|  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **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:** | | Информатика | | | | | | | |
| **по направлению:** | | Прикладные математика и физика | | | | | | | |
| **профиль подготовки:** |  | Физика перспективных технологий: электроника и квантовые технологии | | | | | | | |
|  |  | Физтех-школа Электроники, Фотоники и Молекулярной Физики | | | | | | | |
|  | | кафедра информатики и вычислительной математики | | | | | | | |
| **курс:** | | 1 | | | | | | |  |
| **квалификация:** | | бакалавр | | | | | | |  |
|  |  |  |  |  |  |  | |  |  |
| Семестры, формы промежуточной аттестации: | | | | | |  | |  |  |
|  |  | 1 (осенний) - Дифференцированный зачет | | | |  | |  |  |
|  |  | 2 (весенний) - Дифференцированный зачет | | | |  | |  |  |
|  |  |  |  |  |  |  | |  |  |
| Аудиторных часов: 180 всего, в том числе: | | | | | |  | |  |  |
|  | лекции: 60 час. | | | | |  | |  |  |
|  | семинары: 0 час. | | | | |  | |  |  |
|  | лабораторные занятия: 120 час. | | | | |  | |  |  |
|  |  |  |  |  |  |  | |  |  |
| Самостоятельная работа: 180 час. | | | | | |  | |  |  |
|  |  |  |  |  |  |  | |  |  |
| Всего часов: 360, всего зач. ед.: 8 | | | | | |  | |  |  |
|  |  |  |  |  |  |  | |  |  |
| Количество контрольных работ, заданий: 8 | | | | | |  | |  |  |
|  |  |  |  |  |  |  | |  |  |
| Программу составили: | | | | | |  | |  |  |
|  |  |  |  |  |  |  | |  |  |
| Т.Ф. Хирьянов, старший преподаватель | | | | | | | | |  |
| К.Д. Гольдштейн, старший преподаватель | | | | | | | | |  |
|  |  |  |  |  |  |  | |  |  |
| Программа обсуждена на заседании кафедры информатики и вычислительной математики XX.XX.2023 | | | | | | | | | |
|  |  |  |  |  |  |  | |  |  |
|  |  |  |  |  |  |  | |  |  |
|  |  |  |  |  |  |  | |  |  | |
|  |  |  |  |  |  |  | |  |  |

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **Аннотация** | | | | | | | | | | | | | | | | | | |  |
| В данном курсе студенты познакомятся с языком программирования Python 3. Кроме этого студенты изучат широкий спектр алгоритмов и структур данных. Среди них: алгоритмы сортировки, алгоритмы динамического программирования, работа со строками, основы теории графов и др. | | | | | | | | | | | | | | | | | | |  |
|  | |  | |  | |  | |  | |  | |  | | | | |  | |  |
| **1. Цели и задачи** | | | | | | | | | | | |  | | | | |  | |  |
|  | |  | |  | |  | |  | |  | |  | | | | |  | |  |
| **Цель дисциплины** | | | |  | |  | |  | |  | |  | | | | |  | |  |
| Научить студентов программировать на языке Python 3 на уровне, достаточном для использования ИКТ в курсе вычислительной математики, в исследовательской научной и в последующей профессиональной деятельности. | | | | | | | | | | | | | | | | | | |  |
|  | |  | |  | |  | |  | |  | |  | | | | |  | |  |
| **Задачи дисциплины** | | | | | | | | | | | |  | | | | |  | |  |
| 1. Обеспечить чёткое понимание студентами основ информатики и ИКТ, включая некоторые области математики (системы счисления, логика, дискретная математика, теория графов); | | | | | | | | | | | | | | | | | | |  |
| 2. Обучить студентов основным алгоритмам обработки числовой и текстовой информации; | | | | | | | | | | | | | | | | | | |  |
| 3. Сформировать у обучающихся навык использования языка программирования Python 3 для решения конкретных прикладных задач; | | | | | | | | | | | | | | | | | | |  |
| 4. Научить студентов писать программный код коллективно с использованием промышленного стиля программирования и утилит, необходимых при совместной работе над программным продуктом. | | | | | | | | | | | | | | | | | | |  |
|  | |  | |  | |  | |  | |  | |  | | | | |  | |  |
| **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-1.1 Анализирует задачу, выделяя этапы ее решения, действия по решению задачи | | | | | | | | | | | | |  |
| УК-1.2 Находит, критически анализирует и выбирает информацию, необходимую для решения поставленной задачи | | | | | | | | | | | | |  |
| УК-1.3 Рассматривает различные варианты решения задачи, оценивает их преимущества и недостатки | | | | | | | | | | | | |  |
| УК-1.4 Грамотно, логично, аргументированно формирует собственные суждения и оценки | | | | | | | | | | | | |  |
| УК-1.5 Определяет и оценивает практические последствия возможных вариантов решения задачи | | | | | | | | | | | | |  |
| ОПК-1 Способен применять фундаментальные знания, полученные в области физико-математических и (или) естественных наук, и использовать их в профессиональной деятельности | | | | | | ОПК-1.1 Способен анализировать поставленную задачу, намечать пути ее решения | | | | | | | | | | | | |  |
| ОПК-1.2 Способен строить математические модели, производить количественные расчеты и оценки | | | | | | | | | | | | |  |
| ОПК-1.3 Способен определять границы применимости полученных результатов | | | | | | | | | | | | |  |
