

МИНИСТЕРСТВО НАУКИ И ВЫСШЕГО ОБРАЗОВАНИЯ РОССИЙСКОЙ
ФЕДЕРАЦИИ
федеральное государственное автономное образовательное учреждение высшего
образования
"САНКТ-ПЕТЕРБУРГСКИЙ ГОСУДАРСТВЕННЫЙ УНИВЕРСИТЕТ
АЭРОКОСМИЧЕСКОГО ПРИБОРОСТРОЕНИЯ"

Кафедра № 82

УТВЕРЖДАЮ

Руководитель образовательной программы

д.э.н., проф. _____

(должность, уч. степень, звание)

А.С. Будагов _____

(инициалы, фамилия)

(подпись)

« 25 » февраля 2026 г.

РАБОЧАЯ ПРОГРАММА ДИСЦИПЛИНЫ

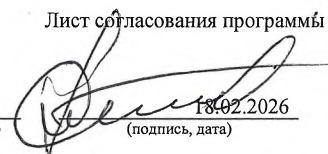
«Дополнительные разделы инженерного ядра»
(Наименование дисциплины)

Код направления подготовки/ специальности	09.03.03
Наименование направления подготовки/ специальности	Прикладная информатика
Наименование направленности/ специализации	Интеллектуальные информационные системы и технологии
Форма обучения	очная
Год приема	2026

Лист согласования программы

Программу составил (а)

Проф., д. пед. н., доц.
(должность, уч. степень, звание)


(подпись, дата)

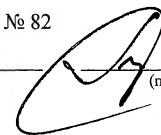
А. Г. Степанов
(инициалы, фамилия)

Программа одобрена на заседании кафедры № 82

« 19 » 02 2026 г, протокол № 7

Заведующий кафедрой № 82

д.э.н., проф.
(уч. степень, звание)



19.02.2026
(подпись, дата)

А.С. Будагов
(инициалы, фамилия)

Заместитель директора института №8 по методической работе

доц., к.э.н., доц.
(должность, уч. степень, звание)



19.02.2026
(подпись, дата)

Л.В. Рудакова
(инициалы, фамилия)

Аннотация

Дисциплина «Дополнительные разделы инженерного ядра» входит в образовательную программу высшего образования – программу бакалавриата по направлению подготовки/ специальности 09.03.03 «Прикладная информатика» направленности/специализации «Интеллектуальные информационные системы и технологии». Дисциплина реализуется кафедрой «№82».

Дисциплина не является обязательной при освоении обучающимся образовательной программы и направлена на углубленное формирование следующих компетенций:

УК-2 «Способен определять круг задач в рамках поставленной цели и выбирать оптимальные способы их решения, исходя из действующих правовых норм, имеющихся ресурсов и ограничений»

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»

Содержание дисциплины охватывает круг вопросов, связанных с разработкой, внедрением и эксплуатацией программно-технических комплексов в условиях современной цифровой среды. Предметная область включает изучение методов и инструментов для решения профессиональных задач с использованием информационных технологий, а также формирование компетенций, необходимых для эффективного применения цифровых средств, обеспечения информационной безопасности и создания программного обеспечения.

Преподавание дисциплины предусматривает следующие формы организации учебного процесса: практические занятия, самостоятельная работа обучающегося.

Программой дисциплины предусмотрены следующие виды контроля: текущий контроль успеваемости, промежуточная аттестация в форме зачета (4 семестр).

Общая трудоемкость освоения дисциплины составляет 1 зачетную единицу, 36 часов.

Язык обучения по дисциплине «русский»

1. Перечень планируемых результатов обучения по дисциплине

1.1. Цели преподавания дисциплины

Цель дисциплины заключается в формировании у обучающихся необходимых знаний, умений и навыков для разработки программного обеспечения, применения информационно-коммуникационных технологий и обеспечения информационной безопасности, а также в создании поддерживающей образовательной среды для освоения современных цифровых инструментов и развития навыков решения междисциплинарных задач, что позволит будущим специалистам успешно применять полученные компетенции в профессиональной деятельности.

1.2. Дисциплина является факультативной дисциплиной по направлению образовательной программы высшего образования (далее – ОП ВО).

1.3. Перечень планируемых результатов обучения по дисциплине, соотнесенных с планируемыми результатами освоения ОП ВО.

В результате изучения дисциплины обучающийся должен обладать следующими компетенциями или их частями. Компетенции и индикаторы их достижения приведены в таблице 1.

