

**ОЦЕНОЧНЫЕ МАТЕРИАЛЫ
ПО ДИСЦИПЛИНЕ**
Технологии программирования

Код модуля
1153109(2)

Модуль
Основы программирования

Екатеринбург

Оценочные материалы составлены автором(ами):

№ п/п	Фамилия, имя, отчество	Ученая степень, ученое звание	Должность	Подразделение
1	Созькин Андрей Владимирович	кандидат технических наук, без ученого звания	Доцент	информационных технологий и систем управления
2	Степура Юлия Владимировна		Старший преподаватель	Интеллектуальных информационных технологий
3	Шадрин Денис Борисович		Старший преподаватель	Кафедра интеллектуальных информационных технологий

Согласовано:

Управление образовательных программ

Т.Г. Комарова

Авторы:

- Созыкин Андрей Владимирович, Доцент, информационных технологий и систем управления
- Степура Юлия Владимировна, Старший преподаватель, Интеллектуальных информационных технологий
- Шадрин Денис Борисович, Старший преподаватель, Кафедра интеллектуальных информационных технологий

1. СТРУКТУРА И ОБЪЕМ ДИСЦИПЛИНЫ Технологии программирования

1.	Объем дисциплины в зачетных единицах	4	
2.	Виды аудиторных занятий	Лекции Лабораторные занятия	
3.	Промежуточная аттестация	Экзамен	
4.	Текущая аттестация	Контрольная работа	1
		Домашняя работа	1

2. ПЛАНИРУЕМЫЕ РЕЗУЛЬТАТЫ ОБУЧЕНИЯ (ИНДИКАТОРЫ) ПО ДИСЦИПЛИНЕ МОДУЛЯ Технологии программирования

Индикатор – это признак / сигнал/ маркер, который показывает, на каком уровне обучающийся должен освоить результаты обучения и их предъявление должно подтвердить факт освоения предметного содержания данной дисциплины, указанного в табл. 1.3 РПМ-РПД.

Таблица 1

Код и наименование компетенции	Планируемые результаты обучения (индикаторы)	Контрольно-оценочные средства для оценивания достижения результата обучения по дисциплине
1	2	3
ПК-1 -Способен проектировать, разрабатывать, интегрировать, проверять на работоспособность программное обеспечение (модули, компоненты, продукты) и осуществлять разработку технических документов, адресованных	З-1 - Характеризовать алгоритмы решения типовых задач, области и способы их применения З-2 - Различать синтаксис языков программирования, особенности программирования на этих языках, стандартные библиотеки языков программирования П-1 - Создавать блок-схемы алгоритмов функционирования разрабатываемых программных продуктов в соответствии с	Домашняя работа Контрольная работа Лабораторные занятия Лекции Экзамен

<p>специалисту по информационным технологиям и пользователям (Информатика и вычислительная техника)</p>	<p>требованиями технического задания П-2 - Создавать и оптимизировать программный код на языках программирования высокого и низкого уровня с использованием специализированных программных средств П-3 - Иметь практический опыт использования инструмента контроля версий Git (GitHub, GitLab и др.) и комбинированной среды управления проектами Redmine и аналогов У-1 - Различать особенности стандартных алгоритмов для решения задач в соответствующих областях профессиональной деятельности У-2 - Выбирать языки программирования для написания программного кода с учетом технического задания</p>	
<p>ПК-1 -Способен проектировать, разрабатывать, интегрировать, проверять на работоспособность программное обеспечение (модули, компоненты, продукты) и осуществлять разработку технических документов, адресованных специалисту по информационным технологиям и пользователям (Прикладная информатика)</p>	<p>3-1 - Характеризовать алгоритмы решения типовых задач, области и способы их применения 3-2 - Различать синтаксис языков программирования, особенности программирования на этих языках, стандартные библиотеки языков программирования П-1 - Создавать блок-схемы алгоритмов функционирования разрабатываемых программных продуктов в соответствии с требованиями технического задания П-2 - Создавать и оптимизировать программный код на языках программирования высокого и низкого уровня с использованием специализированных программных средств</p>	<p>Домашняя работа Контрольная работа Лабораторные занятия Лекции Экзамен</p>