| 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-2.2 Знает и умеет применять численные математические методы и прикладное программное обеспечение для решения научных задач в профессиональной области | | | | | | | | | | | | |  |
| ОПК-2.3 Знает основные требования информационной безопасности | | | | | | | | | | | | |  |
| ОПК-4 Способен осуществлять сбор и обработку научно-технической и (или) технологической информации для решения фундаментальных и прикладных задач | | | | | | ОПК-4.1 Владеет методами научного поиска и интеллектуального анализа информации при решении задач профессиональной деятельности | | | | | | | | | | | | |  |
| ОПК-4.2 Знает основные источники научно-технической и (или) технологической информации в области профессиональной деятельности | | | | | | | | | | | | |  |
| ОПК-4.4 Владеет навыками работы с компьютером и компьютерными сетями с целью получения, хранения и обработки научной (технической, технологической) информации | | | | | | | | | | | | |  |
| ОПК-5 Способен участвовать в проведении фундаментальных и прикладных исследований и разработок, самостоятельно осваивать новые теоретические, в том числе, математические методы исследований, и работать на современной экспериментальной научно-исследовательской, измерительно-аналитической и технологической аппаратуре | | | | | | ОПК-5.1 Способен решать поставленные задачи в области теоретических и экспериментальных исследований и разработок | | | | | | | | | | | | |  |
| ОПК-5.2 Обладает способностью к освоению новых знаний на основе изучения литературы, научных статей и других источников | | | | | | | | | | | | |  |
| ОПК-5.3 Способен к профессиональной эксплуатации современной экспериментальной научно-исследовательской (измерительно-аналитической и технологической) аппаратуры | | | | | | | | | | | | |  |
| ПК-1 Способен планировать и проводить научные эксперименты (в избранной предметной области) и (или) теоретические (аналитические и имитационные) исследования | | | | | | ПК-1.4 Умеет строить математические модели для описания и исследования процессов и явлений в соответствующих научных областях | | | | | | | | | | | | |  |
| ПК-1.8 Владеет навыками работы с современными языками программирования и программными пакетами для научных расчетов | | | | | | | | | | | | |  |
| ПК-2 Способен анализировать полученные в ходе научно-исследовательской работы данные и делать научные выводы (заключения) | | | | | | ПК-2.1 Владеет методами статистической обработки и анализа научных данных | | | | | | | | | | | | |  |
| ПК-2.2 Умеет находить ключевые параметры, определяющие изучаемое явление, и производить численные оценки по порядку величины | | | | | | | | | | | | |  |
| ПК-3 Способен выбирать и применять подходящее оборудование, инструменты и методы исследований для решения задач в избранной предметной области | | | | | | ПК-3.3 Умеет производить оценку точности численных методов, используемых на ЭВМ, вычислительной сложности используемых алгоритмов и объема требуемых вычислительных ресурсов | | | | | | | | | | | | |  |
| ПК-4 Способен критически оценивать применимость используемых методик и методов | | | | | | ПК-4.3 Способен обосновать причинно-следственные отношения используемых понятий и моделей | | | | | | | | | | | | |  |
|  | |  | |  | |  | |  | |  | |  | | | | |  | |  |
| **3. Перечень планируемых результатов обучения по дисциплине (модулю)** | |  | |  | |  | |  | |  | |  | | | | |  | |  |
| В результате освоения дисциплины обучающиеся должны | | | | | | | | | | | |  | | | | |  | |  |
| знать: | | | |  | |  | |  | |  | |  | | | | |  | |  |
| • Основы теории алгоритмов; • свойства алгоритмов, проблемы алгоритмической сложности и алгоритмической неразрешимости; • основы дискретной математики; • основы алгоритмического языка программирования Python; • общие характеристики интерпретируемых и компилируемых языков программирования; • общие понятия о структурах данных: стеки, очереди, списки, деревья, таблицы; • приёмы разработки программ; • принципы программирования структур данных для современных программ, типовые решения, применяемые для создания программ; • основы работы с пакетами прикладных программ в области математики и физики. | | | | | | | | | | | | | | | | | | |  |
| уметь: | | | | | | | | | | | |  | | | | |  | |  |
| • Выбирать оптимальные алгоритмы для современных программ; • разрабатывать полные законченные программы на одном из языков высокого уровня; программы на одном или нескольких языках программирования, как индивидуально, так и в команде, с использованием современных средств написания и отладки программ; • использовать знания по информатике для приложений в инновационной, конструкторско-технологической и производственно-технологической сферах деятельности; • работать как на уровне языка командного интерпретатора, так и с использованием графического пользовательского интерфейса; • использовать сигналы и оконные сообщения для взаимодействия процессов между собой и с операционной системой; • создавать безопасные программы, использовать современные средства для написания и отладки программ; • работать с пакетами прикладных программ, включая использование развитых графических возможностей этих пакетов. | | | | | | | | | | | | | | | | | | |  |