Таблица 1 – Перечень компетенций и индикаторов их достижения

Категория (группа) компетенции	Код и наименование компетенции	Код и наименование индикатора достижения компетенции
Универсальные компетенции	УК-2 Способен определять круг задач в рамках поставленной цели и выбирать оптимальные способы их решения, исходя из действующих правовых норм, имеющихся ресурсов и ограничений	УК-2.В.3 владеть навыками использования цифровых средств для решения поставленной задачи
Общепрофессиональные компетенции	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-3.У.1 уметь решать стандартные задачи профессиональной деятельности на основе информационной и библиографической культуры с применением информационно-коммуникационных технологий и с учетом основных требований информационной безопасности
Общепрофессиональные компетенции	ОПК-7 Способен разрабатывать	ОПК-7.В.1 владеть навыками программирования, отладки и

	алгоритмы и программы, пригодные для практического применения	тестирования прототипов программно-технических комплексов задач
--	---	---

2. Место дисциплины в структуре ОП

Дисциплина может базироваться на знаниях, ранее приобретенных обучающимися при изучении следующих дисциплин:

- «Основы цифровой грамотности»,
- «Информатика»,
- «Основы программирования»,
- «Основы проектной деятельности»

3. Объем и трудоемкость дисциплины

Данные об общем объеме дисциплины, трудоемкости отдельных видов учебной работы по дисциплине (и распределение этой трудоемкости по семестрам) представлены в таблице 2.

Таблица 2 – Объем и трудоемкость дисциплины

Вид учебной работы	Всего	Трудоемкость по семестрам
		№4
1	2	3
Общая трудоемкость дисциплины, ЗЕ/ (час)	1/ 36	1/ 36
Из них часов практической подготовки		
Аудиторные занятия, всего час.	17	17
в том числе:		
лекции (Л), (час)		
практические/семинарские занятия (ПЗ), (час)	17	17
лабораторные работы (ЛР), (час)		
курсовой проект (работа) (КП, КР), (час)		
экзамен, (час)		
Самостоятельная работа, всего (час)	19	19
Вид промежуточной аттестации: зачет, дифф. зачет, экзамен (Зачет, Дифф. зач, Экз.)	Зачет,	Зачет,

4. Содержание дисциплины

4.1. Распределение трудоемкости дисциплины по разделам и видам занятий.

Разделы, темы дисциплины и их трудоемкость приведены в таблице 3.

Таблица 3 – Разделы, темы дисциплины, их трудоемкость

Разделы, темы дисциплины	Лекции (час)	ПЗ (СЗ) (час)	ЛР (час)	КП/КР (час)	СР (час)
Семестр 4					
Раздел 1. Основы программирования на Python Тема 1.1 Разработка базовой программы для решения задачи с использованием цифровых средств.		4			4

Раздел 2. Работа MySQL в Python Тема 2.1 Расширение функционала программы и сохранение данных в базу данных.		4			4
Раздел 3. Информационная безопасность Тема 3.1 Анализ информационной безопасности программы и базы данных.		4			4
Раздел 4. UI/UX Тема 4.1 Развитие навыков создания программных интерфейсов и их тестирования.		4			4
Раздел 5. Быстрая разработка приложений (RAD) Тема 5.1 Быстрое создание прототипа дополнительного функционала.		1			3
Итого в семестре:		17			19
Итого	0	17	0	0	19

Практическая подготовка заключается в непосредственном выполнении обучающимися определенных трудовых функций, связанных с будущей профессиональной деятельностью.

4.2. Содержание разделов и тем лекционных занятий.

Содержание разделов и тем лекционных занятий приведено в таблице 4.

Таблица 4 – Содержание разделов и тем лекционного цикла

Номер раздела	Название и содержание разделов и тем лекционных занятий
Учебным планом не предусмотрено	

4.3. Практические (семинарские) занятия

Темы практических занятий и их трудоемкость приведены в таблице 5.

Таблица 5 – Практические занятия и их трудоемкость

№ п/п	Темы практических занятий	Формы практических занятий	Трудоемкость, (час)	Из них практической подготовки, (час)	№ раздела дисциплины
Семестр 4					
1	Разработка базовой программы для решения задачи с использованием цифровых средств.	Индивидуальная работа за компьютером, мозговой штурм, анализ кейсов	4	4	1
2	Расширение функционала программы и сохранение данных в базу данных.	Индивидуальная работа за компьютером, мозговой штурм, анализ кейсов	4	4	2
3	Анализ информационной безопасности программы и базы данных.	Индивидуальная работа за компьютером, мозговой штурм, анализ кейсов	4	4	3
4	Разработка графического интерфейса для программы.	Индивидуальная работа за компьютером, мозговой штурм, анализ кейсов	4	4	4

5	Быстрое создание прототипа дополнительного функционала.	Индивидуальная работа за компьютером, мозговой штурм, анализ кейсов	1	1	5
Всего			17		

4.4. Лабораторные занятия

Темы лабораторных занятий и их трудоемкость приведены в таблице 6.