	<p>П-3 - Иметь практический опыт использования инструмента контроля версий Git (GitHub, GitLab и др.) и комбинированной среды управления проектами Redmine и аналогов</p> <p>У-1 - Различать особенности стандартных алгоритмов для решения задач в соответствующих областях профессиональной деятельности</p> <p>У-2 - Выбирать языки программирования для написания программного кода с учетом технического задания</p>	
<p>ПК-1 -Способен проектировать, разрабатывать, интегрировать, проверять на работоспособность программное обеспечение (модули, компоненты, продукты) и осуществлять разработку технических документов, адресованных специалисту по информационным технологиям и пользователям (Программная инженерия)</p>	<p>З-1 - Характеризовать алгоритмы решения типовых задач, области и способы их применения</p> <p>З-2 - Различать синтаксис языков программирования, особенности программирования на этих языках, стандартные библиотеки языков программирования</p> <p>П-1 - Создавать блок-схемы алгоритмов функционирования разрабатываемых программных продуктов в соответствии с требованиями технического задания</p> <p>П-2 - Создавать и оптимизировать программный код на языках программирования высокого и низкого уровня с использованием специализированных программных средств</p> <p>П-3 - Иметь практический опыт использования инструмента контроля версий Git (GitHub, GitLab и др.) и комбинированной среды управления проектами Redmine и аналогов</p> <p>У-1 - Различать особенности стандартных алгоритмов для решения задач в</p>	<p>Домашняя работа Контрольная работа Лабораторные занятия Лекции Экзамен</p>

	соответствующих областях профессиональной деятельности У-2 - Выбирать языки программирования для написания программного кода с учетом технического задания	
ПК-2 -Способен разрабатывать тесты, подготавливать тестовые данные, проводить тестирование, разрабатывать документы для тестирования и анализировать результаты тестирования программного обеспечения (Информатика и вычислительная техника)	З-1 - Изложить теорию тестирования (модели тестирования, планирование тестирования, тест-дизайн, проектирование тестов и др.) З-2 - Перечислить основные техники тестирования, стандарты в области тестирования, методологии, применяемые к необходимым приложениям П-1 - Проводить необходимые виды тестирования в соответствии с планом тестирования П-2 - Выполнять анализ полученных результатов тестирования и оформлять в соответствии с требуемым форматом П-3 - Имеет практический опыт работы с тестовыми средами и системами управления тестированием (Test Link и аналоги) в своей профессиональной деятельности У-1 - Идентифицировать цели, объекты, входные данные и виды тестирования (приемочное, установочное, альфа- и бета-тестирование и др.) У-2 - Оценивать важность (приоритет выполнения) различных тестов (на основе приоритетов пользователя, проектных задач и рисков возникновения ошибки)	Домашняя работа Контрольная работа Лабораторные занятия Лекции Экзамен
ПК-2 -Способен разрабатывать тесты, подготавливать тестовые данные,	З-1 - Изложить теорию тестирования (модели тестирования, планирование	Домашняя работа Контрольная работа Лабораторные занятия Лекции

<p>проводить тестирование, разрабатывать документы для тестирования и анализировать результаты тестирования программного обеспечения (Прикладная информатика)</p>	<p>тестирования, тест-дизайн, проектирование тестов и др.) З-2 - Перечислить основные техники тестирования, стандарты в области тестирования, методологии, применяемые к необходимым приложениям П-1 - Проводить необходимые виды тестирования в соответствии с планом тестирования П-2 - Выполнять анализ полученных результатов тестирования и оформлять в соответствии с требуемым форматом П-3 - Имеет практический опыт работы с тестовыми средами и системами управления тестированием (Test Link и аналоги) в своей профессиональной деятельности У-1 - Идентифицировать цели, объекты, входные данные и виды тестирования (приемочное, установочное, альфа- и бета-тестирование и др.) У-2 - Оценивать важность (приоритет выполнения) различных тестов (на основе приоритетов пользователя, проектных задач и рисков возникновения ошибки)</p>	<p>Экзамен</p>
<p>ПК-2 -Способен разрабатывать тесты, подготавливать тестовые данные, проводить тестирование, разрабатывать документы для тестирования и анализировать результаты тестирования программного обеспечения</p>	<p>З-1 - Изложить теорию тестирования (модели тестирования, планирование тестирования, тест-дизайн, проектирование тестов и др.) З-2 - Перечислить основные техники тестирования, стандарты в области тестирования, методологии, применяемые к необходимым приложениям П-1 - Проводить необходимые виды тестирования в соответствии с планом тестирования</p>	<p>Домашняя работа Контрольная работа Лабораторные занятия Лекции Экзамен</p>

