

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

УТВЕРЖДЕНО

**Проректор по учебной работе и
довузовской подготовке**

А.А. Воронов

	Рабочая программа дисциплины (модуля)
по дисциплине:	Программирование на языке Python
по направлению:	Биотехнология
профиль подготовки:	Системная и синтетическая биология Физтех-школа Биологической и Медицинской Физики кафедра информатики и вычислительной математики
курс:	3
квалификация:	бакалавр

Семестр, формы промежуточной аттестации: 5 (осенний) - Дифференцированный зачет

Аудиторных часов: 60 всего, в том числе:

лекции: 0 час.

семинары: 0 час.

лабораторные занятия: 60 час.

Самостоятельная работа: 30 час.

Всего часов: 90, всего зач. ед.: 2

Количество контрольных работ, заданий: 4

Программу составил: В.И. Голубев, канд. физ.-мат. наук, доцент, доцент

Программа обсуждена на заседании кафедры информатики и вычислительной математики 06.02.2020

Аннотация

Преимущества и недостатки языка Python 3. Дзен Python.

Концепция присваивания в Python. Переменные, значения и их типы. Понятие о динамической типизации. Обмен двух переменных значениями. Кортежи переменных.

Арифметические операции. Возведение в степень, деление нацело.

Цикл while. Инструкции управления циклом. Вложенный цикл while.

Однопроходные алгоритмы: подсчёт, сумма, произведение. Среднее арифметическое.

Оператор if. Каскадная условная конструкция elif. Логические операции and, or, not в Python.

Однопроходные алгоритмы: поиск числа в потоке, максимум.

Тест простоты числа. Разложение числа на цифры. Алгоритм Евклида

Тип str. Неизменяемость строки. Длина строки len(s). Наивный поиск подстроки в строке. Методы строк find, rfind, count, replace.

Тип list. Изменяемость списка. Ссылочная модель данных в Python. Операторы == и is. Стандартные функции len, max, min, sum. Срезы списков. Присваивание в срез. Методы списка. Копирование объектов list.

List comprehensions: генерация списков. Алгоритм обращения массива. Алгоритм циклического сдвига в массиве.

Описание функций. Именованные параметры функций.

Цикл for и его особенности в Python. Функция range().

Сортировка вставками. Сортировка выбором. Сортировка методом пузырька.

1. Цели и задачи

Цель дисциплины

Научить студентов программировать на языке Python на уровне, достаточном для использования в исследовательской научной и в последующей профессиональной деятельности.

Задачи дисциплины

- ☐ обеспечить чёткое понимание студентами основ информатики и ИКТ, включая некоторые области математики (системы счисления, логика, дискретная математика);
- ☐ обучить студентов основным алгоритмам обработки числовой и текстовой информации;
- ☐ сформировать у обучающихся навык использования языка программирования Python 3 для решения конкретных прикладных задач.

2. Перечень формируемых компетенций

Освоение дисциплины направлено на формирование следующих компетенций:

Код и наименование компетенции	Индикаторы достижения компетенции
ОПК-2 Способен использовать современные информационные технологии и программные средства при решении задач профессиональной деятельности, соблюдая требования информационной безопасности	ОПК-2.2 Знает и умеет применять численные математические методы и прикладное программное обеспечение для решения научных задач в профессиональной области

3. Перечень планируемых результатов обучения по дисциплине (модулю)

В результате освоения дисциплины обучающиеся должны

знать:

- ☐ основы алгоритмического языка программирования Python;
- ☐ приёмы разработки программ.

уметь:

- ☐ выбирать оптимальные алгоритмы для программ;
- ☐ разрабатывать полные законченные программы на языке Python с использованием современных средств написания и отладки программ;
- ☐ использовать математические пакеты языка Python для автоматизации решения прикладных и фундаментальных задач.

владеть:

- ☐ навыком исследования и решения теоретических и прикладных задач;
- ☐ навыком программирования решений алгоритмических проблем на языке Python;
- ☐ способами представления полученных результатов;
- ☐ навыками самостоятельной работы и использования информации из баз знаний в Интернет.

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	Основы синтаксиса языка Python			15	8
2	Работа с коллекциями			15	7
3	Работа с файлами			15	8
4	Функции и работа с ними			15	7
Итого часов				60	30
Подготовка к экзамену		0 час.			
Общая трудоёмкость		90 час., 2 зач.ед.			

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

Семестр: 5 (Осенний)

1. Основы синтаксиса языка Python

Переменные. Выражения. Функции. Условные операторы и циклы.

2. Работа с коллекциями

Методы коллекций. Перебор коллекций. Условные выражения и коллекции. Сравнение коллекций.

3. Работа с файлами

Методы open и close. Конструкция with as. Чтение и запись данных в различных кодировках.

4. Функции и работа с ними

Значения аргументов функции по умолчанию. Обязательные и не обязательные аргументы. Позиционные и именованные аргументы.

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

Компьютерный класс с доской, проектором или телевизором, подключенный к сети, установленным необходимым ПО, доступом к учебным материалам.