Таблица 6 – Лабораторные занятия и их трудоемкость

№ п/п	Наименование лабораторных работ	Трудоемкость, (час)	Из них практической подготовки, (час)	№ раздела дисциплины
Учебным планом не предусмотрено				
Всего				

4.5. Выполнение курсового проекта/ курсовой работы

Учебным планом не предусмотрено

4.6. Самостоятельная работа обучающихся

Виды самостоятельной работы и ее трудоемкость приведены в таблице 7.

Таблица 7 – Виды самостоятельной работы и ее трудоемкость

Вид самостоятельной работы	Всего, час	Семестр 4, час
1	2	3
Изучение теоретического материала дисциплины (ТО)	10	10
Курсовое проектирование (КП, КР)	-	-
Расчетно-графические задания (РГЗ)	-	-
Выполнение реферата (Р)	-	-
Подготовка к текущему контролю успеваемости (ТКУ)	4	4
Домашнее задание (ДЗ)	-	-
Контрольные работы заочников (КРЗ)	-	-
Подготовка к промежуточной аттестации (ПА)	5	5
Всего:	19	19

5. Перечень учебно-методического обеспечения

для самостоятельной работы обучающихся по дисциплине (модулю)

Учебно-методические материалы для самостоятельной работы обучающихся указаны в п.п. разделов 6-11.

6. Перечень печатных и электронных учебных изданий

Перечень печатных и электронных учебных изданий приведен в таблице 8.

Таблица 8– Перечень печатных и электронных учебных изданий

Шифр/	Библиографическая ссылка	Количество
-------	--------------------------	------------

URL адрес		экземпляров в библиотеке (кроме электронных экземпляров)
https://disk.yandex.ru/i/T6J4Br87To8uew	Машнин Т. Создание настольных Python приложений с графическим интерфейсом пользователя / Машнин Т. – Москва: ЛитРес: Самиздат, 2021. – 131с. ISBN: 978-5-532-96908-7	
https://disk.yandex.ru/i/0sChjQ1vRx0gfA	Грофф, Джеймс Р., Вайнберг, Пол Н., Оппель, Эндрю Дж. SQL: полное руководство, 3-е изд. : Пер. с англ. - М.: ООО "И.Д. Вильямс", 2015 - 960 с. ISBN 978-5-8459-1654-9 (рус.)	
https://disk.yandex.ru/i/ZW3jpEDPq-9dnA	Мэтиз Эрик Изучаем Python: программирование игр, визуализация данных, веб-приложения. 3-е изд., доп. и перераб. — СПб.: Питер, 2025 — 560 с.: ил. — (Серия «Библиотека программиста»). ISBN 978-5-4461-4112-8	
https://disk.yandex.ru/i/_1SYOPZKCS8FNg	Михаил Тарасов. Уязвимость SQL-инъекция. Практическое руководство для хакеров. / Михаил Тарасов. 2019. – 133 с.	

7. Перечень электронных образовательных ресурсов информационно-телекоммуникационной сети «Интернет»

Перечень электронных образовательных ресурсов информационно-телекоммуникационной сети «Интернет», необходимых для освоения дисциплины приведен в таблице 9.

Таблица 9 – Перечень электронных образовательных ресурсов информационно-телекоммуникационной сети «Интернет»

URL адрес	Наименование
https://pro.guar.ru	Материалы и задания по дисциплине

8. Перечень информационных технологий

8.1. Перечень программного обеспечения, используемого при осуществлении образовательного процесса по дисциплине.

Перечень используемого программного обеспечения представлен в таблице 10.

Таблица 10– Перечень программного обеспечения

№ п/п	Наименование
1	Microsoft Windows OS Договор №1303-3 от 30.12.2019 или аналог на Linux
2	Microsoft Office 2016 Договор №278 от 18.06.2020 / LibreOffice
3	Anaconda Community (бесплатная версия)
4	MySQL (свободно распространяемо программное обеспечение)
5	VirtualBOX (бесплатная версия)

8.2. Перечень информационно-справочных систем, используемых при осуществлении образовательного процесса по дисциплине

Перечень используемых информационно-справочных систем представлен в таблице 11.

Таблица 11– Перечень информационно-справочных систем

№ п/п	Наименование
1	ЭБС Лань
2	Znanium
3	eLIBRARY
4	КиберЛенинка

9. Материально-техническая база

Состав материально-технической базы, необходимой для осуществления образовательного процесса по дисциплине, представлен в таблице 12.

Таблица 12 – Состав материально-технической базы

№ п/п	Наименование составной части материально-технической базы	Номер аудитории (при необходимости)
1	Аудитория для практических занятий	14-06 – 14-11, 13-06 (Ленсовета 14)
2	Аудитория для самостоятельной работы студента	31-07 (Ленсовета 14)
3	Аудитория для промежуточной аттестации	14-06 – 14-11 (Ленсовета 14)

10. Оценочные средства для проведения промежуточной аттестации

10.1. Состав оценочных средств для проведения промежуточной аттестации обучающихся по дисциплине приведен в таблице 13.