| владеть: | | | | | | | | | | | |  | | | | |  | |  |
| • Языком программирования Python и методами создания программ с использованием стандартных библиотек; • средствами отладки программ на Python; • навыками программирования с использованием средств операционной системы для решения исследовательских задач; • основами работы с прикладными пакетами Python и принципами написания дополнительных модулей; • навыками освоения современных архитектур ЭВМ. | | | | | | | | | | | | | | | | | | |  |
|  | |  | |  | |  | |  | |  | |  | | | | |  | |  |
| **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 | | Знакомство с Python 3 | | | | 2 | |  | | 4 | | | | | 5 | | | |  |
| 2 | | Однопроходные алгоритмы | | | | 2 | |  | | 4 | | | | | 5 | | | |  |
| 3 | | Системы счисления | | | | 2 | |  | | 4 | | | | | 5 | | | |  |
| 4 | | Функции | | | | 2 | |  | | 4 | | | | | 5 | | | |  |
| 5 | | Списки и алгоритмы на списках | | | | 2 | |  | | 4 | | | | | 5 | | | |  |
| 6 | | Изменяемость списка list в Python | | | | 2 | |  | | 4 | | | | | 5 | | | |  |
| 7 | | Сортировки | | | | 2 | |  | | 4 | | | | | 5 | | | |  |
| 8 | | Рекурсия | | | | 2 | |  | | 4 | | | | | 5 | | | |  |
| 9 | | Быстрые сортировки | | | | 2 | |  | | 4 | | | | | 8 | | | |  |
| 10 | | Двоичный поиск | | | | 2 | |  | | 4 | | | | | 8 | | | |  |
| 11 | | Динамическое программирование | | | | 2 | |  | | 4 | | | | | 8 | | | |  |
| 12 | | Строки | | | | 2 | |  | | 4 | | | | | 8 | | | |  |
| 13 | | Двумерное динамическое программирование | | | | 2 | |  | | 4 | | | | | 8 | | | |  |
| 14 | | Структуры FIFO и LIFO | | | | 2 | |  | | 4 | | | | | 10 | | | |  |
| 15 | | Конечные и клеточные автоматы | | | | 2 | |  | | 4 | | | | |  | | | |  |
| 16 | | Сложность задач | | | | 2 | |  | | 4 | | | | | 5 | | | |  |
| 17 | | Хеширование | | | | 2 | |  | | 4 | | | | | 5 | | | |  |
| 18 | | Словари и множества в Python | | | | 2 | |  | | 4 | | | | | 5 | | | |  |
| 19 | | Связные списки | | | | 2 | |  | | 4 | | | | | 5 | | | |  |
| 20 | | Очередь и очередь с приоритетами | | | | 2 | |  | | 4 | | | | | 5 | | | |  |
| 21 | | Основы теории графов | | | | 2 | |  | | 4 | | | | | 5 | | | |  |
| 22 | | Хранение графа в памяти | | | | 2 | |  | | 4 | | | | | 5 | | | |  |
| 23 | | Поиск в глубину | | | | 2 | |  | | 4 | | | | | 5 | | | |  |
| 24 | | Поиск в ширину | | | | 2 | |  | | 4 | | | | | 8 | | | |  |
| 25 | | Поиск кратчайшего пути | | | | 2 | |  | | 4 | | | | | 8 | | | |  |
| 26 | | Остовные деревья | | | | 2 | |  | | 4 | | | | | 8 | | | |  |
| 27 | | Основы теории игр | | | | 2 | |  | | 4 | | | | | 8 | | | |  |
| 28 | | Двоичные деревья поиска | | | | 4 | |  | | 8 | | | | | 8 | | | |  |
| 29 | | Асимптотически сложные задачи на графах | | | | 2 | |  | | 4 | | | | | 10 | | | |  |
| Итого часов | | | | | | 60 | |  | | 120 | | | | | 180 | | | |  |
| Подготовка к экзамену | | | | | | 0 час. | | | | | | | | | | | | |  |
| Общая трудоёмкость | | | | | | 360 час., 8 зач.ед. | | | | | | | | | | | | |  |
|  | |  | |  | |  | |  | |  | |  | | | | |  | |  |
| 4.2. | | Содержание дисциплины (модуля), структурированное по темам (разделам) | | | | | | | | | | | | | | | | |  |
|  | |  | |  | |  | |  | |  | |  | | | | |  | |  |
| Семестр: 1 (Осенний) | | | | | | | | | | | |  | | | | |  | |  |
|  | |  | |  | |  | |  | |  | |  | | | | |  | |  |
|  | | 1. Знакомство с Python 3 | | | | | | | | | | | | | | | | |  |
|  | | | | | | | | | | | |  | | | | |  | |  |
|  | | Ход исполнения программы. Почему в Python нет goto. Интерактивный режим. Арифметические операции и их приоритеты. Типы данных. Преобразование типа. Ввод-вывод. Именованные параметры print() sep, end. Переменные. Присваивание: =, +=, -=, \*=, /=. «Трамвайное присваивание». Множественное присваивание. Обмен переменных значениями. Цикл for и функция range(). Однопроходные алгоритмы: сумма, произведение. Оператор ветвления if. Переменные-счётчики. Среднее арифметическое. Тип bool. Логические операции. Битовые операции &, |, ^. | | | | | | | | | | | | | | | | |  |
|  | |  | |  | |  | |  | |  | |  | | | | |  | |  |
|  | | 2. Однопроходные алгоритмы | | | | | | | | | | | | | | | | |  |
|  | | | | | | | | | | | |  | | | | |  | |  |
|  | | Обработка потока чисел с терминальным элементом. Поиск числа в потоке. Фильтрация потока чисел. Вложенные ветвления. Каскадные ветвления if-elif-else. Цикл while. Инструкции break и continue. Переменные-флаги. Максимальное число в потоке. Местоположение максимума. Количество равных максимуму. Поиск трёх максимумов за один проход. | | | | | | | | | | | | | | | | |  |
