестве ответа ожидается последовательность элементов.
2. Внимательно прочитать предложенные варианты ответа.
3. Построить верную последовательность из предложенных элементов.
4. Записать буквы/цифры (в зависимости от задания) вариантов ответа в нужной последовательности без пробелов и знаков препинания (например, БВА или 135)

1.	Установить правильную последовательность алгоритма решения квадратного уравнения	1. записать корни уравнения. 2. вычислить дискриминант 3. определить количество корней по значению дискриминанта 4. найти корни уравнения 5. записать уравнение в стандартной форме	52341	ОПК-2	2
2.	Установить правильную последовательность алгоритма вычисления среднего арифметического	1. подсчитать количество чисел 2. собрать данные (числа) 3. найти сумму всех чисел 4. записать результат 5. разделить сумму на количество чисел	23154	ОПК-2	2
3.	Установить правильную последовательность алгоритма вычисления факториала числа n	1. инициализировать переменную для хранения результата (например, result=1) 2. умножать result на каждое число от 2 до n 3. проверить, является ли n равным 0 или 1 (факториал равен 1) 4. записать значение result 5. определить число n	53124	ОПК-2	2
4.	Установить правильную последовательность алгоритма поиска наибольшего общего делителя (НОД)	1. повторять шаги 2-4, пока b не станет равным 0 2. в противном случае, вычислить a mod b 3. присвоить a значение b, a b — значение a mod b 4. определить два числа a и b 5. если b=0, то НОД равен a	45231	ОПК-2	2
5.	Установить правильную последовательность выполнения действий $\frac{-b + \sqrt{b^2 - 4ac}}{2a}$	1. -4*a*c)) 2. b**2 3. +sqrt(4. / 5. (2*a) 6. (-b	632145	ОПК-3	2

6.	Установить правильную последовательность этапов решения задач на компьютере	1. составление программы 2. отладка и тестирование программы 3. разработка алгоритма 4. постановка задачи 5. математическая формализация (выбор метода решения) 6. проведение расчетов и анализ полученных результатов	453126	ОПК-3	2
7.	Установить правильную последовательность выполнения команд e^{2*x+y}	1. +y) 2. 2* 3. exp(4. x	3241	ОПК-3	2
8.	Установить правильную последовательность условного оператора if	1. <блок_операторов_2> 2. if <условие>: 3. else: 4. <блок_операторов_1>	2431	ОПК-3	2
9.	Установить правильную последовательность при разработке алгоритма определения количества значений больших 5 в одномерном массиве из 25 чисел	1. выполним проверку в цикле i=0 to 24 если A[i]>5 то вычислить n5=n5+1 2. по завершении цикла - выполнить печать n5 3. имеется массив из 25 элементов, например A[0] - A[24] 4. создадим переменную-счетчик (значений больших «5») n5=0	3412	ОПК-7	3
10.	Установить правильную последовательность при разработке алгоритма вычисления суммы элементов массива A(M) и среднего значения	1. конец цикла 2. начало цикла параметр цикла I изменяется от 1 до M с шагом 1 вычисление суммы S=S+A(I) 3. задание начального значения переменной суммы S=0 4. вычисление среднего SR=S/M	3214	ОПК-7	3
11.	Установить правильную последовательность при разработке алгоритма вычисления произведения элементов массива A(M)	1. конец цикла 2. начало цикла параметр цикла I изменяется от 1 до M с шагом 1 3. задание начального значения переменной произведения P=1 4. вычисление произведения P=P*A(I)	3241	ОПК-7	3