Таблица 13 – Состав оценочных средств для проведения промежуточной аттестации

Вид промежуточной аттестации	Перечень оценочных средств
Зачет	Список вопросов; Тесты;

10.2. В качестве критериев оценки уровня сформированности (освоения) компетенций обучающимися применяется 5-балльная шкала оценки сформированности

компетенций, которая приведена в таблице 14. В течение семестра может использоваться 100-балльная шкала модульно-рейтинговой системы Университета, правила использования которой, установлены соответствующим локальным нормативным актом ГУАП.

Таблица 14 – Критерии оценки уровня сформированности компетенций

Оценка компетенции	Характеристика сформированных компетенций
5-балльная шкала	
«отлично» «зачтено»	Обучающийся: – глубоко и всесторонне усвоил программный материал; – уверенно, логично, последовательно и грамотно его излагает; – опираясь на знания основной и дополнительной литературы, тесно связывает усвоенные научные положения с практической деятельностью направления; – умело обосновывает и аргументирует выдвигаемые им идеи; – делает выводы и обобщения; – свободно владеет системой специализированных понятий. – правильно выполнил от 90% до 100% тестовых заданий**.
«хорошо» «зачтено»	Обучающийся: – твердо усвоил программный материал, грамотно и по существу излагает его, опираясь на знания основной литературы; – не допускает существенных неточностей; – увязывает усвоенные знания с практической деятельностью направления; – аргументирует научные положения; – делает выводы и обобщения; – владеет системой специализированных понятий. – правильно выполнил от 70% до 89% тестовых заданий**.
«удовлетворительно» «зачтено»	– обучающийся усвоил только основной программный материал, по существу излагает его, опираясь на знания только основной литературы; – допускает несущественные ошибки и неточности; – испытывает затруднения в практическом применении знаний направления; – слабо аргументирует научные положения; – затрудняется в формулировании выводов и обобщений; – частично владеет системой специализированных понятий. – правильно выполнил от 51% до 69% тестовых заданий**.
«неудовлетворительно» «не зачтено»	– обучающийся не усвоил значительной части программного материала; – допускает существенные ошибки и неточности при рассмотрении проблем в конкретном направлении; – испытывает трудности в практическом применении знаний; – не может аргументировать научные положения; – не формулирует выводов и обобщений. – правильно выполнил менее 51% тестовых заданий**.

10.3. Типовые контрольные задания или иные материалы.

Вопросы (задачи) для экзамена представлены в таблице 15.

Таблица 15 – Вопросы (задачи) для экзамена

№ п/п	Перечень вопросов (задач) для экзамена	Код индикатора
	Учебным планом не предусмотрено	

Вопросы (задачи) для зачета / дифф. зачета представлены в таблице 16.

Таблица 16 – Вопросы (задачи) для зачета / дифф. зачета

№ п/п	Перечень вопросов (задач) для зачета / дифф. зачета	Код
-------	---	-----

		индикатора
1	Какие основные этапы включает разработка базовой программы на Python для решения конкретной задачи?	УК-2.В.3
2	Какие цифровые инструменты и библиотеки Python можно использовать для анализа данных?	УК-2.В.3
3	Каким образом можно интегрировать MySQL с Python для хранения и обработки данных?	УК-2.В.3
4	Какие методы отладки и тестирования кода наиболее эффективны при работе с цифровыми средствами?	УК-2.В.3
5	Как использование UI/UX-дизайна помогает улучшить взаимодействие пользователя с программой?	УК-2.В.3
6	Какие преимущества дает применение RAD-подхода при создании прототипов программных решений?	УК-2.В.3
7	Как правильно организовать поиск информации для решения задачи с использованием библиографических ресурсов?	ОПК-3.У.1
8	Какие требования информационной безопасности необходимо учитывать при разработке программного обеспечения?	ОПК-3.У.1
9	Какие инструменты информационно-коммуникационных технологий наиболее подходят для анализа и документирования процесса разработки ПО?	ОПК-3.У.1
10	Как обеспечить защиту данных при работе с базами данных MySQL в Python?	ОПК-3.У.1
11	Какую роль играет информационная культура при выборе источников данных для программного проекта?	ОПК-3.У.1
12	Какие стандарты информационной безопасности следует учитывать при тестировании программных продуктов?	ОПК-3.У.1
13	Какие основные принципы программирования на Python необходимо учитывать при создании прототипа программы?	ОПК-7.В.1
14	Как провести отладку программы, если возникла ошибка при подключении к базе данных MySQL?	ОПК-7.В.1
15	Какие методы тестирования можно использовать для проверки функциональности программного интерфейса?	ОПК-7.В.1
16	Какие шаги необходимо выполнить для создания и тестирования прототипа дополнительного функционала программы?	ОПК-7.В.1
17	Какие инструменты Python наиболее эффективны для быстрого прототипирования программных решений?	ОПК-7.В.1
18	Какие подходы к отладке и тестированию помогают минимизировать количество ошибок в программно-технических комплексах?	ОПК-7.В.1

Перечень тем для выполнения курсового проекта/ курсовой работы представлены в таблице 17.