|  | |  | |  | |  | |  | |  | |  | | | | |  | |  |
|  | | 3. Системы счисления | | | | | | | | | | | | | | | | |  |
|  | | | | | | | | | | | |  | | | | |  | |  |
|  | | Целочисленное деление и взятие остатка, их отличие в С++ и Python. Позиционные системы счисления и литералы целых чисел в Python. Анализ цифр числа в произвольной системе счисления. Переводы из одной системы в другую. | | | | | | | | | | | | | | | | |  |
|  | |  | |  | |  | |  | |  | |  | | | | |  | |  |
|  | | 4. Функции | | | | | | | | | | | | | | | | |  |
|  | | | | | | | | | | | |  | | | | |  | |  |
|  | | Описание функций с параметрами. Синхронный вызов. Стек вызовов. Локальность переменных. Утиная типизация в Python. Метод грубой силы. Поиск НОД и НОК. Алгоритм Евклида. Тест простоты. Разложение числа на множители. | | | | | | | | | | | | | | | | |  |
|  | |  | |  | |  | |  | |  | |  | | | | |  | |  |
|  | | 5. Списки и алгоритмы на списках | | | | | | | | | | | | | | | | |  |
|  | | | | | | | | | | | |  | | | | |  | |  |
|  | | Создание списка чисел заданной длины. Функция len(). Индексация элементов от 0 до N-1. Скорость взятия и замены элемента A[i]. Распечатка массива. Задачи на заполнение массива. Заполнение массива числами Фибоначчи. Линейный поиск в массиве. Поэлементное копирование массива. Копирование задом-наперёд. Циклический сдвиг в массиве. Обращение массива. | | | | | | | | | | | | | | | | |  |
|  | |  | |  | |  | |  | |  | |  | | | | |  | |  |
|  | | 6. Изменяемость списка list в Python | | | | | | | | | | | | | | | | |  |
|  | | | | | | | | | | | |  | | | | |  | |  |
|  | | Ссылочная модель данных. Оператор идентичности is. Добавление и удаление элемента в начале и конце массива. Отличие по скорости A.pop(0) и A.pop(), и почему это так. Списковые включения («генераторы списков»). Решето Эратосфена. Частотный анализ (метод подсчёта). | | | | | | | | | | | | | | | | |  |
|  | |  | |  | |  | |  | |  | |  | | | | |  | |  |
|  | | 7. Сортировки | | | | | | | | | | | | | | | | |  |
|  | | | | | | | | | | | |  | | | | |  | |  |
|  | | Постановка задачи. Сортировка обезьяны. Сортировка выбором. Сортировка вставками. Ленивые and и or. Проверка упорядоченности массива за O(N). Сортировка дурака. Сортировка методом пузырька (через while с переменной-флагом). Синхронная сортировка нескольких массивов. Устойчивость сортировок. Сортировка подсчётом. Поразрядная сортировка для двоичной СС. Асимптотическая сложность алгоритмов. | | | | | | | | | | | | | | | | |  |
|  | |  | |  | |  | |  | |  | |  | | | | |  | |  |
|  | | 8. Рекурсия | | | | | | | | | | | | | | | | |  |
|  | | | | | | | | | | | |  | | | | |  | |  |
|  | | Принцип «Разделяй и властвуй». Глубина рекурсии, прямой и обратный ход, рекуррентный и крайний случай. Ханойские башни. Генерация комбинаторных объектов. Перебор с возвратом. Рекурсивная генерация всех чисел длины M. Генерация всех перестановок. Примеры кодирования рекурсии: быстрое возведение в степень, НОД. | | | | | | | | | | | | | | | | |  |
|  | |  | |  | |  | |  | |  | |  | | | | |  | |  |
|  | | 9. Быстрые сортировки | | | | | | | | | | | | | | | | |  |
|  | | | | | | | | | | | |  | | | | |  | |  |
|  | | Быстрая сортировка Тони Хоара. Слияние двух упорядоченных массивов. Сортировка слиянием. Неустойчивость сортировок. | | | | | | | | | | | | | | | | |  |
|  | |  | |  | |  | |  | |  | |  | | | | |  | |  |
|  | | 10. Двоичный поиск | | | | | | | | | | | | | | | | |  |
|  | | | | | | | | | | | |  | | | | |  | |  |
|  | | Бинпоиск. Поиск корня непрерывной функции методом деления пополам. Бинарный поиск по ответу. Бинарный поиск в массиве за O(logN). | | | | | | | | | | | | | | | | |  |
|  | |  | |  | |  | |  | |  | |  | | | | |  | |  |
|  | | 11. Динамическое программирование | | | | | | | | | | | | | | | | |  |
|  | | | | | | | | | | | |  | | | | |  | |  |
|  | | Вычисление чисел Фибоначчи и проблема перевычислений. Рекурсия с кешированием. Одномерное динамическое программирование. Задачи о Кузнечике. Восстановление пути минимальной стоимости. | | | | | | | | | | | | | | | | |  |
|  | |  | |  | |  | |  | |  | |  | | | | |  | |  |
|  | | 12. Строки | | | | | | | | | | | | | | | | |  |
|  | | | | | | | | | | | |  | | | | |  | |  |
|  | | Тип str. Неизменяемость строки. Наивный поиск подстроки в строке. Методы строк find, rfind, count, replace. Методы split и join. Разбиение на подстроки, объединение. Срезы строк. Префикс-функция. Алгоритм Кнута-Морриса-Пратта. | | | | | | | | | | | | | | | | |  |
|  | |  | |  | |  | |  | |  | |  | | | | |  | |  |
|  | | 13. Двумерное динамическое программирование | | | | | | | | | | | | | | | | |  |
|  | | | | | | | | | | | |  | | | | |  | |  |
|  | | Вычисление расстояния Левенштейна. Восстановление последовательности редакционных изменений. Наибольшая общая подпоследовательность. Наибольшая возрастающая подпоследовательность. | | | | | | | | | | | | | | | | |  |