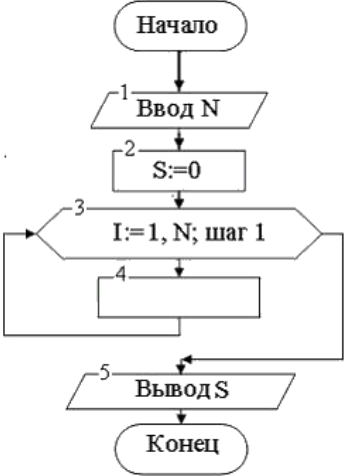
12.	Установить правильную последовательность при разработке алгоритма подсчета элементов массива A(M), удовлетворяющих условию A(I)>T	1. K=K+1 2. если A(I)>T то 3. начало цикла параметр цикла I изменяется от 1 до M с шагом 1 4. задание K=0 5. конец цикла	43215	ОПК-7	3
13.	Расположите в правильном порядке основные шаги при составлении программы	1. тестирование 2. проектирование 3. определение требований 4. внедрение и сопровождение 5. разработка и кодирование	32514	ПК-2	3
14.	Установить правильную последовательность при разработке алгоритма создания словаря	1. 'C++': 'БьёрнСтрауструп'} 2. languages = 3. = {'Python': 'Твидо ванРоссум'} 4. 'C#': 'Андерс Хейлсберг'	2341	ПК-2	3
15.	Расположите перечисленные действия в правильной последовательности, чтобы создать простое приложение на Python	1. написать код программы 2. сохранить файл с расширением .py 3. запустить приложение 4. открыть редактор кода	4123	ПК-2	3
16.	Расположите операции над множествами в порядке их приоритета (от высшего к низшему)	1. разность множеств 2. пересечение множеств 3. объединение множеств	321	ПК-2	3

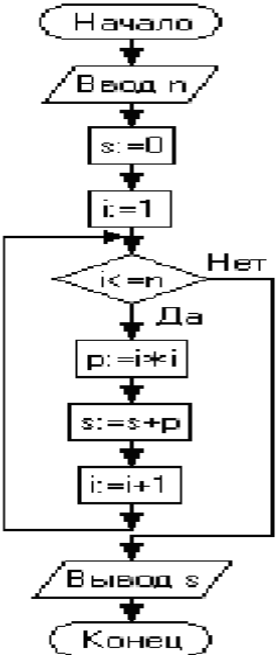
3. Задание открытого типа с развернутым ответом/ задача

Инструкция (сценарий выполнения):

1. Внимательно прочитать текст задания и понять суть вопроса.
2. Продумать логику и полноту ответа.
3. Записать ответ, используя четкие компактные формулировки.
4. В случае расчетной задачи записать решение и ответ

1.	Алгоритм задан блок-схемой <pre> graph TD Start([Начало]) --> Input[/Ввод K/] Input --> Init[S:=0] Init --> LoopStart{i:=2, K; шаг 2} LoopStart --> Sum[S:=S+i] Sum --> LoopStart Sum --> Output[/Вывод S/] Output --> End([Конец]) </pre> Описать результат выполнения алгоритма при вводе натурального числа K		сумма четных чисел среди первых K натуральных чисел	ОПК-2	2
----	---	--	---	-------	---

2.	Алгоритм вычисления $\sum_{I=1}^N \left(\frac{1}{I} \right)$ задан в виде блок-схемы:  <pre> graph TD Start([Начало]) --> Input[/Ввод N/] Input --> Init[S:=0] Init --> LoopStart[I:=1, N; шаг 1] LoopStart --> Block4[] Block4 --> LoopStart Block4 --> Output[/Вывод S/] Output --> End([Конец]) </pre> Записать команду, используемую в блоке 4		$S := S + 1/I$	ОПК-2	2
----	---	--	----------------	-------	---

3.	Дана схема алгоритма  <pre> graph TD Start([Начало]) --> Input[/Ввод n/] Input --> S0[s:=0] S0 --> I1[i:=1] I1 --> Loop{ i <= n } Loop -- Да --> P[i*i] P --> S[s+p] S --> I[i+1] I --> Loop Loop -- Нет --> Output[/Вывод s/] Output --> End([Конец]) </pre> В результате выполнения алгоритма при входном значении $n=6$ значение переменной s будет равно _____		Решение: $n=6, s=0$ открываем цикл с предусловием, внутри которого вычисляем p и s. 1. итерация $p=1*1=1$ $s=0+1=1$ 2. итерация $p=2*2=4$ $s=1+4=5$ 3. итерация $p=3*3=9$ $s=5+9=14$ 4. итерация $p=4*4=16$ $s=14+16=30$ 5. итерация $p=5*5=25$ $s=30+25=55$ 6. итерация $p=6*6=36$ $s=55+36=91$ Ответ: 91	ОПК-2	2
----	--	--	---	-------	---

