



МИНИСТЕРСТВО НАУКИ И ВЫСШЕГО ОБРАЗОВАНИЯ
РОССИЙСКОЙ ФЕДЕРАЦИИ

Федеральное государственное бюджетное образовательное учреждение
высшего образования
«Магнитогорский государственный технический университет им. Г.И.
Носова»



УТВЕРЖДАЮ
Директор ИЭиАС
В.Р. Храмшин

03.02.2026 г.

РАБОЧАЯ ПРОГРАММА ДИСЦИПЛИНЫ (МОДУЛЯ)

ОСНОВЫ ПРОГРАММИРОВАНИЯ НА PYTHON

Направление подготовки (специальность)
22.03.02 Metallurgy

Направленность (профиль/специализация) программы
Информационные технологии в современных литейных процессах

Уровень высшего образования - бакалавриат

Форма обучения
очная

Институт/ факультет	Институт энергетики и автоматизированных систем
Кафедра	Информатики и информационной безопасности
Курс	2
Семестр	4

Магнитогорск
2026 год

Рабочая программа составлена на основе ФГОС ВО - бакалавриат по направлению подготовки 22.03.02 Metallургия (приказ Минобрнауки России от 02.06.2020 г. № 702)

Рабочая программа рассмотрена и одобрена на заседании кафедры информатики и информационной безопасности 22.01.2026, протокол № 5

Зав. кафедрой  И. И. Баранкова

Рабочая программа одобрена методической комиссией
03.02.2026 г. протокол № 5

Председатель  В. Р. Храмшин

Согласовано:

Зав. кафедрой Литейных процессов и материаловедения

 Н.А. Феоктистов

Рабочая программа составлена:
ст. преподаватель кафедры ИиИБ,

 Ю.А. Мазнина

Рецензент:

проректор по цифровизации , канд. техн. наук

 К. А. Рубан

Лист актуализации рабочей программы

Рабочая программа пересмотрена, обсуждена и одобрена для реализации в 2027 - 2028 учебном году на заседании кафедры Информатики и информационной безопасности

Протокол от _____ 20__ г. № ____
Зав. кафедрой _____

Рабочая программа пересмотрена, обсуждена и одобрена для реализации в 2028 - 2029 учебном году на заседании кафедры Информатики и информационной безопасности

Протокол от _____ 20__ г. № ____
Зав. кафедрой _____

Рабочая программа пересмотрена, обсуждена и одобрена для реализации в 2029 - 2030 учебном году на заседании кафедры Информатики и информационной безопасности

Протокол от _____ 20__ г. № ____
Зав. кафедрой _____

Рабочая программа пересмотрена, обсуждена и одобрена для реализации в 2030 - 2031 учебном году на заседании кафедры Информатики и информационной безопасности

Протокол от _____ 20__ г. № ____
Зав. кафедрой _____

1 Цели освоения дисциплины (модуля)

Целью дисциплины «Основы программирования на Python» является изучение основ современных методов и средств программирования высокого уровня и формирование у обучающихся навыков их практического применения на примере языка программирования Python.

Дисциплина «Основы программирования на Python» предусматривает освоение методологий структурного, процедурного и функционального программирования, а также методов тестирования и отладки программ.

2 Место дисциплины (модуля) в структуре образовательной программы

Дисциплина Основы программирования на Python входит в часть учебного плана формируемую участниками образовательных отношений образовательной программы.

Для изучения дисциплины необходимы знания (умения, владения), сформированные в результате изучения дисциплин/ практик:

Цифровая грамотность

Структура и организация корпоративных информационных систем

Знания (умения, владения), полученные при изучении данной дисциплины будут необходимы для изучения дисциплин/практик:

Основы ООП и MVC на Python

Базы данных. SQL-инъекции

Угрозы кибербезопасности

Пентест

Подготовка к сдаче и сдача государственного экзамена

3 Компетенции обучающегося, формируемые в результате освоения дисциплины (модуля) и планируемые результаты обучения

В результате освоения дисциплины (модуля) «Основы программирования на Python» обучающийся должен обладать следующими компетенциями:

Код индикатора	Индикатор достижения компетенции
ДПК-004-2	Способен разрабатывать алгоритмы и компьютерные программы, пригодные для практического применения
ДПК-004-2.1	Применяет основные алгоритмы к решению прикладных программ
ДПК-004-2.2	Использует системы программирования для разработки компьютерных программ

4. Структура, объём и содержание дисциплины (модуля)

Общая трудоемкость дисциплины составляет 3 зачетных единиц 108 академических часов, в том числе:

- контактная работа – 36,1 академических часов;
- аудиторная – 36 академических часов;
- внеаудиторная – 0,1 академических часов;
- самостоятельная работа – 71,9 академических часов;
- в форме практической подготовки – 0 академических часов;

Форма аттестации - зачет

Раздел/ тема дисциплины	Семестр	Аудиторная контактная работа (в академических часах)			Самостоятельная работа студента	Вид самостоятельной работы	Форма текущего контроля успеваемости и промежуточной аттестации	Код компетенции
		Лек.	лаб. зан.	практ. зан.				
1. Введение в Python								
1.1 История развития языков программирования. Компиляторы и интерпретаторы. Инструментальные средства для разработки программного обеспечения. Понятие алгоритма, понятие блок-схемы	4			2	4	Самостоятельное изучение учебной и научной литературы, работа с ЭБС и интернет-источниками. Работа с материалами образовательного портала. Подготовка и выполнение практических работ. Подготовка к тестированию	Практические работы. Тестирование	ДПК-004-2.2
1.2 Знакомство с языком программирования Python: поддерживаемые парадигмы программирования, сферы применения. Интерпретатор Python. Знакомство со средой разработки				1	2	Самостоятельное изучение учебной и научной литературы, работа с ЭБС и интернет-источниками. Работа с материалами образовательного портала. Подготовка и выполнение практических работ. Подготовка к	Практические работы. Тестирование	ДПК-004-2.2