(Программная инженерия)	<p>П-2 - Выполнять анализ полученных результатов тестирования и оформлять в соответствии с требуемым форматом</p> <p>П-3 - Имеет практический опыт работы с тестовыми средами и системами управления тестированием (Test Link и аналоги) в своей профессиональной деятельности</p> <p>У-1 - Идентифицировать цели, объекты, входные данные и виды тестирования (приемочное, установочное, альфа- и бета-тестирование и др.)</p> <p>У-2 - Оценивать важность (приоритет выполнения) различных тестов (на основе приоритетов пользователя, проектных задач и рисков возникновения ошибки)</p>	
-------------------------	---	--

3. ПРОЦЕДУРЫ КОНТРОЛЯ И ОЦЕНИВАНИЯ РЕЗУЛЬТАТОВ ОБУЧЕНИЯ В РАМКАХ ТЕКУЩЕЙ И ПРОМЕЖУТОЧНОЙ АТТЕСТАЦИИ ПО ДИСЦИПЛИНЕ МОДУЛЯ В БАЛЬНО-РЕЙТИНГОВОЙ СИСТЕМЕ (ТЕХНОЛОГИЧЕСКАЯ КАРТА БРС)

3.1. Процедуры текущей и промежуточной аттестации по дисциплине

1. Лекции: коэффициент значимости совокупных результатов лекционных занятий – не предусмотрено		
Текущая аттестация на лекциях	Сроки – семестр, учебная неделя	Максимальная оценка в баллах
Весовой коэффициент значимости результатов текущей аттестации по лекциям – не предусмотрено		
Промежуточная аттестация по лекциям – нет Весовой коэффициент значимости результатов промежуточной аттестации по лекциям – не предусмотрено		
2. Практические/семинарские занятия: коэффициент значимости совокупных результатов практических/семинарских занятий – не предусмотрено		
Текущая аттестация на практических/семинарских занятиях	Сроки – семестр, учебная неделя	Максимальная оценка в баллах

Весовой коэффициент значимости результатов текущей аттестации по практическим/семинарским занятиям– не предусмотрено		
Промежуточная аттестация по практическим/семинарским занятиям–нет		
Весовой коэффициент значимости результатов промежуточной аттестации по практическим/семинарским занятиям– не предусмотрено		
3. Лабораторные занятия: коэффициент значимости совокупных результатов лабораторных занятий –0.5		
Текущая аттестация на лабораторных занятиях	Сроки – семестр, учебная неделя	Максимальная оценка в баллах
<i>выполнение и защита лабораторных работ</i>	3,16	40
<i>домашняя работа</i>	3,7	30
<i>контрольная работа</i>	3,14	30
Весовой коэффициент значимости результатов текущей аттестации по лабораторным занятиям -1.00		
Промежуточная аттестация по лабораторным занятиям –нет		
Весовой коэффициент значимости результатов промежуточной аттестации по лабораторным занятиям – не предусмотрено		
4. Онлайн-занятия: коэффициент значимости совокупных результатов онлайн-занятий –0.5		
Текущая аттестация на онлайн-занятиях	Сроки – семестр, учебная неделя	Максимальная оценка в баллах
<i>онлайн-курс "Прикладное программирование на Python"</i>	3,16	100
Весовой коэффициент значимости результатов текущей аттестации по онлайн-занятиям -0.5		
Промежуточная аттестация по онлайн-занятиям –экзамен		
Весовой коэффициент значимости результатов промежуточной аттестации по онлайн-занятиям – 0.5		