4.	Дана блок-схема алгоритма <pre> graph TD Start([Н а ч а л о]) --> Input[/Ввод а/] Input --> Process[a = а * 4] Process --> Output[/Вывод а/] Output --> End([К о н е ц]) </pre> Определить значение а, которое будет получено, если начальное значение а=5		Решение: $a=5$ $a=5*4=20$ Ответ: 20	ОПК-2	2
5.	Переведите конструкцию языка <pre> S=0 While S<10: print(S) S=S+1 </pre>		Пока условие истинно, выполняется инструкция. Иначе выход из цикла.	ОПК-3	2
6.	Дайте характеристику переменной в Python		Под переменной в Python понимается имя/идентификатор, который может принимать некоторое значение.	ОПК-3	2

7.	Дан фрагмент программы на языке Python <pre>a=7 While a>5:a=a-1</pre> Определите значение переменной a, при котором произойдет завершение цикла		Решение: пока a>5, цикл работает, в противном случае цикл завершает свое действие. a=7-1=6 a=6-1=5 a=5-1=4 Ответ: 5	ОПК-3	2
8.	Дан фрагмент программы на языке Python <pre>Import math IF X>0: Y = X**2 ELSE: Y=math.sqrt(X);</pre> Запишите значение переменной Y, при X = 6		Решение: X = 6 Условие X>0 выполняется, следовательно X=X**2=6^2=36 Ответ: 36	ОПК-3	2
9.	Дан фрагмент программы на языке Python <pre>a=345 print (a%10)</pre> Запишите результат, который получится в результате выполнения оператора print		Решение: 345/10=34,5 Остаток от деления 5 Ответ: 5	ОПК-7	3
10.	Дан фрагмент программы на языке Python <pre>a=26 b=7 b=a//b c=a%b print (c)</pre> В ответ запишите значение c, полученное в результате выполнения программы		Решение: a=26 b=7 b=26//7=3 c=26%3=2 Ответ: 2	ОПК-7	3

11.	Дана программа: <pre>a, b, c = map (int, input().split()) s = a+b*c print(s)</pre> В ответ запишите значение s, полученное в результате выполнения программы, если с клавиатуры будут введены числа 4, 12, 2		Решение: <pre>a=4 b=12 c=2 s=4+12*2=28</pre> Ответ: 28	ОПК-7	3
12.	Дана программа: <pre>a, b, c = map (int, input().split()) s = a*b+c print(s)</pre> В ответ запишите значение s, полученное в результате выполнения программы, если с клавиатуры будут введены числа 4, 12, 2		Решение: <pre>a=4 b=12 c=2 s=4*12+2=50</pre> Ответ: 50	ОПК-7	3
13.	Дайте характеристику словарям в Python		Под словарями в Python понимается неупорядоченные коллекции произвольных объектов с доступом по ключу.	ПК-2	3
14.	Дайте характеристику множествам в Python		Под множествами в Python понимается структура данных, которые содержат неупорядоченные и неиндексированные элементы.	ПК-2	3
15.	Дан фрагмент программы на языке Python <pre>car = { "brand": "Форд", "model": "Мустанг", "year": 1964 }</pre> Запишите программную строку для значения ключа "year" с 1964 на 2025.		Решение: в словаре car необходимо изменить значение ключа "year" Ответ: <pre>car["year"]=2025</pre>	ПК-2	3

