

**Федеральное государственное автономное образовательное
учреждение высшего образования
«Московский физико-технический институт
(национальный исследовательский университет)»**

УТВЕРЖДЕНО

**Директор физтех-школы
радиотехники и компьютерных
технологий**

Д.А. Гаврилов

	Рабочая программа дисциплины (модуля)
по дисциплине:	Язык программирования Python
по направлению:	Информатика и вычислительная техника
профиль подготовки:	Компьютерные технологии и вычислительная техника Физтех-школа Радиотехники и Компьютерных Технологий кафедра перспективных вычислительных технологий
курс:	3
квалификация:	бакалавр

Семестр, формы промежуточной аттестации: 6 (весенний) - Дифференцированный зачет

Аудиторных часов: 45 всего, в том числе:

лекции: 30 час.

семинары: 15 час.

лабораторные занятия: 0 час.

Самостоятельная работа: 45 час.

Всего часов: 90, всего зач. ед.: 2

Программу составили:

С.В. Сидоров, канд. техн. наук, старший преподаватель

А.Г. Жогов, преподаватель

Программа обсуждена на заседании кафедры перспективных вычислительных технологий 16.05.2023

Аннотация

В курсе рассматриваются возможности языка Python для разработки программ. Уделяется внимание проектированию архитектуры, код-ревью и рафакторингу кода.

1. Цели и задачи

Цель дисциплины

Изучение концепций разработки программного обеспечения (ПО) и их реализации на мультипарадигменном языке Python.

Задачи дисциплины

Формирование навыков создания приложений на языке Python.

2. Перечень формируемых компетенций

Освоение дисциплины направлено на формирование следующих компетенций:

Код и наименование компетенции	Индикаторы достижения компетенции
УК-1 Способен осуществлять поиск, критический анализ и синтез информации, применять системный подход для решения поставленных задач	УК-1.2 Находит, критически анализирует и выбирает информацию, необходимую для решения поставленной задачи
ОПК-2 Способен использовать современные информационные технологии и программные средства при решении задач профессиональной деятельности, соблюдая требования информационной безопасности	ОПК-2.1 Способен применять современные вычислительную технику и сервисы сети Интернет в области (сфере) профессиональной деятельности
ОПК-3 Способен составлять и оформлять научные и (или) технические (технологические, инновационные) отчеты (публикации, проекты)	ОПК-3.3 Владеет методами визуального и графического представления результатов научной (научно-технической, инновационной технологической) деятельности в виде отчетов, научных публикаций
ПК-1 Способен ставить, формализовывать и решать задачи, в том числе разрабатывать и исследовать математические модели изучаемых явлений и процессов, системно анализировать научные проблемы, получать новые научные результаты	ПК-1.1 Способен находить, анализировать и обобщать информацию об актуальных результатах исследований в рамках тематической области своей профессиональной деятельности
	ПК-1.2 Способен выдвигать гипотезы, строить математические модели для описания изучаемых явлений и процессов, оценивать качество разработанной модели
	ПК-1.3 Способен применять теоретические и (или) экспериментальные методы исследований к конкретной научной задаче и интерпретировать полученные результаты
ПК-2 Способен самостоятельно или в качестве члена (руководителя) малого коллектива организовывать и проводить научные исследования и их апробацию	ПК-2.1 Знает принципы построения научной работы, методы сбора и анализа полученного материала, способы аргументации
	ПК-2.2 Способен планировать и проводить научные исследования самостоятельно или в качестве члена (руководителя) малого научного коллектива
	ПК-2.3 Способен проводить апробацию результатов научно-исследовательской работы посредством публикации научных статей и участия в конференциях

3. Перечень планируемых результатов обучения по дисциплине (модулю)

В результате освоения дисциплины обучающиеся должны знать:

Возможности языка Python для написания программ.

уметь:

Самостоятельно программировать на языке Python, как с использованием готовых модулей, так и создавая свои собственные.

владеть:

Навыками использования современных технологий программирования, навыками разработки информационных систем для решения прикладных задач.