Таблица 17 – Перечень тем для выполнения курсового проекта / курсовой работы

№ п/п	Примерный перечень тем для выполнения курсового проекта/ курсовой работы
	Учебным планом не предусмотрено

Вопросы для проведения промежуточной аттестации в виде тестирования представлены в таблице 18.

Таблица 18 – Примерный перечень вопросов для тестов

№ п/п	Примерный перечень вопросов для тестов	Код индикатора								
1	<p>Задание закрытого типа с выбором одного правильного ответа</p> <p>Вопрос: Какой метод является наиболее безопасным для защиты данных?</p> <p>a) Base64 b) SHA-256 c) MD5 d) ROT13</p> <p>Правильный ответ: b</p>	УК-2.В.3								
2	<p>Задание закрытого типа с выбором нескольких правильных ответов</p> <p>Вопрос: Какие принципы относятся к информационной безопасности?</p> <p>1. Конфиденциальность 2. Целостность 3. Доступность 4. Простота реализации 5. Экономическая эффективность</p> <p>Правильные ответы: 1,2,3</p>	УК-2.В.3								
3	<p>Задание закрытого типа на установление соответствия</p> <p>Вопрос: Сопоставьте этапы разработки программы с их описанием:</p> <table border="1" data-bbox="349 1256 1166 1413"> <thead> <tr> <th>Этап</th> <th>Содержание</th> </tr> </thead> <tbody> <tr> <td>1. Анализ требований</td> <td>А) Создание интерфейса</td> </tr> <tr> <td>2. Проектирование</td> <td>Б) Определение задачи</td> </tr> <tr> <td>3. Реализация</td> <td>В) Написание кода</td> </tr> </tbody> </table> <p>Правильное соответствие: А - 2, В - 1, С - 3</p>	Этап	Содержание	1. Анализ требований	А) Создание интерфейса	2. Проектирование	Б) Определение задачи	3. Реализация	В) Написание кода	УК-2.В.3
Этап	Содержание									
1. Анализ требований	А) Создание интерфейса									
2. Проектирование	Б) Определение задачи									
3. Реализация	В) Написание кода									
4	<p>Задание закрытого типа на установление правильной последовательности</p> <p>Вопрос: Установите правильную последовательность этапов работы с базой данных:</p> <p>1. Подключение к БД 2. Выполнение запроса 3. Получение результатов 4. Закрытие соединения</p> <p>Правильная последовательность: 1 → 2 → 3 → 4</p>	УК-2.В.3								
5	<p>Задание открытого типа (с коротким ответом)</p> <p>Вопрос: Как называется процесс преобразования информации в формат, непонятный для несанкционированного пользователя?</p>	УК-2.В.3								

	Эталонный ответ: Шифрование											
6	<p>Задание открытого типа (с развёрнутым обоснованием)</p> <p>Вопрос: Опишите основные этапы быстрой разработки приложений (RAD)</p> <p>Ключевые элементы ответа:</p> <p>Планирование проекта Моделирование бизнес-процессов Создание прототипов Тестирование Внедрение Итеративное улучшение</p>	УК-2.В.3										
7	<p>Задание закрытого типа с выбором одного правильного ответа</p> <p>Вопрос: Что является приоритетным при разработке UI/UX?</p> <ol style="list-style-type: none"> 1. Красочность интерфейса 2. Удобство пользователя 3. Сложность реализации 4. Скорость загрузки <p>Правильный ответ: 2</p>	ОПК-3.У.1										
8	<p>Задание закрытого типа с выбором нескольких правильных ответов</p> <p>Вопрос: Какие аспекты важны при обеспечении информационной безопасности?</p> <ol style="list-style-type: none"> 1. Хэширование паролей 2. Использование HTTPS 3. Частота обновления контента 4. Резервное копирование <p>Ответ: 1, 2, 4</p>	ОПК-3.У.1										
9	<p>Задание закрытого типа на установление соответствия</p> <p>Вопрос: Сопоставьте этапы разработки программы с их описанием:</p> <table border="1" style="width: 100%; border-collapse: collapse;"> <thead> <tr> <th>Этап</th> <th>Содержание</th> </tr> </thead> <tbody> <tr> <td>1. Основы программирования на Python</td> <td>А) Работа с базами данных</td> </tr> <tr> <td>2. Работа MySQL в Python</td> <td>Б) Создание безопасного ПО</td> </tr> <tr> <td>3. Информационная безопасность</td> <td>В) Быстрая разработка приложений</td> </tr> <tr> <td>4. Быстрая разработка приложений (RAD)</td> <td>Г) Написание базовых программ</td> </tr> </tbody> </table> <p>Правильное соответствие: А-4, Б-1, В-2, Г-3</p>	Этап	Содержание	1. Основы программирования на Python	А) Работа с базами данных	2. Работа MySQL в Python	Б) Создание безопасного ПО	3. Информационная безопасность	В) Быстрая разработка приложений	4. Быстрая разработка приложений (RAD)	Г) Написание базовых программ	ОПК-3.У.1
Этап	Содержание											
1. Основы программирования на Python	А) Работа с базами данных											
2. Работа MySQL в Python	Б) Создание безопасного ПО											
3. Информационная безопасность	В) Быстрая разработка приложений											
4. Быстрая разработка приложений (RAD)	Г) Написание базовых программ											
10	<p>Задание закрытого типа на установление правильной последовательности</p> <p>Вопрос: Расположите этапы работы с базой данных в правильном порядке:</p>	ОПК-3.У.1										