						тестированию		
1.3 Литералы, переменные, типы данных, инструкции, приоритет операторов. Соглашения по оформлению кода PEP8	4			1	2	Самостоятельное изучение учебной и научной литературы, работа с ЭБС и интернет-источниками. Работа с материалами образовательного портала. Подготовка и выполнение практических работ. Подготовка к тестированию	Практические работы. Тестирование	ДПК-004-2.2
1.4 Работа со строками				2	6	Самостоятельное изучение учебной и научной литературы, работа с ЭБС и интернет-источниками. Работа с материалами образовательного портала. Подготовка и выполнение практических работ. Подготовка к тестированию	Практические работы. Тестирование	ДПК-004-2.2
Итого по разделу				6	14			
2. Основные синтаксические конструкции языка программирования Python								
2.1 Стандартные библиотеки Python для вычислений: math, random	4			2	4	Самостоятельное изучение учебной и научной литературы, работа с ЭБС и интернет-источниками. Работа с материалами образовательного портала. Подготовка и выполнение практических работ. Подготовка к аудиторной контрольной работе (рубежному	Практические работы. Аудиторная контрольная работа (рубежный контроль)	ДПК-004-2.2

						контролю)		
2.2 Операторы ветвлений. Оператор match. Тернарный оператор.				2	4	Самостоятельное изучение учебной и научной литературы, работа с ЭБС и интернет-источниками. Работа с материалами образовательного портала. Подготовка и выполнение практических работ. Подготовка к аудиторной контрольной работе (рубежному контролю)	Практические работы. Аудиторная контрольная работа (рубежный контроль)	ДПК-004-2.2
2.3 Циклы	4			4	8	Самостоятельное изучение учебной и научной литературы, работа с ЭБС и интернет-источниками. Работа с материалами образовательного портала. Подготовка и выполнение практических работ. Подготовка к аудиторной контрольной работе (рубежному контролю)	Практические работы. Аудиторная контрольная работа (рубежный контроль)	ДПК-004-2.2
2.4 Функции				4	8	Самостоятельное изучение учебной и научной литературы, работа с ЭБС и интернет-источниками. Работа с материалами образовательного портала. Подготовка и выполнение практических работ. Подготовка к аудиторной	Практические работы. Аудиторная контрольная работа (рубежный контроль)	ДПК-004-2.2

						контрольной работе (рубежному контролю)		
Итого по разделу				12	24			
3. Коллекции в языке программирования Python: списки, кортежи, множества, словари								
3.1 Списки. Алгоритмы поиска и сортировки	4			4	8	Самостоятельное изучение учебной и научной литературы, работа с ЭБС и интернет-источниками. Работа с материалами образовательного портала. Подготовка и выполнение практических работ. Подготовка к аудиторной контрольной работе (рубежному контролю)	Практические работы. Аудиторная контрольная работа (рубежный контроль)	ДПК-004-2.1, ДПК-004-2.2
3.2 Кортежи				1	2	Самостоятельное изучение учебной и научной литературы, работа с ЭБС и интернет-источниками. Работа с материалами образовательного портала. Подготовка и выполнение практических работ. Подготовка к аудиторной контрольной работе (рубежному контролю)	Практические работы. Аудиторная контрольная работа (рубежный контроль)	ДПК-004-2.1, ДПК-004-2.2
3.3 Множества				1	2	Самостоятельное изучение учебной и научной литературы, работа с ЭБС и интернет-источниками. Работа с	Практические работы. Аудиторная контрольная работа (рубежный контроль)	ДПК-004-2.1, ДПК-004-2.2

						материалами образовательного портала. Подготовка и выполнение практических работ. Подготовка к аудиторной контрольной работе (рубежному контролю)		
3.4 Словари	4			4	8	Самостоятельное изучение учебной и научной литературы, работа с ЭБС и интернет-источниками. Работа с материалами образовательного портала. Подготовка и выполнение практических работ. Подготовка к аудиторной контрольной работе (рубежному контролю)	Практические работы. Аудиторная контрольная работа (рубежный контроль)	ДПК-004-2.1, ДПК-004-2.2
Итого по разделу				10	20			
4. Работа с файлами в языке программирования Python								
4.1 Принципы работы с текстовыми и бинарными файлами	4			2	4	Самостоятельное изучение учебной и научной литературы, работа с ЭБС и интернет-источниками. Работа с материалами образовательного портала. Подготовка и выполнение практических работ. Подготовка к аудиторной контрольной работе (рубежному контролю)	Практические работы. Аудиторная контрольная работа (рубежный контроль)	ДПК-004-2.1, ДПК-004-2.2
4.2 Стандартные				2	4	Самостоятельно	Практические	ДПК-004-