3.2. Процедуры текущей и промежуточной аттестации курсовой работы/проекта

Текущая аттестация выполнения курсовой работы/проекта	Сроки – семестр, учебная неделя	Максимальная оценка в баллах
Весовой коэффициент текущей аттестации выполнения курсовой работы/проекта– не предусмотрено		
Весовой коэффициент промежуточной аттестации выполнения курсовой работы/проекта– защиты – не предусмотрено		

3.1. Процедуры текущей и промежуточной аттестации по дисциплине

2. Лекции: коэффициент значимости совокупных результатов лекционных занятий – не предусмотрено		
Текущая аттестация на лекциях	Сроки – семестр, учебная неделя	Максимальная оценка в баллах

Весовой коэффициент значимости результатов текущей аттестации по лекциям – не предусмотрено		
Промежуточная аттестация по лекциям – нет		
Весовой коэффициент значимости результатов промежуточной аттестации по лекциям – не предусмотрено		
2. Практические/семинарские занятия: коэффициент значимости совокупных результатов практических/семинарских занятий – не предусмотрено		
Текущая аттестация на практических/семинарских занятиях	Сроки – семестр, учебная неделя	Максимальная оценка в баллах
Весовой коэффициент значимости результатов текущей аттестации по практическим/семинарским занятиям– не предусмотрено		
Промежуточная аттестация по практическим/семинарским занятиям–нет		
Весовой коэффициент значимости результатов промежуточной аттестации по практическим/семинарским занятиям– не предусмотрено		
3. Лабораторные занятия: коэффициент значимости совокупных результатов лабораторных занятий –не предусмотрено		
Текущая аттестация на лабораторных занятиях	Сроки – семестр, учебная неделя	Максимальная оценка в баллах
Весовой коэффициент значимости результатов текущей аттестации по лабораторным занятиям -не предусмотрено		
Промежуточная аттестация по лабораторным занятиям –нет		
Весовой коэффициент значимости результатов промежуточной аттестации по лабораторным занятиям – не предусмотрено		
4. Онлайн-занятия: коэффициент значимости совокупных результатов онлайн-занятий –1		
Текущая аттестация на онлайн-занятиях	Сроки – семестр, учебная неделя	Максимальная оценка в баллах
<i>онлайн-курс "Прикладное программирование на Python"</i>	3,16	100
Весовой коэффициент значимости результатов текущей аттестации по онлайн-занятиям -0.5		
Промежуточная аттестация по онлайн-занятиям –экзамен		
Весовой коэффициент значимости результатов промежуточной аттестации по онлайн-занятиям – 0.5		

3.2. Процедуры текущей и промежуточной аттестации курсовой работы/проекта

Текущая аттестация выполнения курсовой работы/проекта	Сроки – семестр, учебная неделя	Максимальная оценка в баллах
Весовой коэффициент текущей аттестации выполнения курсовой работы/проекта– не предусмотрено		
Весовой коэффициент промежуточной аттестации выполнения курсовой работы/проекта– защиты – не предусмотрено		

4. КРИТЕРИИ И УРОВНИ ОЦЕНИВАНИЯ РЕЗУЛЬТАТОВ ОБУЧЕНИЯ ПО ДИСЦИПЛИНЕ МОДУЛЯ

4.1. В рамках БРС применяются утвержденные на кафедре/институте критерии (признаки) оценивания достижений студентов по дисциплине модуля (табл. 4) в рамках контрольно-оценочных мероприятий на соответствие указанным в табл.1 результатам обучения (индикаторам).

Таблица 4

Критерии оценивания учебных достижений обучающихся

Результаты обучения	Критерии оценивания учебных достижений, обучающихся на соответствие результатам обучения/индикаторам
Знания	Студент демонстрирует знания и понимание в области изучения на уровне указанных индикаторов и необходимые для продолжения обучения и/или выполнения трудовых функций и действий, связанных с профессиональной деятельностью.
Умения	Студент может применять свои знания и понимание в контекстах, представленных в оценочных заданиях, демонстрирует освоение умений на уровне указанных индикаторов и необходимых для продолжения обучения и/или выполнения трудовых функций и действий, связанных с профессиональной деятельностью.
Опыт /владение	Студент демонстрирует опыт в области изучения на уровне указанных индикаторов.
Другие результаты	Студент демонстрирует ответственность в освоении результатов обучения на уровне запланированных индикаторов. Студент способен выносить суждения, делать оценки и формулировать выводы в области изучения. Студент может сообщать преподавателю и коллегам своего уровня собственное понимание и умения в области изучения.