	<ol style="list-style-type: none"> 1. Создание таблиц 2. Подключение к БД 3. Внесение данных 4. Выборка данных <p>Правильная последовательность: 2 → 1 → 3 → 4</p>					
11	<p>Задание открытого типа (с коротким ответом)</p> <p>Вопрос: Как называется процесс быстрого создания прототипов?</p> <p>Эталонный ответ: RAD (Rapid Application Development)</p>	ОПК-3.У.1				
12	<p>Задание открытого типа (с развёрнутым обоснованием)</p> <p>Вопрос: Опишите процесс разработки безопасного приложения на Python.</p> <p>Ключевые элементы ответа:</p> <p>Планирование архитектуры с учетом безопасности Использование безопасных методов хранения данных Внедрение системы аутентификации Применение шифрования Тестирование на уязвимости Регулярное обновление и патчинг</p>	ОПК-3.У.1				
13	<p>Задание закрытого типа с выбором одного правильного ответа</p> <p>Вопрос: Какой раздел курса является фундаментальным для начала работы над программно-техническим комплексом?</p> <ol style="list-style-type: none"> 1. Информационная безопасность 2. Основы программирования на Python 3. Работа MySQL в Python 4. UI/UX <p>Правильный ответ: 2</p>	ОПК-7.В.1				
14	<p>Задание закрытого типа с выбором нескольких правильных ответов</p> <p>Вопрос: Какие разделы курса необходимы для полного цикла тестирования программно-технического комплекса?</p> <ol style="list-style-type: none"> 1. Основы программирования на Python 2. Работа MySQL в Python 3. Информационная безопасность 4. UI/UX 5. Быстрая разработка приложений (RAD) <p>Ответ: 1, 3, 4, 5</p>	ОПК-7.В.1				
15	<p>Задание закрытого типа на установление соответствия</p> <p>Вопрос: Сопоставьте этапы разработки программного обеспечения:</p> <table border="1" style="width: 100%; border-collapse: collapse;"> <thead> <tr> <th style="width: 50%;">Этап</th> <th style="width: 50%;">Содержание</th> </tr> </thead> <tbody> <tr> <td>1. Этап 1</td> <td>А) Создание пользовательского</td> </tr> </tbody> </table>	Этап	Содержание	1. Этап 1	А) Создание пользовательского	ОПК-7.В.1
Этап	Содержание					
1. Этап 1	А) Создание пользовательского					

	интерфейса	
2. Этап 2	Б) Обеспечение безопасности данных	
3. Этап 3	В) Работа с базами данных	
4. Этап 4	Г) Быстрая разработка прототипов	
Правильное соответствие: А-2, Б-4, В-3, Г-1		
16	<p>Задание закрытого типа на установление правильной последовательности</p> <p>Вопрос: Установите правильную последовательность этапов разработки программно-технического комплекса:</p> <ol style="list-style-type: none"> 1. Разработка UI/UX интерфейса 2. Создание базовой программы 3. Обеспечение информационной безопасности 4. Подключение к базе данных 5. Создание прототипа дополнительного функционала <p>Правильная последовательность: 2 → 4 → 1 → 3 → 5</p>	ОПК-7.В.1
17	<p>Задание открытого типа (с коротким ответом)</p> <p>Вопрос: Как называется методология разработки, при которой создание прототипа происходит максимально быстро с возможностью его многократной доработки?</p> <p>Эталонный ответ: Быстрая разработка приложений (RAD)</p>	ОПК-7.В.1
18	<p>Задание открытого типа (с развёрнутым обоснованием)</p> <p>Вопрос: Опишите процесс разработки и тестирования программно-технического комплекса</p> <p>Ключевые элементы ответа:</p> <p>Процесс начинается с создания базовой программы на Python. Далее реализуется подключение к базе данных MySQL, после чего проводится анализ информационной безопасности всего комплекса. На следующем этапе разрабатывается и тестируется пользовательский интерфейс. Используя методологию RAD, создаются и тестируются прототипы дополнительного функционала. На каждом этапе выполняется отладка и тестирование соответствующих компонентов системы.</p>	ОПК-7.В.1

Ключи правильных ответов на тестовые вопросы размещены в приложении к РПД.