библиотеки Python для работы с популярными форматами файлов (csv, json, xml)						е изучение учебной и научной литературы, работа с ЭБС и интернет-источниками. Работа с материалами образовательного портала. Подготовка и выполнение практических работ. Выполнение индивидуального домашнего задания (рубежному контролю)	работы. Индивидуальное домашнее задание (рубежный контроль)	2.1, ДПК-004-2.2
Итого по разделу				4	8			
5. Прочие возможности языка программирования Python								
5.1 Стандартные библиотеки Python для работы с датами и временем: datetime, time	4			2	1	Самостоятельное изучение учебной и научной литературы, работа с ЭБС и интернет-источниками. Работа с материалами образовательного портала. Подготовка и выполнение практических работ. Подготовка к тестированию	Практические работы. Тестирование	ДПК-004-2.1, ДПК-004-2.2
5.2 Стандартные библиотеки Python для работы с сервисами операционной системы: os, sys. Работа zip-архивами				2	1	Самостоятельное изучение учебной и научной литературы, работа с ЭБС и интернет-источниками. Работа с материалами образовательного портала. Подготовка и выполнение практических работ. Подготовка к тестированию	Практические работы. Тестирование	ДПК-004-2.1, ДПК-004-2.2
Итого по разделу				4	2			
6. Аттестация								

6.1 Подготовка к зачету	4				3,9	Самостоятельное изучение учебной и научной литературы, работа с ЭБС и интернет-источниками. Работа с материалами образовательного портала. Подготовка к зачету	Зачет	ДПК-004-2.1, ДПК-004-2.2
Итого по разделу					3,9			
Итого за семестр			36	71,9			зачёт	
Итого по дисциплине			36	71,9			зачет	

5 Образовательные технологии

Для реализации предусмотренных видов учебной работы в качестве образовательных технологий в преподавании дисциплины «Основы программирования на Python» используются традиционная и модульно-компетентностная технологии.

Реализация компетентностного подхода предусматривает использование в учебном процессе активных и интерактивных форм проведения занятий в сочетании с внеаудиторной работой с целью формирования и развития профессиональных навыков обучающихся.

При проведении учебных занятий преподаватель обеспечивает развитие у обучающихся навыков командной работы, межличностной коммуникации, принятия решений, лидерских качеств посредством проведения интерактивных лекций, групповых дискуссий, ролевых игр, тренингов, анализа ситуаций, учета особенностей профессиональной деятельности выпускников и потребностей работодателей.

Формы учебных занятий с использованием традиционных технологий:

- обзорные лекции – для рассмотрения общих вопросов информатики и информационных технологий, для систематизации и закрепления знаний;
- информационные – для ознакомления с техническими средствами реализации информационных процессов, со стандартами организации сетей, основными приемами защиты информации, и другой справочной информацией;
- лекции-визуализации – для наглядного представления способов решения алгоритмических и функциональных задач, визуализации результатов решения задач;
- семинар – практическое занятие, посвященное освоению конкретных умений и навыков по предложенному алгоритму.

Формы учебных занятий с использованием технологий проблемного обучения:

- проблемная лекция – изложение материала, предполагающее постановку проблемных и дискуссионных вопросов, освещение различных научных подходов, авторские комментарии, связанные с различными моделями интерпретации изучаемого материала (для развития исследовательских навыков и изучения способов решения задач);
- лекции с заранее запланированными ошибками – направленные на поиск обучающимися синтаксических и алгоритмических ошибок при решении алгоритмических и функциональных задач, с последующей диагностикой слушателей и разбором сделанных ошибок;
- практическое занятие в форме практикума – организация учебной работы, направленная на решение комплексной учебно-познавательной задачи, требующей от обучающегося применения как научно-теоретических знаний, так и практических навыков;
- практическое занятие на основе кейс-метода – обучение в контексте моделируемой ситуации, воспроизводящей реальные условия научной, производственной, общественной деятельности; обучающиеся должны проанализировать ситуацию, разобраться в сути проблем, предложить возможные решения и выбрать лучшее из них; кейсы базируются на реальном фактическом материале или же приближены к реальной ситуации;
- подготовка тематических рефератов, содержащих разделы, частично или полностью выносимые на самостоятельное изучение.

Формы учебных занятий с использованием игровых технологий:

- учебная игра – форма воссоздания предметного и социального содержания будущей профессиональной деятельности специалиста, моделирования таких систем

отношений, которые характерны для этой деятельности как целого;

– деловая игра – моделирование различных ситуаций, связанных с выработкой и принятием совместных решений, обсуждением вопросов в режиме «мозгового штурма», реконструкцией функционального взаимодействия в коллективе и т.п.

Технологии проектного обучения:

– творческий проект – учебно-познавательная деятельность обучающихся осуществляется в рамках рамочного задания, подчиняясь логике и интересам участников проекта, жанру конечного результата (газета, фильм, праздник, издание, экскурсия, подготовка заданий конкурсов и т.п.);

– информационный проект – учебно-познавательная деятельность с ярко выраженной эвристической направленностью (поиск, отбор и систематизация информации о каком-то объекте, ознакомление участников проекта с этой информацией, ее анализ и обобщение для презентации более широкой аудитории).

6 Учебно-методическое обеспечение самостоятельной работы обучающихся

Представлено в приложении 1.

7 Оценочные средства для проведения промежуточной аттестации

Представлены в приложении 2.

8 Учебно-методическое и информационное обеспечение дисциплины

а) Основная литература:

1. Гагарина, Л. Г. Основы проектирования и разработки информационных систем : учебное пособие / Л.Г. Гагарина, Ю.С. Шевнина. — Москва : ИНФРА-М, 2024. — 211 с. — (Высшее образование). — DOI 10.12737/1872684. - ISBN 978-5-16-017759-5. - Текст : электронный. - URL: <https://znanium.ru/catalog/product/1872684> (дата обращения: 10.03.2026). – Режим доступа: по подписке.