16.	Дайте характеристику понятию эмуляция в вычислительной технике		Под эмуляцией в вычислительной технике понимается процесс создания программного средства, которое имитирует другую программу.	ПК-2	3
4. Задания открытого типа с кратким ответом/ вставить термин, словосочетание....., дополнить предложенное					
Инструкция (сценарий выполнения):					
1. Внимательно прочитать текст задания и понять суть вопроса.					
2. Продумать логику и полноту ответа.					
3. Записать ответ в виде термина, словосочетания, дополнить предложенное					
1.	Человек, робот, собака, автомат, компьютер, который выполняет чьи-то команды - это _____		исполнитель	ОПК-2	2
2.	Свойство алгоритма, заключающиеся в том, что алгоритм должен состоять из конкретных действий, следующих в определенном порядке, называется. _____		дискретность	ОПК-2	2
3.	Свойство алгоритма, заключающиеся в том, что один и тот же алгоритм можно использовать с разными исходными данными, называется. _____		массовость	ОПК-2	2
4.	Какой тип алгоритмической структуры необходимо применить, если последовательность команд выполняется или не выполняется в зависимости от условия _____		ветвление	ОПК-2	2
5.	Программа Python называется _____		скрипт	ОПК-3	2
6.	_____ в Python это логический тип данных, встроенный в Python по умолчанию		Bool	ОПК-3	2

7.	Языки программирования, отражающие характеристики некоторой предметной области, позволяющие быстро и эффективно писать программы для решения задач определенного класса относятся к _____		проблемно-ориентированным	ОПК-3	2
8.	Необходимо написать оператор для вывода счетчика в фрагменте программы: <code>for i in range(n):</code> _____		<code>print (i)</code>	ОПК-3	2
9.	Совокупность символов, используемых в языке программирования, которая включает буквы различного регистра, цифры и дополнительные символы это: _____		алфавит	ОПК-7	3
10.	Программный код, который выполняется внутри цикла, называется _____		телом цикла	ОПК-7	3
11.	Процесс исследования программного кода на наличие ошибок с целью их исправления это _____		отладка	ОПК-7	3
12.	Процесс улучшения производительности программного обеспечения путём изменения его исходного кода это _____		оптимизация	ОПК-7	3
13.	_____ в Python - неупорядоченные коллекции произвольных объектов с доступом по ключу. Их иногда ещё называют ассоциативными массивами или хеш-таблицами		словари	ПК-2	3
14.	Процесс создания программного средства, которое имитирует другую программу _____		эмуляция	ПК-2	3
15.	Ключ словаря в Python позволяет однозначно _____ элементы словаря		идентифицировать	ПК-2	3
16.	_____ в Python – это структура данных, которые содержат неупорядоченные элементы. Элементы также не являются индексированными.		множества	ПК-2	3

5. Задания комбинированного типа с выбором одного/нескольких правильного ответа из предложенных с последующим объяснением своего выбора

Инструкция (сценарий выполнения): 1. Внимательно прочитать текст задания и понять суть вопроса. 2. Продумать логику и полноту ответа. 3. Записать номер правильного ответа или номера правильных ответов без пробелов и запятых (в зависимости от задания) и дать обоснование, используя четкие компактные формулировки.					
1.	Свойство алгоритма, означающее, что данный алгоритм применим к решению целого класса задач, называется	1. определенность 2. массовость 3. понятность 4. результативность	2 Обоснование: свойство массовость означает, что алгоритм применим к решению целого класса задач	ОПК-2	2
2.	Формульно-словесный способ записи алгоритма характеризуется тем, что описание осуществляется с помощью:	1. слов 2. специальных знаков 3. формул 4. символов	3 Обоснование: формулы используются для записи алгоритма формульно-словесным способом	ОПК-2	2
3.	Основными типами алгоритмов являются алгоритмы:	1. циклические 2. линейные 3. нелинейные 4. разветвляющиеся	124 Обоснование: в алгоритмизации основными типами алгоритмов являются: линейные, разветвляющиеся, циклические	ОПК-2	2
4.	Для создания искусственного интеллекта используется язык	1. Питон 2. Паскаль 3. Лисп 4. Ява	134 Обоснование: для создания ИИ используются языки программирования Питон, Лисп, Ява	ОПК-2	2