Таблица 18.1 – Критерии оценивания тестирования

Тип задания	Критерии оценки
Задание закрытого типа на установление соответствия считается верным, если установлены все соответствия (позиции из одного столбца верно сопоставлены с позициями другого столбца)	Полное совпадение с верным ответом оценивается 1 баллом, неверный ответ или его отсутствие – 0 баллов
Задание закрытого типа на установление	Полное совпадение с верным ответом

последовательности считается верным, если правильно указана вся последовательность цифр	оценивается 1 баллом, если допущены ошибки или ответ отсутствует – 0 баллов
Задание комбинированного типа с выбором одного верного ответа из четырех предложенных и обоснованием выбора считается верным, если правильно указана цифра и приведены конкретные аргументы, используемые при выборе ответа	Полное совпадение с верным ответом оценивается 1 баллом, неверный ответ или его отсутствие – 0 баллов
Задание комбинированного типа с выбором нескольких вариантов ответа из предложенных и развернутым обоснованием выбора считается верным, если правильно указаны цифры и приведены конкретные аргументы, используемые при выборе ответов	Полное совпадение с верным ответом оценивается 1 баллом, если допущены ошибки или ответ отсутствует – 0 баллов
Задание открытого типа с кратким / развернутым ответом считается верным, если ответ совпадает с эталонным по содержанию и полноте	Правильный ответ за задание оценивается в 3 балла, если допущена одна ошибка \ неточность \ ответ правильный, но не полный - 1 балл, если допущено более 1 ошибки \ ответ неправильный \ ответ отсутствует – 0 баллов

Перечень тем контрольных работ по дисциплине обучающихся заочной формы обучения, представлены в таблице 19.

Таблица 19 – Перечень контрольных работ

№ п/п	Перечень контрольных работ
	Не предусмотрено

10.4. Методические материалы, определяющие процедуры оценивания индикаторов, характеризующих этапы формирования компетенций, содержатся в локальных нормативных актах ГУАП, регламентирующих порядок и процедуру проведения текущего контроля успеваемости и промежуточной аттестации обучающихся ГУАП.

11. Методические указания для обучающихся по освоению дисциплины

11.1. Методические указания для обучающихся по освоению лекционного материала

Не предусмотрено учебным планом по данной дисциплине

11.2. Методические указания для обучающихся по участию в семинарах

Не предусмотрено учебным планом по данной дисциплине

11.3. Методические указания для обучающихся по прохождению практических занятий

Практическое занятие является одной из основных форм организации учебного процесса, заключающаяся в выполнении обучающимися под руководством преподавателя комплекса учебных заданий с целью усвоения научно-теоретических основ учебной дисциплины, приобретения умений и навыков, опыта творческой деятельности.

Целью практического занятия для обучающегося является привитие обучающимся умений и навыков практической деятельности по изучаемой дисциплине.

Планируемые результаты при освоении обучающимся практических занятий:

- закрепление, углубление, расширение и детализация знаний при решении конкретных задач;
- развитие познавательных способностей, самостоятельности мышления, творческой активности;
- овладение новыми методами и методиками изучения конкретной учебной дисциплины;
- выработка способности логического осмысления полученных знаний для выполнения заданий;
- обеспечение рационального сочетания коллективной и индивидуальной форм обучения.

Требования к проведению практических занятий

Разработка базовой программы для решения задачи с использованием цифровых средств

1. Определите задачу, которую программа будет решать (например, расчет данных или обработка информации).
2. Создайте структуру проекта:
 - 2.1 Создайте папку проекта.
 - 2.2 Разделите код на модули (например, main.py, utils.py).
3. Напишите код программы на Python, реализующий основную логику.
4. Проверьте корректность работы программы на тестовых данных.
5. Добавьте комментарии к коду и напишите краткое описание программы.

Расширение функционала программы и сохранение данных в базу данных

Определите, какие данные нужно сохранять в базе данных (например, результаты вычислений).