2. Гниденко, И. Г. Технологии и методы программирования : учебник для вузов / И. Г. Гниденко, Ф. Ф. Павлов, Д. Ю. Федоров. — 2-е изд., перераб. и доп. — Москва : Издательство Юрайт, 2026. — 241 с. — (Высшее образование). — ISBN 978-5-534-18130-2. — Текст : электронный // Образовательная платформа Юрайт [сайт]. — URL: <https://urait.ru/bcode/583784> (дата обращения: 10.03.2026).

3. Карякин, М. И. Технологии программирования и компьютерный практикум на языке Python : учебное пособие / М. И. Карякин, К. А. Ватульян, Р. М. Мнухин ; Южный федеральный университет. - Ростов-на-Дону ; Таганрог : Издательство Южного федерального университета, 2022. - 242 с. - ISBN 978-5-9275-4108-9. - Текст : электронный. - URL: <https://znanium.com/catalog/product/2057604> (дата обращения: 10.03.2025). – Режим доступа: по подписке.

4. Конструирование программного обеспечения : учебное пособие / под ред. Л.Г. Гагариной. — Москва : ИНФРА-М, 2024. — 319 с. — (Высшее образование). — DOI 10.12737/1893880. - ISBN 978-5-16-017861-5. - Текст : электронный. - URL: <https://znanium.com/catalog/product/1893880> (дата обращения: 10.03.2026). – Режим доступа: по подписке

5. Лаврищева, Е. М. Программная инженерия и технологии программирования сложных систем : учебник для вузов / Е. М. Лаврищева. — 2-е изд., испр. и доп. — Москва : Издательство Юрайт, 2025. — 432 с. — (Высшее образование). — ISBN 978-5-534-07604-2. — Текст : электронный // Образовательная платформа Юрайт [сайт]. — URL: <https://urait.ru/bcode/561885> (дата обращения: 10.03.2026).

6. Федоров, Д. Ю. Программирование на Python : учебное пособие для вузов / Д. Ю. Федоров. — 6-е изд., перераб. и доп. — Москва : Издательство Юрайт, 2026. — 187 с. — (Высшее образование). — ISBN 978-5-534-19666-5. — Текст : электронный // Образовательная платформа Юрайт [сайт]. — URL: <https://urait.ru/bcode/585806> (дата

обращения: 10.03.2026).

7. Черткова, Е. А. Компьютерные технологии обучения : учебник для вузов / Е. А. Черткова. — 3-е изд., испр. и доп. — Москва : Издательство Юрайт, 2026. — 245 с. — (Высшее образование). — ISBN 978-5-534-12532-0. — Текст : электронный // Образовательная платформа Юрайт [сайт]. — URL: <https://urait.ru/bcode/584740> (дата обращения: 10.03.2026).

б) Дополнительная литература:

1. Гагарина, Л. Г. Технология разработки программного обеспечения : учебное пособие / Л.Г. Гагарина, Е.В. Кокорева, Б.Д. Сидорова-Виснадул ; под ред. Л.Г. Гагариной. — Москва : ФОРУМ : ИНФРА-М, 2024. — 400 с. — (Высшее образование). - ISBN 978-5-8199-0707-8. - Текст : электронный. - URL: <https://znanium.ru/catalog/product/2149181> (дата обращения: 10.03.2026). – Режим доступа: по подписке.

2. Гуриков, С. Р. Основы алгоритмизации и программирования на Python : учебное пособие / С.Р. Гуриков. — Москва : ИНФРА-М, 2026. — 343 с. — (Среднее профессиональное образование). - ISBN 978-5-16-016906-4. - Текст : электронный. - URL: <https://znanium.ru/catalog/product/2216924> (дата обращения: 10.03.2026). – Режим доступа: по подписке

3. Жуков, Р. А. Язык программирования Python. Практикум : учебное пособие / Р.А. Жуков. — Москва : ИНФРА-М, 2026. — 216 с. + Доп. материалы [Электронный ресурс]. — (Среднее профессиональное образование). - ISBN 978-5-16-015638-5. - Текст : электронный. - URL: <https://znanium.ru/catalog/product/2216925> (дата обращения: 10.03.2026). – Режим доступа: по подписке.

4. Зыков, С. В. Объектно-ориентированное программирование : учебник и практикум для вузов / С. В. Зыков. — 2-е изд. — Москва : Издательство Юрайт, 2026. — 151 с. — (Высшее образование). — ISBN 978-5-534-16941-6. — Текст : электронный // Образовательная платформа Юрайт [сайт]. — URL: <https://urait.ru/bcode/584131> (дата обращения: 10.03.2026).

5. Зыков, С. В. Программирование. Функциональный подход : учебник и практикум для вузов / С. В. Зыков. — 2-е изд. — Москва : Издательство Юрайт, 2026. — 150 с. — (Высшее образование). — ISBN 978-5-534-16942-3. — Текст : электронный // Образовательная платформа Юрайт [сайт]. — URL: <https://urait.ru/bcode/584399> (дата обращения: 10.03.2026).

6. Чернышев, С. А. 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> (дата обращения: 10.03.2026).

МАКРООБЪЕКТЫ:

1. Баранкова И. И. Определение критически значимых ресурсов объекта защиты при составлении модели угроз информационной безопасности : учебное пособие / И. И. Баранкова, О. В. Пермякова ; МГТУ. - Магнитогорск : МГТУ, 2017. - 1 электрон. опт. диск (CD-ROM). - URL: <https://host.megaprolib.net/MP0109/Download/MObject/1858>. - ISBN 978-5-9967-1031-7. - Текст : электронный. - дата обращения: 10.03.2026.