|  | |  | |  | |  | |  | |  | |  | | | | |  | |  |
|  | | 14. Структуры FIFO и LIFO | | | | | | | | | | | | | | | | |  |
|  | | | | | | | | | | | |  | | | | |  | |  |
|  | | Очереди: FIFO и LIFO. Стек как очередь LIFO. Проверка корректности скобочной последовательности. Обратная польская нотация. | | | | | | | | | | | | | | | | |  |
|  | |  | |  | |  | |  | |  | |  | | | | |  | |  |
|  | | 15. Конечные и клеточные автоматы | | | | | | | | | | | | | | | | |  |
|  | | | | | | | | | | | |  | | | | |  | |  |
|  | | Машина Тьюринга. Конечный автомат как её упрощение. Конечный автомат для поиска подстроки «abcd». Простейшие клеточные автоматы. Игра «Жизнь» Джона Конвея. | | | | | | | | | | | | | | | | |  |
|  | |  | |  | |  | |  | |  | |  | | | | |  | |  |
| Семестр: 2 (Весенний) | | | | | | | | | | | |  | | | | |  | |  |
|  | |  | |  | |  | |  | |  | |  | | | | |  | |  |
|  | | 16. Сложность задач | | | | | | | | | | | | | | | | |  |
|  | | | | | | | | | | | |  | | | | |  | |  |
|  | | Краткое повторение синтаксиса Python. Сложность задач. Детерминированная и недетерминированная машина Тьюринга. Алгоритмически простые и сложные задачи (классы P и NP). Классы NP-complete и NP-hard. | | | | | | | | | | | | | | | | |  |
|  | |  | |  | |  | |  | |  | |  | | | | |  | |  |
|  | | 17. Хеширование | | | | | | | | | | | | | | | | |  |
|  | | | | | | | | | | | |  | | | | |  | |  |
|  | | Хеш-функции, хеширование и хеш-таблицы. Что такое хеш-функция. Примеры. Использование хеширования для гарантии целостности файлов и хранения паролей. Полиномиальный хеш. Алгоритм Рабина-Карпа. Открытая и закрытая хеш-таблицы. Проблема удаления из закрытой хеш-таблицы. Перехеширование. Реализация закрытой хеш-таблицы. | | | | | | | | | | | | | | | | |  |
|  | |  | |  | |  | |  | |  | |  | | | | |  | |  |
|  | | 18. Словари и множества в Python | | | | | | | | | | | | | | | | |  |
|  | | | | | | | | | | | |  | | | | |  | |  |
|  | | Словари и множества в Python. Множество set. Создание и изменение множеств. Работа с элементами. Тип frozenset и зачем он нужен. Операции с множествами, обычные для математики. Словарь dict. Создание и изменение словаря. Пример применения ассоциативного массива. Defaultdict, OrderedDict. | | | | | | | | | | | | | | | | |  |
|  | |  | |  | |  | |  | |  | |  | | | | |  | |  |
|  | | 19. Связные списки | | | | | | | | | | | | | | | | |  |
|  | | | | | | | | | | | |  | | | | |  | |  |
|  | | Кортежи tuple и контейнер NamedTuple. Списки: односвязный, двусвязный, кольцо (реализация ч/з словари). | | | | | | | | | | | | | | | | |  |
|  | |  | |  | |  | |  | |  | |  | | | | |  | |  |
|  | | 20. Очередь и очередь с приоритетами | | | | | | | | | | | | | | | | |  |
|  | | | | | | | | | | | |  | | | | |  | |  |
|  | | Очередь и дек (реализация на списках). Контейнер Deque. Куча (повторение). Сортировка кучей. Модуль heapq. | | | | | | | | | | | | | | | | |  |
|  | |  | |  | |  | |  | |  | |  | | | | |  | |  |
|  | | 21. Основы теории графов | | | | | | | | | | | | | | | | |  |
|  | | | | | | | | | | | |  | | | | |  | |  |
|  | | Введение в теорию графов. Инцидентность, смежность, петля, кратные рёбра, подграф. Эйлеров цикл. Эйлеров путь. Пути в графах. Циклы. Простые пути и циклы. Связность графов. Компоненты связности. Взвешенный граф. Орграфы. Компоненты сильной связности орграфа. Ориентированные ациклические графы. Дерево. Корневое дерево. Остовное дерево графа. | | | | | | | | | | | | | | | | |  |
|  | |  | |  | |  | |  | |  | |  | | | | |  | |  |
|  | | 22. Хранение графа в памяти | | | | | | | | | | | | | | | | |  |
|  | | | | | | | | | | | |  | | | | |  | |  |
|  | | Список рёбер, матрица смежности и списки смежности. Реализация этих способов и асимптотика их работы. Переходы между различными формами хранения графа. Компактная форма хранения списка смежности для константного графа. Хранение деревьев в памяти. | | | | | | | | | | | | | | | | |  |
|  | |  | |  | |  | |  | |  | |  | | | | |  | |  |
|  | | 23. Поиск в глубину | | | | | | | | | | | | | | | | |  |
|  | | | | | | | | | | | |  | | | | |  | |  |
|  | | Обход графа в глубину. Выделение компонент связности (обходом в глубину). Выделение компонент сильной связности орграфа. Проверка двудольности графа. Проверка графа на ацикличность и нахождение цикла. Топологическая сортировка. Поиск мостов и точек сочленения. | | | | | | | | | | | | | | | | |  |
|  | |  | |  | |  | |  | |  | |  | | | | |  | |  |
|  | | 24. Поиск в ширину | | | | | | | | | | | | | | | | |  |
|  | | | | | | | | | | | |  | | | | |  | |  |
|  | | Обход графа в ширину. Очередь при обходе в ширину и её асимптотика. Выделение компонент связности (обходом в ширину). Нахождение кратчайшего цикла в невзвешенном графе. | | | | | | | | | | | | | | | | |  |
|  | |  | |  | |  | |  | |  | |  | | | | |  | |  |