5.	К целочисленным типам данных в языке программирования Python относится	1. Integer 2. Byte 3. Str 4. Int	4 Обоснование: в Python к целочисленным типам данных относится int (integer)	ОПК-3	2
6.	Специальное средство, предназначенное для записи алгоритмов в аналитическом виде	1. алгоритмические языки 2. алгоритмические навыки 3. алгоритмические эксперименты 4. алгоритмические инструкции	1 Обоснование: алгоритмические языки — это специальное средство, предназначенное для записи алгоритмов в аналитическом виде	ОПК-3	2
7.	Язык программирования Python подходит для разработки	1. компьютерных и мобильных приложений 2. аналитика и машинное обучение 3. игр 4. аккумулятор	123 Обоснование: Python подходит для разработки компьютерных и мобильных приложений, игр, для аналитики данных и машинного обучения	ОПК-3	2
8.	Операционные системы на которых может работать Python	1. Windows 2. Linux 3. macOS 4. Yandex	123 Обоснование: Python может работать на операционных системах: Windows, Linux, macOS.	ОПК-3	2
9.	Создатель языка программирования Python	1. Гвидо Ван Россум 2. Дэвид Паттерсон 3. Эрвин Дональд Кнут 4. Джеймс Артур Гослинг	1 Обоснование: Гвидо ван Россум автор языка программирования Python.	ОПК-7	3

10.	Добавить модуль в программу, написанную на языке Python, можно с помощью команды	1. 2. import math() 3. import (math) 4.import .math	1 Обоснование: добавить модуль в программу, написанную на языке Python, можно с помощью команды import math	ОПК-7	3
11.	Выберите правильные имена переменных	1. D 2. D1 3. Dlina 4. длина	123 Обоснование: имя переменной может содержать только латинские буквы, числа и символ нижнего подчёркивания.	ОПК-7	3
12.	К правилам оформления блок-схем алгоритма относятся следующие правила	1. блок-схема всегда рисуется только по готовому программному коду 2. внутри геометрической фигуры указывается поясняющий текст или формула 3. блоки соединяются линиями связи, которые проходят строго горизонтально или вертикально 4. основным направлением линий считается сверху вниз и слева направо, в этом случае не требуется стрелка	234 Обоснование: к правилам оформления блок-схем алгоритма не относится утверждение о том, что блок-схема всегда рисуется только по готовому программному коду	ОПК-7	3
13.	Метод items() в Python возвращает	1. пары ключ/значение 2. ключи 3. типы ключей и значений 4. значения	1 Обоснование: для возвращения пар ключ/значение из существующего словаря используется метод items()	ПК-2	3
14.	Правило изменения ключей в Python-словаре	1. ключи меняются постоянно 2. не более 1 раза в момент инициализации 3. ключи неизменны 4. сколько угодно	3 Обоснование: ключи представляют собой неизменный объект	ПК-2	3

15.	Типы данных допустимых для ключей в словаре Python	1. число 2. строка 3. список 4. кортеж	124 Обоснование: Ключ словаря не может относиться к изменяемому типу данных: список.	ПК-2	3
16.	Основные методы словарей Python	1. pop() 2. del 3. from 4. len()	124 Обоснование: from не является методом словаря Python	ПК-2	3

2.2 Оценочные материалы для текущего контроля

Текущий контроль успеваемости является обязательной частью внутренней системы оценки качества освоения обучающимися образовательной программы. Текущий контроль успеваемости проводится в рамках изучения дисциплины в течение семестра. Виды оценочных материалов, формы контроля, темы и график определяется педагогическим работником.