4. Содержание дисциплины (модуля), структурированное по темам (разделам) с указанием отведенного на них количества академических часов и видов учебных занятий

4.1. Разделы дисциплины (модуля) и трудоемкости по видам учебных занятий

№	Тема (раздел) дисциплины	Трудоемкость по видам учебных занятий, включая самостоятельную работу, час.			
		Лекции	Семинары	Лаборат. работы	Самост. работа
1	Особенности языка Python	2	1		3
2	Синтаксис Python	2	1		3
3	Встроенные типы объектов	2	1		3
4	Управляющие конструкции	2	1		3
5	Модули и пакеты	2	1		3
6	Динамическая типизация	2	1		3
7	Многопоточное программирование	2	1		3
8	Исключения, отладка, линтеры	2	1		3
9	ООП в Python	2	1		3
10	Архитектура программ	4	2		6
11	Принципы SOLID	2	1		3
12	Код-ревью и рефакторинг программ	2	1		3
13	Тестирование	2	1		3
14	Создание графических приложений в Python	2	1		3
Итого часов		30	15		45
Подготовка к экзамену		0 час.			
Общая трудоёмкость		90 час., 2 зач.ед.			

4.2. Содержание дисциплины (модуля), структурированное по темам (разделам)

Семестр: 6 (Весенний)

1. Особенности языка Python

- Положительные стороны Python
- Области применения Python
- Преимущества Python перед другими языками
- Недостатки Python

2. Синтаксис Python

- Интерактивный режим
- Стили кода (PEP8)
- Автоматическое форматирование

3. Встроенные типы объектов

- Числа
- Строки
- Списки
- Кортежи
- Словари
- Множества

4. Управляющие конструкции

- Логические выражения
- Условия
- Циклы

5. Модули и пакеты

- Работа с пакетами PyPI, pip
- Популярные пакеты
- Стандартные модули и пакеты
- Внешние пакеты

6. Динамическая типизация

- «Утиная типизация»
- Специальные методы объектов

7. Многопоточное программирование

- Создание и завершение потоков
- Методы синхронизации

8. Исключения, отладка, линтеры

- Работа с исключениями в Python
- Пользовательские исключения
- Отладка программ
- Отладчик IDE
- Статические проверки кода

9. ООП в Python

- Особенности объектно ориентированного подхода в Python
- Особенности наследования и полиморфизма в Python

10. Архитектура программ

- Что такое архитектура ПО
- Связи и сцепленность
- Характеристики качества архитектуры
- Дзен Python
- Пример изменения

11. Принципы SOLID

- Принцип единственной ответственности
- Принцип открытости/закрытости
- Принцип подстановки Лисков
- Принцип разделения интерфейса
- Принцип инверсии зависимостей

12. Код-ревью и рефакторинг программ

- На что стоит обращать внимание при инспекции кода
- Что такое рефакторинг кода
- Хорошие комментарии в коде

13. Тестирование

- Unit tests
- PyTest
- Code coverage
- Отчеты

14. Создание графических приложений в Python

- Этапы создания GUI
- Создание графического интерфейса на Tkinter
- Принципы построения интерфейса с использованием библиотеки Tk

5. Описание материально-технической базы, необходимой для осуществления образовательного процесса по дисциплине (модулю)

Учебная аудитория, оснащенная компьютером и мультимедийным оборудованием (проектор).

6.Перечень рекомендуемой литературы

Основная литература

1. Простой Python. Современный стиль программирования [Текст]/Б. Любанович, -СПб., Питер, 2018
2. Программирование на Python [Текст]/М. Лутц , -СПб., Символ-Плюс, 2002
- 3.Марк Лутц "Изучаем Python" "Learning Python"

Дополнительная литература

1. Совершенный код. Мастер - класс [Текст] : [учеб. пособие для вузов] / С. Макконнелл ; [пер. с англ.] .— М. : Русская редакция, 2013 .— 896 с.

7. Перечень ресурсов информационно-телекоммуникационной сети "Интернет", необходимых для освоения дисциплины (модуля)

Python Sowtware Foundation – URL: <http://www.python.org>

8. Перечень информационных технологий, используемых при осуществлении образовательного процесса по дисциплине (модулю), включая перечень необходимого программного обеспечения и информационных справочных систем (при необходимости)

Не предусмотрено.

9. Методические указания для обучающихся по освоению дисциплины (модуля)

Студент, изучающий дисциплину, должен с одной стороны, овладеть общим понятийным аппаратом, а с другой стороны, должен научиться применять теоретические знания на практике. В результате изучения дисциплины студент должен знать основные определения, понятия.