4.2 Для оценивания уровня выполнения критериев (уровня достижений обучающихся при проведении контрольно-оценочных мероприятий по дисциплине модуля) используется универсальная шкала (табл. 5).

Таблица 5

Шкала оценивания достижения результатов обучения (индикаторов) по уровням

Характеристика уровней достижения результатов обучения (индикаторов)				
№ п/п	Содержание уровня выполнения критерия оценивания результатов обучения (выполненное оценочное задание)	Шкала оценивания		
		Традиционная характеристика уровня		Качественная характеристика уровня
1.	Результаты обучения (индикаторы) достигнуты в полном объеме, замечаний нет	Отлично (80-100 баллов)	Зачтено	Высокий (В)

2.	Результаты обучения (индикаторы) в целом достигнуты, имеются замечания, которые не требуют обязательного устранения	Хорошо (60-79 баллов)		Средний (С)
3.	Результаты обучения (индикаторы) достигнуты не в полной мере, есть замечания	Удовлетворительно (40-59 баллов)		Пороговый (П)
4.	Освоение результатов обучения не соответствует индикаторам, имеются существенные ошибки и замечания, требуется доработка	Неудовлетворительно (менее 40 баллов)	Не зачтено	Недостаточный (Н)
5.	Результат обучения не достигнут, задание не выполнено	Недостаточно свидетельств для оценивания		Нет результата

5. СОДЕРЖАНИЕ КОНТРОЛЬНО-ОЦЕНОЧНЫХ МЕРОПРИЯТИЙ ПО ДИСЦИПЛИНЕ МОДУЛЯ

5.1. Описание аудиторных контрольно-оценочных мероприятий по дисциплине модуля

5.1.1. Лекции

Самостоятельное изучение теоретического материала по темам/разделам лекций в соответствии с содержанием дисциплины (п. 1.2. РПД)

5.1.2. Лабораторные занятия

Примерный перечень тем

1. Интерпретатор Python и IDE
2. Переменные, основные типы данных. Условия и циклы
3. Работа с различными структурами данных
4. Процедуры, функции и встроенные библиотеки
5. Применение функционального программирования
6. Построение классов
7. Использование библиотек
8. Работа с репозиторием Git
9. Использование отладчика. Документирование и тестирование код
10. Рефакторинг кода
11. Использование API
12. Работа с данными в различных форматах: XML, JSON
13. Использование библиотек pandas, scikit-learn
14. Интеграция с базами данных
15. Создание веб-приложения

LMS-платформа

1. <https://openedu.ru/course/urfu/pyap/>

5.2. Описание внеаудиторных контрольно-оценочных мероприятий и средств текущего контроля по дисциплине модуля

Разноуровневое (дифференцированное) обучение.

Базовый

5.2.1. Контрольная работа

Примерный перечень тем

1. Элементы функционального программирования
2. Объектно-ориентированное программирование

Примерные задания

Контрольная работа состоит из двух частей: теоретической и практической.

Теоретическая часть включает несколько тестовых вопросов, например:

- 1) Каково будет значение следующего выражения Python $4 + 3 \% 5$?
a) 7 b) 2 c) 4 d) 1
- 2) Каким будет результат выполнения кода `'Hello, python'[7:4:-1]`?
- 3) Какой из методов является конструктором класса в Python?
a) `Cat()` b) `self` c) `_init_` d) `constructor`

Практическая часть включает задачи по программированию по теме ЭФП и/или ООП:

- 1) Написать программу, в которой определена функция, работающая со строкой.

Функция поддерживает 2 режима: режим "1" - выделяет слова из строки, печатает их в виде списка; режим "2" - выделяет слова из строки, печатает их заглавными буквами в обратном порядке.

Предусмотреть, вывод соответствующего сообщения, если режим введен не корректно.

2) Описать классы, в которых можно хранить данные структуры, представленной в файле. Файл "input.txt" хранит данные о преподавателях и студентах. Каждая строка - сведения об отдельном человеке. Запись состоит из полей, поля разделены запятыми.