2. Григоренко Л. А. Основы программирования на Python: учебное пособие [для вузов] / Л. А. Григоренко, Ю. А. Мазнина, А. В. Перминова; Магнитогорский гос. технический ун-т им. Г. И. Носова. - Магнитогорск : МГТУ им. Г. И. Носова, 2023. - 1 CD-ROM. - Загл. с титул. экрана. - URL: <https://host.megaprolib.net/MP0109/Download/MObject/21033>. - ISBN 978-5-9967-2905-0. - Текст : электронный. - дата обращения: 10.03.2026.

в) Методические указания:

Методические указания по выполнению практических работ представлены в приложении 3.

Методические указания по выполнению внеаудиторных самостоятельных работ представлены в приложении 4.

г) Программное обеспечение и Интернет-ресурсы:**Программное обеспечение**

Наименование ПО	№ договора	Срок действия лицензии
MS Office 2007 Professional	№ 135 от 17.09.2007	бессрочно
7Zip	свободно распространяемое ПО	бессрочно
Anaconda Python	свободно распространяемое ПО	бессрочно
MS SQL Server Management Studio	свободно распространяемое ПО	бессрочно
MS Visual Studio 2017 Community Edition	свободно распространяемое ПО	бессрочно
Adobe Reader	свободно распространяемое ПО	бессрочно
Браузер Mozilla Firefox	свободно распространяемое ПО	бессрочно
Браузер Yandex	свободно распространяемое ПО	бессрочно
MS Office 2003 Professional	№ 135 от 17.09.2007	бессрочно
Git	свободно распространяемое ПО	бессрочно
MS Visual Studio Code	свободно распространяемое ПО	бессрочно
JetBrains PyCharm Community Edition	свободно распространяемое ПО	бессрочно
NotePad++	свободно распространяемое ПО	бессрочно
FAR Manager	свободно распространяемое ПО	бессрочно
Double Commander	свободно распространяемое ПО	бессрочно
Atom Editor	свободно распространяемое ПО	бессрочно

Профессиональные базы данных и информационные справочные системы

Название курса	Ссылка
Электронная база периодических изданий ООО «ИВИС»	https://eivis.ru/
Российская Государственная библиотека. Каталоги	https://www.rsl.ru/ru/4readers/catalogues/
Электронные ресурсы библиотеки МГТУ им. Г.И. Носова	https://host.megaprolib.net/M/P0109/Web
Международная реферативная и полнотекстовая справочная база данных научных изданий «Springer Nature»	https://www.nature.com/siteindex

Национальная информационно-аналитическая система – Российский индекс научного цитирования (РИНЦ)	URL: https://elibrary.ru/project_risc.asp
--	---

9 Материально-техническое обеспечение дисциплины (модуля)

Материально-техническое обеспечение дисциплины включает:

Лекционные аудитории:

- Мультимедийные средства хранения, передачи и представления информации.

Учебные аудитории для проведения практических занятий, групповых и индивидуальных консультаций, текущего контроля и промежуточной аттестации:

- Персональные компьютеры с ПО, выходом в Интернет и с доступом в электронную информационно-образовательную среду университета.

Помещения для самостоятельной работы обучающихся:

- Персональные компьютеры с ПО, выходом в Интернет и с доступом в электронную информационно-образовательную среду университета.

Лекционные аудитории: ауд. 2124, ауд. 2122, 2113, 226, 238, 365, 388, 433 и др. (мультимедийные аудитории).

Учебные аудитории для проведения практических занятий, групповых и индивидуальных консультаций, текущего контроля и промежуточной аттестации: ауд. 245, 247, 249, 279, 372, 394 и др. (компьютерные классы)

Аудитории для самостоятельной работы: ауд. 132а, компьютерные классы, читальные залы библиотеки.

Учебно-методическое обеспечение самостоятельной работы обучающихся

Аудиторная самостоятельная работа обучающихся на практических занятиях осуществляется под контролем преподавателя в виде решения задач и выполнения упражнений, которые определяет преподаватель для обучающегося с использованием методов ИТ.

Внеаудиторная самостоятельная работа обучающихся осуществляется в виде чтения литературы по соответствующему разделу с проработкой материала и выполнения домашних заданий с консультациями преподавателя, а так же с применением кейс-технологий.

Вопросы для подготовки к зачету и текущему контролю

1. Особенности языка программирования Python. Соглашения по оформлению кода PEP8, организация кода в модели и пакеты.
2. Язык программирования Python: числовые, булевы и строковые литералы, основные операции над простыми типами.
3. Особенности целочисленного деления в языке Python.
4. Язык программирования Python: синтаксические конструкции, условия, циклы, функция range().
5. Тернарный оператор условия, его преимущества и недостатки.
6. Язык программирования Python: системные библиотеки Python: math, time, random, os.
7. Коллекции в языке программирования Python: строки.
8. Отрицательная индексация при работе со строками.
9. Коллекции в языке программирования Python: списки, кортежи, множества.
10. Методы списков и операции со списками. Срезы списков. Присваивание в срез. Копирование списка.
11. В чем преимущество встроеного типа кортеж перед встроеным типом список? Примеры применения кортежей в языке программирования Python.
12. Коллекции в языке программирования Python: словари
13. Функции в языке программирования Python. Область видимости переменных.
14. Именованные параметры функций. Значения параметров по умолчанию.
15. Функции с переменным числом аргументов.
16. Рекурсия.
17. Принципы работы с текстовыми и бинарными файлами.
18. Работа с файловой системой и популярными форматами файлов: csv формат.
19. Работа с файловой системой и популярными форматами файлов: json формат.
20. Работа с файловой системой и популярными форматами файлов: xml формат.
21. Стандартные библиотеки Python для работы с датами и временем: datetime.
22. Стандартные библиотеки Python для работы с датами и временем: time.
23. Стандартные библиотеки Python для работы с файловой системой и сервисами операционной системы: os, sys.
24. Работа с zip-архивами средствами языка программирования Python.