Успешное освоение курса требует напряжённой самостоятельной работы студента. В программе курса приведено минимально необходимое время для работы студента над темой. Самостоятельная работа включает в себя:

- чтение и конспектирование рекомендованной литературы;
- проработку учебного материала (по учебной и научной литературе);
- подготовку к дифференцированному зачету.

Руководство и контроль за самостоятельной работой студента осуществляется в форме индивидуальных консультаций.

Важно добиться понимания изучаемого материала, а не механического его запоминания. При затруднении изучения отдельных тем, вопросов, следует обращаться за консультациями к преподавателю.

ОЦЕНОЧНЫЕ МАТЕРИАЛЫ ПО ДИСЦИПЛИНЕ (МОДУЛЮ)

по направлению:	Информатика и вычислительная техника
профиль подготовки:	Компьютерные технологии и вычислительная техника Физтех-школа Радиотехники и Компьютерных Технологий кафедра перспективных вычислительных технологий
курс:	3
квалификация:	бакалавр

Семестр, формы промежуточной аттестации: 6 (весенний) - Дифференцированный зачет

Разработчики:

С.В. Сидоров, канд. техн. наук, старший преподаватель

А.Г. Жогов, преподаватель

1. Компетенции, формируемые в процессе изучения дисциплины

Код и наименование компетенции	Индикаторы достижения компетенции
УК-1 Способен осуществлять поиск, критический анализ и синтез информации, применять системный подход для решения поставленных задач	УК-1.2 Находит, критически анализирует и выбирает информацию, необходимую для решения поставленной задачи
ОПК-2 Способен использовать современные информационные технологии и программные средства при решении задач профессиональной деятельности, соблюдая требования информационной безопасности	ОПК-2.1 Способен применять современные вычислительную технику и сервисы сети Интернет в области (сфере) профессиональной деятельности
ОПК-3 Способен составлять и оформлять научные и (или) технические (технологические, инновационные) отчеты (публикации, проекты)	ОПК-3.3 Владеет методами визуального и графического представления результатов научной (научно-технической, инновационной технологической) деятельности в виде отчетов, научных публикаций
ПК-1 Способен ставить, формализовывать и решать задачи, в том числе разрабатывать и исследовать математические модели изучаемых явлений и процессов, системно анализировать научные проблемы, получать новые научные результаты	ПК-1.1 Способен находить, анализировать и обобщать информацию об актуальных результатах исследований в рамках тематической области своей профессиональной деятельности
	ПК-1.2 Способен выдвигать гипотезы, строить математические модели для описания изучаемых явлений и процессов, оценивать качество разработанной модели
	ПК-1.3 Способен применять теоретические и (или) экспериментальные методы исследований к конкретной научной задаче и интерпретировать полученные результаты
ПК-2 Способен самостоятельно или в качестве члена (руководителя) малого коллектива организовывать и проводить научные исследования и их апробацию	ПК-2.1 Знает принципы построения научной работы, методы сбора и анализа полученного материала, способы аргументации
	ПК-2.2 Способен планировать и проводить научные исследования самостоятельно или в качестве члена (руководителя) малого научного коллектива
	ПК-2.3 Способен проводить апробацию результатов научно-исследовательской работы посредством публикации научных статей и участия в конференциях

2. Показатели оценивания компетенций

В результате изучения дисциплины «Язык программирования Python» обучающийся должен:

знать:

Возможности языка Python для написания программ.

уметь:

Самостоятельно программировать на языке Python, как с использованием готовых модулей, так и создавая свои собственные.

владеть:

Навыками использования современных технологий программирования, навыками разработки информационных систем для решения прикладных задач.

3. Перечень типовых (примерных) вопросов, заданий, тем для подготовки к текущему контролю

С целью контроля освоения обучающимися учебного материала проводится короткий (10-15 минут) опрос по материалу прошедших занятий в устной или письменной форме

4. Перечень типовых (примерных) вопросов и тем для проведения промежуточной аттестации обучающихся