1. Спроектируйте базу данных:
 - 1.1 Определите структуру таблиц.
 - 1.2 Напишите SQL-скрипт для создания таблиц:

```
CREATE TABLE results (  
    id INTEGER PRIMARY KEY AUTOINCREMENT,  
    input_data TEXT,  
    output_data TEXT,  
    timestamp DATETIME DEFAULT CURRENT_TIMESTAMP  
);
```

2. Подключитесь к базе данных с помощью библиотеки sqlite3

```
import sqlite3  
conn = sqlite3.connect('database.db')  
cursor = conn.cursor()
```

3. Добавьте функционал для записи данных в базу данных

```
def save_to_db(input_data, output_data):
    cursor.execute("INSERT INTO results (input_data, output_data) VALUES (?, ?)", (input_data, output_data))
    conn.commit()
```

4. Убедитесь, что данные корректно сохраняются в базе данных.
5. Проверьте целостность данных после нескольких операций записи.

Анализ информационной безопасности программы и базы данных

1. Определите потенциальные уязвимости программы и базы данных (например, SQL-инъекции, незащищенные пароли).
2. Защитите запросы к базе данных:

2.1 Используйте параметризованные запросы:

```
cursor.execute("SELECT * FROM results WHERE id = ?", (user_input,))
```

3. Настройте права доступа к базе данных.
4. Добавьте логирование действий пользователя:

```
import logging
logging.basicConfig(filename='app.log', level=logging.INFO)
logging.info('User performed an action.')
```

5. Проведите тестирование на наличие уязвимостей.
6. Убедитесь, что чувствительные данные защищены и действия пользователей фиксируются.

Разработка графического интерфейса для программы

1. Определите, какие элементы управления нужны в интерфейсе (например, кнопки, текстовые поля).
2. Выберите библиотеку для создания GUI (например, tkinter).
3. Создайте интерфейс:

3.1 Добавьте элементы управления (кнопки, текстовые поля, метки):

```
import tkinter as tk

root = tk.Tk()
root.title("Программа")

label = tk.Label(root, text="Введите данные:")
label.pack()
```

```

entry = tk.Entry(root)
entry.pack()

def on_submit():
    user_input = entry.get()
    print(f"Вы ввели: {user_input}")

button = tk.Button(root, text="Отправить", command=on_submit)
button.pack()

root.mainloop()

```

4. Свяжите интерфейс с логикой программы.
5. Проверьте удобство использования интерфейса.
6. Убедитесь, что все элементы работают корректно.

Быстрое создание прототипа дополнительного функционала

1. Определите новый функционал (например, экспорт данных в файл).
2. Создайте минимально рабочий прототип:

```

def export_to_csv(data, filename):
    with open(filename, 'w') as file:
        file.write("input_data,output_data\n")
    for row in data:
        file.write(f"{row[0]},{row[1]}\n")

```

3. Интегрируйте прототип в основную программу.
4. Проверьте корректность работы нового функционала.
5. Оптимизируйте код после тестирования.

11.4. Методические указания для обучающихся по выполнению лабораторных работ

Не предусмотрено учебным планом по данной дисциплине

11.5. Методические указания для обучающихся по выполнению курсового проекта/ курсовой работы

Не предусмотрено учебным планом по данной дисциплине

11.6. Методические указания для обучающихся по прохождению самостоятельной работы

В ходе выполнения самостоятельной работы, обучающийся выполняет работу по заданию и при методическом руководстве преподавателя, но без его непосредственного участия.

В процессе выполнения самостоятельной работы у обучающегося формируется целесообразное планирование рабочего времени, которое позволяет ему развивать умения и навыки в усвоении и систематизации приобретаемых знаний, обеспечивает высокий уровень успеваемости в период обучения, помогает получить навыки повышения профессионального уровня.

Методическими материалами, направляющими самостоятельную работу обучающихся являются:

- учебно-методический материал по дисциплине;

11.7. Методические указания для обучающихся по прохождению текущего контроля успеваемости.

Текущий контроль успеваемости предусматривает контроль качества знаний обучающихся, осуществляемого в течение семестра с целью оценивания хода освоения дисциплины.

Текущий контроль успеваемости проводится путем выполнения и защиты первых трех практических работ. Данные текущего контроля успеваемости не учитываются при проведении зачета.

11.8. Методические указания для обучающихся по прохождению промежуточной аттестации.

Промежуточная аттестация обучающихся предусматривает оценивание промежуточных и окончательных результатов обучения по дисциплине. Она включает в себя:

- зачет – это форма оценки знаний, полученных обучающимся в ходе изучения учебной дисциплины в целом или промежуточная (по окончании семестра) оценка знаний обучающимся по отдельным разделам дисциплины с аттестационной оценкой «зачтено» или «не зачтено».

Зачет проходит в форме тестирования, примерный перечень вопросов которого представлен в таблице 18. Для получения допуска к зачету необходимо выполнить и защитить все практические работы

Лист внесения изменений в рабочую программу дисциплины

Дата внесения изменений и дополнений. Подпись внесшего изменения	Содержание изменений и дополнений	Дата и № протокола заседания кафедры	Подпись зав. кафедрой