Примеры индивидуальных домашних заданий (заданий рубежного контроля)

1. Имя файла задается в консоли. Проведите анализ текста из файла: подсчитайте общее количество символов в тексте, количество символов без пробелов, количество слов и количество предложений в тексте. Предложения могут заканчиваться точкой, вопросительным, восклицательным знаком или троеточием. Также возможен случай, что предложение оборвалось с концом файла без заключительного знака препинания. Если

файл отсутствует, выведите сообщение «Ошибка при открытии файла».

2. Дан файл с экспортом сданных решений в системе ejudge. Контекст проводился в формате личной олимпиады. Вам понадобятся следующие поля:

- User_Id – уникальный числовой идентификатор пользователя;
- User_Login – уникальный логин пользователя (также можно использовать, как уникальный числовой идентификатор);
- User_Inv – у скрытого пользователя в этом поле записан символ "I";
- Lang – идентификатор языка программирования.
- Score – Количество баллов, полученных за задачу.

Посчитайте общее число участников конкурса, а также то, сколько участников использовало тот или иной язык программирования. Считается, что участник использует язык программирования, если он сдал хотя бы одно решение, набравшее больше 0 баллов. Скрытых пользователей учитывать не нужно.

Выведите общее число пользователей (Total) в первой строке, считаются только участники, сдавшие хотя бы одну задачу на ненулевой балл. После слова Total или названия языка программирования через двоеточие и пробел выведите число участников.

В последующих строках выведите все использованные языки программирования, список упорядочите по убыванию числа участников, использующих данный язык программирования, при равных значениях – в лексикографическом порядке названия языка программирования. Пример результата работы программы:

```
Total: 19  
python3: 19  
C++: 5
```

Примеры заданий аудиторных контрольных работ (рубежного контроля)

1. Программа получает на вход в одной строке элементы списка – целые числа, разделенные пробелом. Количество элементов произвольное. На следующей строке вводится одно число g – значение поиска. Реализуйте линейный алгоритм поиска введенного значения g . В случае успеха выведите порядковый номер (индекс) первого найденного элемента в списке при условии, что индексация начинается с единицы. Если данный элемент отсутствует, необходимо вывести строку `ErrorValue`.

2. Реализуйте алгоритм Евклида для нахождения наибольшего общего делителя с помощью рекурсивной функции. Функция должна иметь два аргумента.

Оценочные средства для проведения промежуточной аттестации

а) Планируемые результаты обучения и оценочные средства для проведения промежуточной аттестации

Код индикатора	Индикатор достижения компетенции	Оценочные средства
<p>ДПК-004-2: Способен разрабатывать алгоритмы и компьютерные программы, пригодные для практического применения</p> <ul style="list-style-type: none"> - ДПК-004-2.1: Применяет основные алгоритмы к решению прикладных программ - ДПК-004-2.2: Использует системы программирования для разработки компьютерных программ 		
<p>ДПК-004-2.1</p>	<p>Применяет основные алгоритмы к решению прикладных программ</p>	<p>Примеры практических заданий для зачета:</p> <ol style="list-style-type: none"> 1. Электричка отправляется в h1:m1 и едет h2:m2. Выведите время прибытия электрички на электронных часах в формате hh:mm. 2. Напишите программу, которая находит рекордное количество вхождений (не обязательно подряд) символа в строку. 3. На языке программирования Python реализуйте вычисление факториала через лямбда-функцию. <p>Теоретические вопросы к зачету:</p> <ol style="list-style-type: none"> 1. Особенности языка программирования Python. Соглашения по оформлению кода pep8, организация кода в модели и пакеты. 1. Язык программирования Python: числовые, булевы и строковые литералы, основные операции над простыми типами. 2. Особенности целочисленного деления в языке Python. 3. Язык программирования Python: синтаксические конструкции, условия, циклы, функция range(). 4. Тернарный оператор условия, его преимущества и недостатки. 2. Язык программирования Python: системные библиотеки Python: math, time, random, os. 3. Коллекции в языке программирования Python: строки. 5. Отрицательная индексация при работе со строками. 6. Коллекции в языке программирования Python: списки, кортежи, множества. 7. Методы списков и операции со списками. Срезы списков. Присваивание в срез. Копирование списка. 8. В чем преимущество встроенного типа кортеж перед встроенным типом список? Примеры применения кортежей в языке программирования Python. 9. Коллекции в языке программирования Python: словари