|  | | 25. Поиск кратчайшего пути | | | | | | | | | | | | | | | | |  |
|  | | | | | | | | | | | |  | | | | |  | |  |
|  | | Алгоритм Дейкстры поиска кратчайшего пути. Алгоритмы Флойда-Уоршелла и Беллмана-Форда. | | | | | | | | | | | | | | | | |  |
|  | |  | |  | |  | |  | |  | |  | | | | |  | |  |
|  | | 26. Остовные деревья | | | | | | | | | | | | | | | | |  |
|  | | | | | | | | | | | |  | | | | |  | |  |
|  | | Алгоритм Прима. Алгоритм Краскала. | | | | | | | | | | | | | | | | |  |
|  | |  | |  | |  | |  | |  | |  | | | | |  | |  |
|  | | 27. Основы теории игр | | | | | | | | | | | | | | | | |  |
|  | | | | | | | | | | | |  | | | | |  | |  |
|  | | Игры на ациклических графах. Игра «Ним». Сумма игр. Функция Шпрага-Гранди. | | | | | | | | | | | | | | | | |  |
|  | |  | |  | |  | |  | |  | |  | | | | |  | |  |
|  | | 28. Двоичные деревья поиска | | | | | | | | | | | | | | | | |  |
|  | | | | | | | | | | | |  | | | | |  | |  |
|  | | Двоичные деревья поиска. Асимптотика основных операций. Баланси-ровка деревьев. АВЛ-дерево и красно-чёрное дерево. Декартово дерево. | | | | | | | | | | | | | | | | |  |
|  | |  | |  | |  | |  | |  | |  | | | | |  | |  |
|  | | 29. Асимптотически сложные задачи на графах | | | | | | | | | | | | | | | | |  |
|  | | | | | | | | | | | |  | | | | |  | |  |
|  | | Гамильтонов граф. Построение гамильтонова цикла. Задачи о коммивояжере и о китайском почтальоне. Приближенные алгоритмы для NP-полных задач. | | | | | | | | | | | | | | | | |  |
|  | |  | |  | |  | |  | |  | |  | | | | |  | |  |
| **5. Описание материально-технической базы, необходимой для осуществления образовательного процесса по дисциплине (модулю)** | | | | | | | | | | | | | | | | | | |  |
|  | | | | | | | | | | | |  | | | | |  | |  |
|  | | Большая лекционная аудитория, подходящая для учебного потока (факультет, оснащённая мультимедиа проектором и экраном для чтения лекций. Учебные аудитории, учебный сетевой компьютерный класс с установленным необходимым программным обеспечением. | | | | | | | | | | | | | | | | |  |
|  | |  | |  | |  | |  | |  | |  | | | | |  | |  |
| **6.Перечень рекомендуемой литературы** | | | | | | | | | | | | | | | | | | |  |
|  | | | |  | |  | |  | |  | |  | | | | |  | |  |
| Основная литература | | | | | | | | | | | | | | | | |  | |  |
|  | | 1. Алгоритмы [Текст] : [учеб. пособие для вузов] / С. Дасгупта, Х. Пападимитриу, У. Вазирани ; пер. с англ. А. А. Куликова ; под ред. А. Шеня .— М. : МЦНМО, 2014 .— 320 с. | | | | | | | | | | | | | | | | |  |
|  | |  | |  | |  | |  | |  | |  | | | | |  | |  |
| Дополнительная литература | | | | | | | | | | | | | | | | |  | |  |
|  | | 1. Алгоритмы : построение и анализ [Текст] : учебник для вузов / Т. Кормен, Ч. Лейзерсон, Р. Ривест .— М. : МЦНМО, 1999 .— 263 с. | | | | | | | | | | | | | | | | |  |
|  | |  | |  | |  | |  | |  | |  | | | | |  | |  |
| **7. Перечень ресурсов информационно-телекоммуникационной сети "Интернет", необходимых для освоения дисциплины (модуля)** | | | | | | | | | | | | | | | | | | |  |
|  | | | | | | | | | | | | | | | | | | |  |
|  | | 1. https://python.org | | | | | | | | | | | | | | | | |  |
|  | | 2. https://e-maxx.ru/algo/ | | | | | | | | | | | | | | | | |  |
|  | | 2. https://github.com | | | | | | | | | | | | | | | | |  |
|  | | 3. http://judge.mipt.ru | | | | | | | | | | | | | | | | |  |
|  | | 4. http://acm.mipt.ru | | | | | | | | | | | | | | | | |  |
|  | |  | |  | |  | |  | |  | |  | | | | |  | |  |
| **8. Перечень информационных технологий, используемых при осуществлении образовательного процесса по дисциплине (модулю), включая перечень необходимого программного обеспечения и информационных справочных систем (при необходимости)** | | | | | | | | | | | | | | | | | | |  |
|  | |  | |  | |  | |  | |  | |  | | | | |  | |  |
|  | | На ПК в компьютерных классах должно быть установлено следующее ПО: | | | | | | | | | | | | | | | | |  |
|  | | 1. Операционная система GNU/Linux; | | | | | | | | | | | | | | | | |  |
|  | | 2. Интерпретатор Python версии не ниже 3.9; | | | | | | | | | | | | | | | | |  |
|  | | 3. Командный интерпретатор Ipython с отладчиком; | | | | | | | | | | | | | | | | |  |
|  | | 4. Среда разработки JetBrains Python Charm community edition; | | | | | | | | | | | | | | | | |  |
|  | | 5. Среда разработки IDLE; | | | | | | | | | | | | | | | | |  |
|  | | 6. Библиотеки SciPy, Pandas, seaborn, tkinter для Python 3; | | | | | | | | | | | | | | | | |  |
|  | | На лекциях используются мультимедийные технологии, включая демонстрацию презентаций. | | | | | | | | | | | | | | | | |  |
|  | | Для контроля и коррекции знаний, обучающиеся могут использовать компьютерное тестирование, в том числе на сайте judge.mipt.ru. | | | | | | | | | | | | | | | | |  |
|  | | В процессе самостоятельной работы обучающихся возможно использование любые среды программирования. | | | | | | | | | | | | | | | | |  |