2.3 Типовые вопросы для промежуточной аттестации

Вопросы для промежуточной аттестации (зачет) по оценке освоения компетенции ОПК-2 (2 семестр)

1. Понятие алгоритма и его свойства
2. Способы записи алгоритмов
3. Основные алгоритмические конструкции
4. Программы и программное обеспечение.
5. Методология разработки программных продуктов
6. Этапы жизненного цикла ПО
7. Модели жизненного цикла

Вопросы для промежуточной аттестации (зачет) по оценке освоения компетенции ОПК-3 (2 семестр)

8. Языки программирования и их классификация
9. Виды трансляторов
10. Структура программы на языке Python
11. Основные элементы языка Python
12. Операторы языка Python
13. Организация выбора и проверки условий в языке Python
14. Организация циклов в языке Python
15. Процедуры и функции
16. Простые типы данных языка Python
17. Структурированные типы данных языка Python
18. Понятие и виды поиска
19. Особенности алгоритмов поиска данных
20. Понятие и виды сортировки
21. Особенности алгоритмов упорядочивания данных
22. Особенности алгоритма нахождения суммы элементов массива, удовлетворяющих некоторому условию
23. Особенности алгоритма нахождения среднего арифметического элементов массива, удовлетворяющих некоторому условию

Вопросы для промежуточной аттестации (экзамен) по оценке освоения компетенции ОПК-2 (3 семестр)

1. Понятие алгоритма и его свойства
2. Способы записи алгоритмов
3. Основные алгоритмические конструкции

Вопросы для промежуточной аттестации (экзамен) по оценке освоения компетенции ОПК-3 (3 семестр)

4. Программы и программное обеспечение.
5. Методология разработки программных продуктов
6. Этапы жизненного цикла ПО
7. Модели жизненного цикла

Вопросы для промежуточной аттестации (экзамен) по оценке освоения

компетенции ОПК-7 (3 семестр)

8. Языки программирования и их классификация
9. Виды трансляторов
10. Структура программы на языке Python
11. Основные элементы языка Python
12. Операторы языка Python
13. Организация выбора и проверки условий в языке Python
14. Организация циклов в языке Python
15. Процедуры и функции
16. Простые типы данных языка Python
17. Структурированные типы данных языка Python
18. Понятие и виды поиска
19. Особенности алгоритмов поиска данных
20. Понятие и виды сортировки
21. Особенности алгоритмов упорядочивания данных
22. Особенности алгоритма нахождения суммы элементов массива, удовлетворяющих некоторому условию
23. Особенности алгоритма нахождения среднего арифметического элементов массива, удовлетворяющих некоторому условию
24. Особенности алгоритма нахождения индексов элементов массива, удовлетворяющих некоторому условию
25. Особенности алгоритма изменения значения элементов массива, удовлетворяющих некоторому условию
26. Особенности алгоритма вставки элемента массива
27. Особенности алгоритма удаления элемента массива
28. Особенности алгоритма определения индексов максимального элемента двумерного массива
29. Особенности алгоритма нахождения количества элементов двумерного массива, удовлетворяющих некоторому условию
30. Особенности алгоритма нахождения максимального элемента двумерного массива среди элементов, удовлетворяющих некоторому условию
31. Особенности алгоритма нахождения суммы элементов с заданными свойствами в некоторой строке массива
32. Особенности алгоритма нахождения индексов элементов массива с заданными свойствами
33. Особенности алгоритма определения максимального элемента в некоторой строке массива
34. Особенности алгоритма изменения значений элементов массива с заданными свойствами в некотором столбце массива
35. Особенности алгоритма нахождения суммы элементов с заданными свойствами в каждой строке массива
36. Особенности алгоритма определения номера строки массива с максимальной суммой значений
37. Особенности алгоритма определения индекса столбца для максимального элемента в каждой строке массива
38. Особенности алгоритма нахождения среднего арифметического значений элементов с заданными свойствами в каждой строке массива
39. Особенности алгоритма обмена местами двух элементов массива с заданными индексами
40. Особенности алгоритма обмена местами двух строк массива
41. Особенности алгоритма формирования одномерного массива из элементов двумерного массива, удовлетворяющих некоторому условию
42. Особенности алгоритма формирования одномерного массива из индексов элементов двумерного массива, удовлетворяющих некоторому условию

**Вопросы для промежуточной аттестации (экзамен) по оценке освоения
компетенции ПК-2 (3 семестр)**

- 43. Ошибки программ: понятие и виды
- 44. Организация тестирования программ
- 45. Понятие отладки программ.