Код индикатора	Индикатор достижения компетенции	Оценочные средства
		<p>10. Функции в языке программирования Python. Область видимости переменных.</p> <p>11. Именованные параметры функций. Значения параметров по умолчанию.</p> <p>12. Функции с переменным числом аргументов.</p> <p>13. Рекурсия.</p>
ДПК-004-2.2	Использует системы программирования для разработки компьютерных программ	<p>Примеры практических заданий для зачета:</p> <p>1. Создайте текстовый файл, в котором на каждой строке написана дата в формате dd/mm/yyyy. Используя язык Python, в другой файл выведите даты в порядке возрастания в формате yyyy/mm/dd.</p> <p>2. В csv-файле даны даны рост и вес нескольких человек: 164;71.5 181;78.3 151;52.8</p> <p>Вычислите средний рост и вес и сохраните результаты вычислений в файл формата json.</p> <p>Теоретические вопросы к зачету:</p> <p>1. Принципы работы с текстовыми и бинарными файлами.</p> <p>2. Стандартные библиотеки Python для работы с файловой системой: os, sys.</p> <p>3. Работа с файловой системой и популярными форматами файлов: csv-формат.</p> <p>4. Работа с файловой системой и популярными форматами файлов: json-формат.</p> <p>5. Работа с файловой системой и популярными форматами файлов: zip-архивы.</p> <p>6. Стандартные библиотеки Python для работы с датами и временем: datetime.</p> <p>7. Стандартные библиотеки Python для работы с датами и временем: time.</p> <p>8. Стандартные библиотеки Python для работы с сервисами операционной системы: os, sys.</p>

б) Примерный перечень индивидуальных заданий (заданий рубежного контроля)

1. Имя файла задается в консоли. Проведите анализ текста из файла: подсчитайте общее количество символов в тексте, количество символов без пробелов, количество слов и количество предложений в тексте. Предложения могут заканчиваться точкой, вопросительным, восклицательным знаком или троеточием. Также возможен случай, что предложение оборвалось с концом файла без заключительного знака препинания. Если файл отсутствует, выведите сообщение «Ошибка при открытии файла».

2. Дан файл с экспортом сданных решений в системе ejudge. Контест проводился в формате личной олимпиады. Вам понадобятся следующие поля:

- User_Id – уникальный числовой идентификатор пользователя;
- User_Login – уникальный логин пользователя (также можно использовать, как

уникальный числовой идентификатор);

- User_Inv – у скрытого пользователя в этом поле записан символ "I";
- Lang – идентификатор языка программирования.
- Score – Количество баллов, полученных за задачу.

Посчитайте общее число участников конкурса, а также то, сколько участников использовало тот или иной язык программирования. Считается, что участник использует язык программирования, если он сдал хотя бы одно решение, набравшее больше 0 баллов. Скрытых пользователей учитывать не нужно.

Выведите общее число пользователей (Total) в первой строке, считаются только участники, сдавшие хотя бы одну задачу на ненулевой балл. После слова Total или названия языка программирования через двоеточие и пробел выведите число участников.

В последующих строках выведите все использованные языки программирования, Список упорядочите по убыванию числа участников, использующих данный язык программирования, при равных значениях – в лексикографическом порядке названия языка программирования. Пример результата работы программы:

Total: 19

python3: 19

C++: 5

в) Порядок проведения промежуточной аттестации, показатели и критерии оценивания

Промежуточная аттестация по дисциплине включает теоретические вопросы, позволяющие оценить уровень усвоения обучающимися знаний, и практические задания, выявляющие степень сформированности умений и навыков, проводится в форме зачета.

Показатели и критерии оценивания зачета:

– на оценку «зачтено» – обучающийся должен успешно пройти запланированные рубежные контроли и показать пороговый уровень знаний на уровне воспроизведения и объяснения информации;

– на оценку «не зачтено» – обучающийся не прошел запланированные рубежные контроли и не может показать знания на уровне воспроизведения и объяснения информации.

Методические указания по выполнению практических работ

Практические работы проводятся в компьютерных классах целью получения практических умений для формирования и развития профессиональных навыков и соответствующих компетенций по дисциплине. При подготовке к выполнению заданий практической работы используйте лекции, справочный материал программного обеспечения, рекомендованную литературу и цифровые образовательные ресурсы соответствующих методических материалов, размещенных в сети Интернет или локальной сети университета. Перед выполнением практической работы необходимо получить свой вариант индивидуального задания у преподавателя. Прежде чем приступить к выполнению практической работы, внимательно прочтите рекомендации к ее выполнению. Ознакомьтесь с перечнем рекомендуемой литературы, повторите теоретический материал, относящийся к теме работы. Ответьте на контрольные вопросы, выполните задания для самостоятельного выполнения. По результатам практической работы предоставляется отчет. Отчет к практическим работам должен содержать:

- название практической работы;
- цель и задачи работы;
- краткие теоретические сведения;
- задания по практической работе;
- ход работы - описание последовательности действий при выполнении работы;
- выводы или результаты.

Результаты выполнения практической работы могут быть представлены в электронном варианте или распечатанные. Результаты выполнения заданий практической работы можно сохранить на образовательном портале в личном кабинете и использовать при подготовке к экзамену.

Защита работы и результаты оценивания

Защита проводится в два этапа:

1. Демонстрируются результаты выполнения задания. В случае выполнения практической работы, предусматривающей разработку программы, при помощи тестового примера доказывается, что результат, получаемый при выполнении программы, является правильным.

2. Для защиты работы студенту необходимо ответить на дополнительные вопросы преподавателя. Каждая практическая работа оценивается определенным количеством баллов исходя из 5-бальной системы оценок.

Практическая работа считается выполненной и защищенной, если выполнены все задания и даны правильные ответы преподавателю на заданные вопросы. Практическая работа считается выполненной и незащищенной, если выполнены все задания, но полученные результаты являются неверными или не даны правильные ответы преподавателю на заданные вопросы и ответы были не полные. Обучающемуся, не выполнившему в полном объеме все задания практической работы, или пропустившему по уважительной причине практическую работу, необходимо выполнить ее самостоятельно в компьютерном классе, результаты выполненной работы сохранить на съемном накопителе или на образовательном портале. Результаты предоставить в сроки, указанные преподавателем вместе с отчетом, демонстрацией полученных результатов в компьютерном классе или предоставлением материалов на электронном образовательном ресурсе.