|  | |  | | | | | | | | | | | | | | | | |  |
|  | |  | |  | |  | |  | |  | |  | | | | |  | |  |
| **9. Методические указания для обучающихся по освоению дисциплины (модуля)** | | | | | | | | | | | | | | | | | | |  |
|  | |  | |  | |  | |  | |  | |  | | | | |  | |  |
| Изложение материала происходит преимущественно на лекциях, сопровождается мультимедиа-презентацией с примерами кода и блок-схемами алгоритмов. На лабораторных занятиях также происходит изложение нового материала: в начале каждой лабораторной работы, по мере необходимости, а также в личных беседах преподавателя с учебными группами. На контестах изложение нового материала исключено, преподаватель оказывает только консультативное содействие для успешного решения задач. | | | | | | | | | | | | | | | | | | |  |
|  | | | | | | | | | | | | | | | | | | |  |
| Учёт, контроль и оценка знаний студентов | | | | | | | | | | | | | | | | | | |  |
| В течение семестра успеваемость отслеживается автоматически — по результатам контестов, а также преподавателем по своевременности выполнения лабораторных работ. Таким образом достигается раннее выявление отстающих студентов с передачей докладных в дирекцию. | | | | | | | | | | | | | | | | | | |  |
| Посещаемость лекций не отмечается, но контесты завязаны на материал лекций, что делает посещение лекций насущной необходимостью в течение семестра. | | | | | | | | | | | | | | | | | | |  |
| Дифференцированный зачёт принимает лектор или семинарский преподаватель другой учебной группы в устной форме с учётом оценки за работу в семестре, выставленной семинарским преподавателем. Устный ответ практически исключает списывание, показывает владение базовой терминологией предмета, умение говорить на языке информатики и алгоритмов, а также позволяет проверить знание сложных алгоритмов, которые долго программируются, но могут быть относительно легко устно объяснены. | | | | | | | | | | | | | | | | | | |  |
|  | | | | | | | | | | | | | | | | | | |  |
|  | | | | | | | | | | | | | | | | | | |  |
|  | | | | | | | | | | | | | | | | | |  | |
|  | | | | | | | | | | | | | | | | | |  | |
|  | | | | | | | | | | | | | | | | | |  | |
|  | | | | | | | | | | | | | | | | | |  | |
|  | **ПРИЛОЖЕНИЕ** | | | | | | | | | | | | | | | | |  | |
|  |  | |  | |  | |  | |  | |  | | | | |  | |  | |
|  |  | |  | | | | | | | |  | | | | | | |  | |
|  |  | |  | |  | |  | |  | |  | | | | |  | |  | |
|  |  | |  | |  | |  | |  | |  | |  |  | |  | |  | |
| **ОЦЕНОЧНЫЕ МАТЕРИАЛЫ ПО ДИСЦИПЛИНЕ (МОДУЛЮ)** | | | | | | | | | | | | | | | | | |  | |
|  |  | |  | |  | |  | |  | |  | | | | |  | |  | |
|  | | | | | | | | | | | | | | | | | |  | |
| **по направлению:** | | | Прикладные математика и физика | | | | | | | | | | | | | | | | |
| **профиль подготовки:** |  | | Физика перспективных технологий: электроника и квантовые технологии | | | | | | | | | | | | | | | | |
|  |  | | Физтех-школа Электроники, Фотоники и Молекулярной Физики | | | | | | | | | | | | | | | | |
|  | | | кафедра информатики и вычислительной математики | | | | | | | | | | | | | | | | |
| **курс:** | | | 1 | | | | | | | |  | | | | |  | |  | |
| **квалификация:** | | | бакалавр | | | | | | | |  | | | | |  | |  | |
|  |  | |  | |  | |  | |  | |  | | | | |  | |  | |
| Семестры, формы промежуточной аттестации: | | | | | | | | | | |  | | | | |  | |  | |
|  |  | | 1 (осенний) - Дифференцированный зачет | | | | | | | |  | | | | |  | |  | |
|  |  | | 2 (весенний) - Дифференцированный зачет | | | | | | | |  | | | | |  | |  | |
|  |  | |  | |  | |  | |  | |  | | | | |  | |  | |
| **Разработчики:** | | | | | | | | | | |  | | | | |  | |  | |
|  |  | |  | |  | |  | |  | |  | | | | |  | |  | |
| Т.Ф. Хирьянов, старший преподаватель | | | | | | | | | | | | | | | | | |  | |
| К.Д. Гольдштейн, старший 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-1.1 Анализирует задачу, выделяя этапы ее решения, действия по решению задачи | | | | |  |
| УК-1.2 Находит, критически анализирует и выбирает информацию, необходимую для решения поставленной задачи | | | | |  |
| УК-1.3 Рассматривает различные варианты решения задачи, оценивает их преимущества и недостатки | | | | |  |
| УК-1.4 Грамотно, логично, аргументированно формирует собственные суждения и оценки | | | | |  |
| УК-1.5 Определяет и оценивает практические последствия возможных вариантов решения задачи | | | | |  |
| ОПК-1 Способен применять фундаментальные знания, полученные в области физико-математических и (или) естественных наук, и использовать их в профессиональной деятельности | | | ОПК-1.1 Способен анализировать поставленную задачу, намечать пути ее решения | | | | |  |
| ОПК-1.2 Способен строить математические модели, производить количественные расчеты и оценки | | | | |  |
| ОПК-1.3 Способен определять границы применимости полученных результатов | | | | |  |