6. Перечень рекомендуемой литературы

Основная литература

1. Программирование на Python 3 : Подробное руководство [Текст] = Programming in Python 3 : [учеб. пособие для вузов] / М. Саммерфилд; пер. с англ. А. Киселева .— СПб : Символ-Плюс, 2015 .— 608 с.

Дополнительная литература

1. Приемы объектно-ориентированного проектирования. Паттерны проектирования [Текст] : [учеб. пособие для вузов] / Э. Гамма [и др.] ; [пер. с англ. А. Слинкина] .— СПб. : Питер, 2012 .— 368 с.

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

Не используются

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

Основная литература

1. Программирование на Python 3 : Подробное руководство [Текст] = Programming in Python 3 : [учеб. пособие для вузов] / М. Саммерфилд; пер. с англ. А. Киселева .— СПб : Символ-Плюс, 2015 .— 608 с.
2. Python : карманный справочник. М. Лутц ; пер. с англ. И. В. Берштейна .— 5-е изд. — М : Вильямс, 2015 г. ISBN 978-5-8459-1965-6.

Дополнительная литература

1. Язык программирования PYTHON [Текст] : учеб. пособие для вузов / Р. А. Сузи .— 2 изд., испр. — М. : Интернет-Ун-т Информ. Технологий : БИНОМ. Лаб. знаний, 2007 .— 326 с.
2. Програмируем на Python. Майкл Доусон. Издательство: Питер ISBN 978-5-459-00314-7, 978-1435455009; 2012
3. Лекции «Алгоритмы: построение, анализ и реализация на языке программирования Си». Ворожцов А.В., Винокуров Н.А. — МФТИ, 2007

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

Работа по предмету включает посещение лабораторных занятий и самостоятельное решение задач.

Самостоятельная работа включает в себя: чтение и конспектирование рекомендованной литературы, просмотр интернет-ресурсов по тематике курса, решение задач, подготовку к ответам на контрольные вопросы.

ОЦЕНОЧНЫЕ МАТЕРИАЛЫ ПО ДИСЦИПЛИНЕ (МОДУЛЮ)

по направлению: Биотехнология
профиль подготовки: Системная и синтетическая биология
Физтех-школа Биологической и Медицинской Физики
кафедра информатики и вычислительной математики
курс: 3
квалификация: бакалавр

Семестр, формы промежуточной аттестации: 5 (осенний) - Дифференцированный зачет

Разработчик: В.И. Голубев, канд. физ.-мат. наук, доцент, доцент

1. Компетенции, формируемые в процессе изучения дисциплины

Код и наименование компетенции	Индикаторы достижения компетенции
ОПК-2 Способен использовать современные информационные технологии и программные средства при решении задач профессиональной деятельности, соблюдая требования информационной безопасности	ОПК-2.2 Знает и умеет применять численные математические методы и прикладное программное обеспечение для решения научных задач в профессиональной области

2. Показатели оценивания компетенций

В результате изучения дисциплины «Программирование на языке Python» обучающийся должен:

знать:

- ☐ основы алгоритмического языка программирования Python;
- ☐ приёмы разработки программ.

уметь:

- ☐ выбирать оптимальные алгоритмы для программ;
- ☐ разрабатывать полные законченные программы на языке Python с использованием современных средств написания и отладки программ;
- ☐ использовать математические пакеты языка Python для автоматизации решения прикладных и фундаментальных задач.

владеть:

- ☐ навыком исследования и решения теоретических и прикладных задач;
- ☐ навыком программирования решений алгоритмических проблем на языке Python;
- ☐ способами представления полученных результатов;
- ☐ навыками самостоятельной работы и использования информации из баз знаний в Интернет.

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

Проводятся короткие тесты на усвоение материала в начале занятий.

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

Примерный перечень контрольных вопросов:

1. Концепция присваивания в Python
2. Функции
3. Цикл while. Инструкции управления циклом.
4. Оператор if. Каскадная условная конструкция elif.
5. Логические операции в Python.
6. Тип str. Длина строки len(s). неизменяемость строки.
7. Срезы строк.
8. Методы строк find, count, replace, startswith, endswith.
9. Ссылочная модель данных в Python. Операторы == и is. Копирование объектов.
10. Срезы списков. Присваивание в срез. Методы списка.
11. Список строк. Методы split и join для строки.
12. Цикл for и его особенности в Python.
13. List comprehensions: генерация списков.
14. Двумерные массивы (списки списков). Вложенная генерация.
15. Тип set. Множества и работа с ними.
16. Кортежи и их использование.
17. Тип dict. Словарь (ассоциативный массив) и операции с ним.
18. Dict comprehensions: генерация множеств и словарей.
19. Работа с файлами. Методы open, close. Преимущества использования конструкции with as.

Критерии оценивания

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.

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.

Отлично (8) - Выставляется студенту, показавшему всесторонние, систематизированные, глубокие знания учебной программы дисциплины и умение уверенно применять их на практике при решении конкретных задач, правильное обоснование принятых решений, с некоторыми недочетами.

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.

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.

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.

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.

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.

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.

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

Во время проведения дифференцированного зачёта обучающиеся могут пользоваться программой дисциплины, а также справочной литературой, вычислительной техникой, конспектами занятий.