2.4 Перечень примерных тем для курсовой работы

Темы для курсовой работы по оценке освоения компетенций ОПК-2, ОПК-3, ОПК-7, ПК-2 (3 семестр)

1. Создание эффективных алгоритмов для обработки данных структурированного типа.
2. Разработка методов обработки структурированных данных с применением типовых алгоритмов
3. Проектирование и реализация алгоритмов на Python для работы со структурированными данными
4. Методология разработки алгоритмов на Python для работы с различными типами структурированных данных
5. Разработка универсальных алгоритмов для гибкой обработки структурированных данных
6. Алгоритмическое решение задач обработки данных структурированного типа с использованием Python
7. Написать программу, которая формирует двумерную матрицу чисел размерностью $n \times n$. Затем формирует два одномерных массива, причем в один записываются элементы, расположенные на главной диагонали и выше, в другой – элементы матрицы, лежащие ниже главной диагонали, и выводит все массивы на экран. Осуществить бинарный поиск заданного элемента во втором массиве. Результаты поиска вывести на экран.
8. Дана квадратная матрица размерностью $n \times n$. Написать программу формирования последовательности В, элементами которой являются элементы таблицы А, расположенные над главной диагональю. В полученной последовательности найти три наименьших элемента. Если количество элементов между 1-м и 2-м элементами совпадает с количеством элементов между 2-м и 3-м элементами, то поменять их местами, сохранив порядок следования (без использования дополнительного массива). Преобразованный массив вывести на экран.
9. Написать программу, которая предлагает пользователю некоторый список функций для построения графиков: $y=x^4/(x^3+1)$; $y=ax^2+bx+c$; $y=\sin(x)+b$; $y=1/(x^2+x+1)$. После выбора функции, задания коэффициентов и отрезка, на котором выполняется построение, программа строит соответствующий график. Затем значение коэффициентов и положение графика можно изменить (например, с помощью клавиш управления курсором), после чего график перестраивается и записывается обновленное уравнение кривой.
10. Написать программу, позволяющую отсортировать массив двумя способами (обменная и выбором). Обеспечить определение и отображение времени сортировки каждым методом. Предусмотреть операции создания файла, записи данных в файл и чтения файла для обработки.
11. Написать программу тестирования знаний студентов по дисциплине Программирование и основы алгоритмизации. Вопросы для тестирования должны выбираться из файла случайным образом. Программа должна осуществлять подсчет правильных ответов и выводить на экран результаты тестирования (общее число вопросов, количество правильных ответов, оценку).
12. Написать программу быстрой сортировки массива, двоичного поиска повторяющихся элементов и замены их пользователем. Предусмотреть операции создания файла, записи данных в файл и чтения файла для обработки.
13. Написать программу, вычисляющую определитель квадратной матрицы размерностью $n \times n$. Содержимое матрицы должно отображаться на экране. Обеспечить редактирование элементов матрицы, автоматический подсчет определителя и вывод его значения на экран.
14. Написать программу шифрования и дешифрования текстовых файлов. В основу шифрования положить коэффициент смещения букв, в результате использования которого, например, буква А меняется на Д. Исходный и результирующий текст вывести на экран и в файл.
15. Написать программу, моделирующую тренажер по устному счету. Пользователь вводит разрядность операндов, тип операции (+ - * /) на множестве натуральных чисел и количество примеров. Компьютер генерирует случайным образом операнды, вычисляет результат операции и выводит пользователю серию примеров, в каждом из которых один из операндов или результат замаскирован. Пользователь вводит пропущенное число, компьютер проверяет правильность и ведет статистику ошибок.
16. Написать программу шифрования и дешифрования данных, содержащихся в текстовых файлах методом прямой замены. Зашифрованные данные сохранить в файле, исходный и результирующий отобразить на экране.