Правила по технике безопасности для обучающихся при проведении практических работ

1. Практические работы проводятся под наблюдением преподавателя. К выполнению практических работ студенты допускаются только после прослушивания инструктажа по технике безопасности и противопожарным мерам.

2. Обучающийся должен строго выполнять правила техники безопасности и санитарно-гигиенические нормы при работе в компьютерных классах или специализированных лабораториях университета.

Методические указания по выполнению внеаудиторных самостоятельных работ

Общие положения

Настоящие методические указания предназначены для организации внеаудиторной самостоятельной работы студентов и оказания помощи в самостоятельном изучении теоретического и реализации компетенций обучаемых.

Данные методические указания не являются учебным пособием, поэтому перед началом выполнения самостоятельного задания следует изучить соответствующие разделы лекционных занятий, материалов образовательного портала, разделов основной и дополнительной литературы, представленных в пункте 8. «Учебно-методическое и информационное обеспечение дисциплины (модуля)» данной РПД.

Цели и задачи самостоятельной работы

Цель самостоятельной работы – содействие оптимальному усвоению учебного материала обучающимися, развитие их познавательной активности, готовности и потребности в самообразовании.

Задачи самостоятельной работы:

- повышение исходного уровня владения информационными технологиями;
- углубление и систематизация знаний;
- постановка и решение стандартных задач профессиональной деятельности;
- развитие работы с различной по объему и виду информацией, учебной и научной литературой;
- практическое применение знаний, умений;
- самостоятельно использование стандартных программных средств сбора, обработки, хранения и защиты информации
- развитие навыков организации самостоятельного учебного труда и контроля за его эффективностью.

Виды внеаудиторной самостоятельной работы и формы контроля и время на выполнение каждого вида самостоятельной работы указаны в пункте 4. «Структура и содержание дисциплины» данной РПД.

Порядок выполнения

При выполнении текущей внеаудиторной самостоятельной работы обучающемуся следует придерживаться следующего порядка действий:

- 1) внимательно изучить соответствующие теоретические разделы дисциплины, пользуясь материалами (лекционными, презентационными, аудио-визуальными):
 - предоставляемыми преподавателем на лекционных занятиях;
 - предоставляемыми преподавателем в рамках электронных образовательных курсов;
 - содержащимися в учебниках и учебных пособиях ЭБС (электронно-библиотечных систем), электронных каталогов университета и интернет-ресурсов.
- 2) подробно разобрать типовые примеры решения задач, рассмотренные в рамках аудиторной контактной работы с преподавателем;
- 3) применить полученные теоретические знания и практические навыки к решению индивидуальных заданий, к прохождению компьютерных тестирований;
- 4) при необходимости, сформировать перечень вопросов, вызвавших затруднения в процессе самостоятельной работы. Обсудить возникшие вопросы со студентами группы, в рамках командно-проектной работы, и с преподавателем, в рамках консультационной помощи, реализованной либо в контактной форме, либо средствами

информационно-образовательной среды ВУЗа.

Критерии оценки внеаудиторных самостоятельных работ

Качество выполнения внеаудиторной самостоятельной работы обучающихся оценивается посредством текущего контроля самостоятельной работы обучающихся с использованием балльно-рейтинговой системы.

В качестве форм текущего контроля по дисциплине используются: индивидуальные задания, аудиторские контрольные работы, компьютерное тестирование.

Максимальное количество баллов обучающийся получает, если:

- выполняет индивидуальные задания в соответствии со всеми заявленными требованиями;

- дает правильные формулировки, точные определения, понятия терминов;

- может обосновать рациональность решения текущей задачи.;

- обстоятельно с достаточной полнотой излагает соответствующую теоретический раздел;

- правильно отвечает на дополнительные вопросы преподавателя, имеющие целью выяснить степень понимания им данного материала.

50~85% от максимального количества баллов обучающийся получает, если:

- неполно (не менее 70% от полного), но правильно выполнено задание;

- при изложении были допущены 1-2 несущественные ошибки, которые он исправляет после замечания преподавателя;

- дает правильные формулировки, точные определения, понятия терминов;

- может обосновать свой ответ, привести необходимые примеры;

- правильно отвечает на дополнительные вопросы преподавателя, имеющие целью выяснить степень понимания им данного материала.

36~50% от максимального количества баллов обучающийся получает, если:

- неполно (не менее 50% от полного), но правильно изложено задание;

- при изложении была допущена 1 существенная ошибка;

- знает и понимает основные положения данной темы, но допускает неточности в формулировке понятий;

- излагает выполнение задания недостаточно логично и последовательно;

- затрудняется при ответах на вопросы преподавателя.

35% и менее от максимального количества баллов обучающийся получает, если:

- неполно (менее 50% от полного) изложено задание;

- при изложении были допущены существенные ошибки. В "0" баллов преподаватель вправе оценить выполненное обучающимся задание, если оно не удовлетворяет требованиям, установленным преподавателем к данному виду работы или не было представлено для проверки.

Сумма полученных баллов по всем видам заданий внеаудиторной самостоятельной работы составляет рейтинговый показатель обучающегося. Рейтинговый показатель обучающегося влияет на выставление итоговой оценки по результатам изучения дисциплины.

Показатели и критерии оценивания полученных знаний представлены в приложении 2 данной РПД.