1. Этапы жизненного цикла программного обеспечения. Краткая характеристика каждого этапа. Подходы к разработке. Субъекты жизненного цикла.
2. Виды сложности, присущие конструированию программного обеспечения. Привнесённая сложность.
3. Разработка требований. Функциональные и нефункциональные требования. Субъекты разработки.
4. Проектирование. Этапы проектирования. Архитектурное проектирование и другие этапы. Методы и технологии проектирования ПО. Подходы к разработке ПО.
5. Структурный подход. Используемые диаграммы. SADT/DFD/ERD. Методы декомпозиции и графическое представление. Типы связей для функций и данных.
6. Объектно-ориентированный подход. Элементы объектной модели. ООП понятия. UML диаграммы. RUP, стадии и основные процессы.
7. Разработка исходного кода. Требования, предъявляемые к исходному коду. Обоснование требований.
8. Тестирование ПО. Верификация и аттестация. Критерии выпуска ПО. Классификация тестирования.
9. Аспекты разработки отчуждаемого и модифицируемого исходного кода. Комментирование, стиль кодирования, именование объектов. Обоснование их необходимости.
10. Аспекты разработки надёжного исходного кода. Защитное программирование. Режимы сборки ПО.
11. Аспекты разработки эффективного исходного кода. Широко известные контейнеры, применяемые при разработке ПО, сервисы доступа к ним. Сложность сервисов. Оптимизация.
12. Отладка исходного кода. Этапы выявления дефектов. Инструменты отладки.
13. Переносимое ПО. Уровни переносимости. Схема проектирования переносимых программ.
14. Конструирование ПО в команде. Работа с репозиторием. Работа с ветками и стволом.

Критерии оценивания

Оценка отлично (10) выставляется студенту, показавшему всесторонние, систематизированные, глубокие знания учебной программы дисциплины, проявляющему интерес к данной предметной области, продемонстрировавшему умение уверенно и творчески применять их на практике при решении конкретных задач, свободное и правильное обоснование принятых решений.

Оценка отлично (9) выставляется студенту, показавшему всесторонние, систематизированные, глубокие знания учебной программы дисциплины и умение уверенно применять их на практике при решении конкретных задач, свободное и правильное обоснование принятых решений.

Оценка отлично (8) выставляется студенту, показавшему систематизированные, глубокие знания учебной программы дисциплины и умение уверенно применять их на практике при решении конкретных задач, правильное обоснование принятых решений, с некоторыми недочётами.

Оценка хорошо (7) выставляется студенту, если он твердо знает материал, грамотно и по существу излагает его, умеет применять полученные знания на практике, но недостаточно грамотно обосновывает полученные результаты.

Оценка хорошо (6) выставляется студенту, если он твердо знает материал, грамотно и по существу излагает его, умеет применять полученные знания на практике, но допускает в ответе или в решении задач некоторые неточности.

Оценка хорошо (5) выставляется студенту, если он в основном знает материал, грамотно и по существу излагает его, умеет применять полученные знания на практике, но допускает в ответе или в решении задач достаточно большое количество неточностей.

Оценка удовлетворительно (4) выставляется студенту, показавшему фрагментарный, разрозненный характер знаний, недостаточно правильные формулировки базовых понятий, нарушения логической последовательности в изложении программного материала, но при этом он освоил основные разделы учебной программы, необходимые для дальнейшего обучения, и может применять полученные знания по образцу в стандартной ситуации.

Оценка удовлетворительно (3) выставляется студенту, показавшему фрагментарный, разрозненный характер знаний, допускающему ошибки в формулировках базовых понятий, нарушения логической последовательности в изложении программного материала, слабо владеет основными разделами учебной программы, необходимыми для дальнейшего обучения и с трудом применяет полученные знания даже в стандартной ситуации.

Оценка неудовлетворительно (2) выставляется студенту, который не знает большей части основного содержания учебной программы дисциплины, допускает грубые ошибки в формулировках основных принципов и не умеет использовать полученные знания при решении типовых задач.

Оценка неудовлетворительно (1) выставляется студенту, который не знает основного содержания учебной программы дисциплины, допускает грубейшие ошибки в формулировках базовых понятий дисциплины и вообще не имеет навыков решения типовых практических задач.

5. Методические материалы, определяющие процедуры оценивания знаний, умений, навыков и (или) опыта деятельности

Дифференцированный зачет проводится в устной форме. При проведении дифференцированного зачета обучающемуся предоставляется 20 минут на подготовку. Опрос обучающегося проводится в течение 30 минут.

Во время проведения дифференцированного зачета обучающиеся могут пользоваться программой дисциплины.