17. Написать программу, которая будет исправлять неточности редактирования текстового файла. Программа должна убирать знаки пробела перед знаком препинания, ставить один пробел после знака препинания и между словами, исправлять первую букву предложения на заглавную. Информация считывается из текстового файла и записывается в другой текстовый файл на жестком диске.

18. Написать программу, обеспечивающую создание списка абитуриентов. Предусмотреть ввод фамилии, имени и отчества, года рождения, оценок за вступительные экзамены (математика, физика, русский язык). Программа должна позволять сохранить список в файл, загрузить его и произвести сортировку по следующим полям в порядке убывания: общий балл за экзамен; фамилия, имя, отчество; год рождения.

19. Написать программу демонстрации графических возможностей TurboC++ и обучающую работе с основными графическими процедурами и функциями. Программа должна контролировать усвоение изученного материала (в виде теста или в какой-либо другой форме).

20. Написать программу, которая по введенному пользователем вектору чисел строит диаграммы различных видов в графическом режиме. В интерфейсе предусмотреть ввод вектора чисел и вывод диаграммы (не более 5-ти видов).

Выбрать какой-либо школьный предмет (информатика, математика и т.д.), подобрать ребусы по нему и предложить их для решения. Программа должна позволить выбрать тот или иной ребус, проконтролировать его решение и подвести итоги при завершении работы.

21. Написать программу, обеспечивающую создание файла для хранения и обработки данных о студентах учебной группы. Обработка должна включать операции добавления, удаления, корректировки записей и просмотра данных в табличном отображении. Записи организовать в виде связанного списка. Перечисленные операции реализовать с помощью стандартных операторов для заданного абстрактного типа данных.

22. Написать программу, обеспечивающую создание файла для хранения и обработки данных о мониторах. Обработка должна включать операции добавления, удаления, корректировки записей и просмотра данных в табличном отображении. Записи организовать в виде стека. Перечисленные операции реализовать с помощью стандартных операторов для заданного абстрактного типа данных.

23. Написать программу для решения систем линейных уравнений методом Гаусса. Исходные данные и результаты решения вывести в файл. Исходные данные разместить в динамической области памяти.

24. Написать программу для решения систем линейных уравнений итерационным методом Зейделя. Исходные данные и результаты решения вывести в файл. Исходные данные разместить в динамической области памяти.

25. Написать программу для решения нелинейных уравнений методом половинного деления и методом Ньютона. Сравнить методы по быстродействию.

26. Написать программу вычисления интегралов методами прямоугольников, трапеций и Симпсона. Сравнить время реализации методов при заданной точности.

27. Написать программу решения системы линейных уравнений методом Крамера. Количество переменных до 20. Программа должна выводить результаты решения задачи, а также время, затраченное на расчет с точностью до сотых долей секунд.

28. Дана квадратная матрица размерностью $n \times n$. Написать программу формирования последовательности A, элементами которой являются элементы таблицы B, расположенные над второстепенной диагональю. В полученной последовательности найти три наибольших элемента. Если количество элементов между 2-м и 3-м элементами совпадает с количеством элементов между 1-м и 2-м элементами, то поменять их местами, сохранив порядок следования (без использования дополнительного массива). Преобразованный массив вывести на экран и в файл.

29. Написать программу, которая формирует и записывает в файл двумерную матрицу чисел размерностью $n \times n$. Затем читает матрицу из файла и формирует два одномерных массива, причем в один записываются элементы, расположенные на второстепенной диагонали и ниже, в другой – элементы матрицы, лежащие ниже второстепенной диагонали и выводит все массивы на экран. Осуществить бинарный поиск заданного элемента в первом массиве. Результаты поиска вывести на экран.

30. Написать программу, позволяющую отсортировать массив двумя способами (вставками и выбором). Обеспечить определение и отображение времени сортировки каждым методом. Предусмотреть операции создания файла, записи данных в файл и чтения файла для обработки.