Запись о преподавателе состоит из следующих полей: идентификатор записи преподавателя (1); год рождения; должность. Пример записи о преподавателе: 1, Петров Петр Петрович, 1990, ст. преподаватель

Запись о студенте состоит из следующих полей: идентификатор студенческой записи (2); год рождения; академическая группа, в которую записан студент. Пример записи о студенте: 2, Иванов Иван Иванович, 2003, РИ-480012

Написать программу, которая считывает данный из файла и выводит: список студентов в формате: фамилия (возраст), список преподавателей в формате: фамилия (возраст) - должность, общее количество записей.

LMS-платформа

1. <https://openedu.ru/course/urfu/pyar/>

5.2.2. Домашняя работа

Примерный перечень тем

1. Библиотеки для анализа данных в Python
2. Веб-программирование на Python

Примерные задания

Домашняя работа представляет собой написание корректно работающего приложения прикладного назначения. Приложение должно включать выгрузку данных, сохранение данных с последующей их обработкой с помощью библиотеки pandas, scikit-learn или

других библиотек, визуализацию данных средствами matplotlib или других библиотек. Возможно создание веб-приложения с использованием flask или django.

Решение домашнего задания необходимо предоставить в виде ссылки на файл с расширением .py или файл Jupyter Notebook с расширением .ipynb в репозитории GitHub. Должны быть также предоставлены результаты запуска на тестовых файлах.

LMS-платформа

1. <https://openedu.ru/course/urfu/pyar/>

5.3. Описание контрольно-оценочных мероприятий промежуточного контроля по дисциплине модуля

5.3.1. Экзамен

Список примерных вопросов

1. Особенности языка Python
2. Интерпретатор Python. IDE
3. Виртуальная среда окружения в Python
4. Переменные, основные типы данных
5. Динамическая типизация
6. Встроенные типы чисел — int, float, complex. Машинное представление чисел с плавающей точкой и целых. Преобразование типов
7. Встроенный тип str и его методы
8. Встроенные типы последовательностей list, tuple, range и их методы
9. Встроенный объект dict и его методы.
10. Множества. Встроенные типы set и frozenset.
11. Инструкции if/else/elif, логические операторы и выражения сравнения
12. Циклы while и for в Python
13. Функции в Python. Основные понятия
14. Области видимости и пространство имен в Python.
15. Модули в Python. Основные понятия
16. Функциональное программирование в Python
17. Генераторы списков. Функции высшего порядка: map, reduce, filter
18. Принципы объектно-ориентированного программирования
19. Атрибуты класса. Атрибуты данных. Атрибуты-методы. Параметр self.
20. Специальные методы и атрибуты классов. Методы `_init()` и `_del_()`
21. Интерфейсы. Итераторы и декораторы
22. Создание собственной библиотеки. PyPI
23. Репозитории. Git
24. Тестирование и документация в Python
25. Отладчики
26. Профилирование кода
27. Рефакторинг кода. Стили программирования
28. Процессы и потоки
29. API.
30. Форматы источников данных. CSV, XML, JSON
31. Библиотека pandas
32. Библиотека scikit-learn

- 33. Анализ данных в Python
- 34. Инструменты Python для машинного обучения
- 35. Взаимодействие Python с базой данных MongoDB
- 36. Взаимодействие Python с базой данных MariaDB
- 37. Архитектура веб-приложений
- 38. Библиотека Django
- 39. Библиотека Flask
- LMS-платформа
- 1. <https://openedu.ru/course/urfu/pyar/>

5.4 Содержание контрольно-оценочных мероприятий по направлениям воспитательной деятельности

Направление воспитательной деятельности	Вид воспитательной деятельности	Технология воспитательной деятельности	Компетенция	Результаты обучения	Контрольно-оценочные мероприятия
Профессиональное воспитание	профориентационная деятельность	Технология формирования уверенности и готовности к самостоятельной успешной профессиональной деятельности Технология самостоятельной работы	ПК-1	П-3	Домашняя работа Контрольная работа Лабораторные занятия Лекции Экзамен
			ПК-1	П-3	
			ПК-1	П-3	