| 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-2.2 Знает и умеет применять численные математические методы и прикладное программное обеспечение для решения научных задач в профессиональной области | | | | |  |
| ОПК-2.3 Знает основные требования информационной безопасности | | | | |  |
| ОПК-4 Способен осуществлять сбор и обработку научно-технической и (или) технологической информации для решения фундаментальных и прикладных задач | | | ОПК-4.1 Владеет методами научного поиска и интеллектуального анализа информации при решении задач профессиональной деятельности | | | | |  |
| ОПК-4.2 Знает основные источники научно-технической и (или) технологической информации в области профессиональной деятельности | | | | |  |
| ОПК-4.4 Владеет навыками работы с компьютером и компьютерными сетями с целью получения, хранения и обработки научной (технической, технологической) информации | | | | |  |
| ОПК-5 Способен участвовать в проведении фундаментальных и прикладных исследований и разработок, самостоятельно осваивать новые теоретические, в том числе, математические методы исследований, и работать на современной экспериментальной научно-исследовательской, измерительно-аналитической и технологической аппаратуре | | | ОПК-5.1 Способен решать поставленные задачи в области теоретических и экспериментальных исследований и разработок | | | | |  |
| ОПК-5.2 Обладает способностью к освоению новых знаний на основе изучения литературы, научных статей и других источников | | | | |  |
| ОПК-5.3 Способен к профессиональной эксплуатации современной экспериментальной научно-исследовательской (измерительно-аналитической и технологической) аппаратуры | | | | |  |
| ПК-1 Способен планировать и проводить научные эксперименты (в избранной предметной области) и (или) теоретические (аналитические и имитационные) исследования | | | ПК-1.4 Умеет строить математические модели для описания и исследования процессов и явлений в соответствующих научных областях | | | | |  |
| ПК-1.8 Владеет навыками работы с современными языками программирования и программными пакетами для научных расчетов | | | | |  |
| ПК-2 Способен анализировать полученные в ходе научно-исследовательской работы данные и делать научные выводы (заключения) | | | ПК-2.1 Владеет методами статистической обработки и анализа научных данных | | | | |  |
| ПК-2.2 Умеет находить ключевые параметры, определяющие изучаемое явление, и производить численные оценки по порядку величины | | | | |  |
| ПК-3 Способен выбирать и применять подходящее оборудование, инструменты и методы исследований для решения задач в избранной предметной области | | | ПК-3.3 Умеет производить оценку точности численных методов, используемых на ЭВМ, вычислительной сложности используемых алгоритмов и объема требуемых вычислительных ресурсов | | | | |  |
| ПК-4 Способен критически оценивать применимость используемых методик и методов | | | ПК-4.3 Способен обосновать причинно-следственные отношения используемых понятий и моделей | | | | |  |
|  |  |  |  |  |  |  |  |  |
| **2. Показатели оценивания компетенций** | | | | | |  |  |  |
|  |  |  |  |  |  |  |  |  |
| В результате изучения дисциплины «Информатика» обучающийся должен: | | | | | | | |  |
|  |  |  |  |  |  |  |  |  |
| **знать:** | |  |  |  |  |  |  |  |
| • Основы теории алгоритмов; • свойства алгоритмов, проблемы алгоритмической сложности и алгоритмической неразрешимости; • основы дискретной математики; • основы алгоритмического языка программирования Python; • общие характеристики интерпретируемых и компилируемых языков программирования; • общие понятия о структурах данных: стеки, очереди, списки, деревья, таблицы; • приёмы разработки программ; • принципы программирования структур данных для современных программ, типовые решения, применяемые для создания программ; • основы работы с пакетами прикладных программ в области математики и физики. | | | | | | | |  |
| **уметь:** | |  |  |  |  |  |  |  |
| • Выбирать оптимальные алгоритмы для современных программ; • разрабатывать полные законченные программы на одном из языков высокого уровня; программы на одном или нескольких языках программирования, как индивидуально, так и в команде, с использованием современных средств написания и отладки программ; • использовать знания по информатике для приложений в инновационной, конструкторско-технологической и производственно-технологической сферах деятельности; • работать как на уровне языка командного интерпретатора, так и с использованием графического пользовательского интерфейса; • использовать сигналы и оконные сообщения для взаимодействия процессов между собой и с операционной системой; • создавать безопасные программы, использовать современные средства для написания и отладки программ; • работать с пакетами прикладных программ, включая использование развитых графических возможностей этих пакетов. | | | | | | | |  |
| **владеть:** | |  |  |  |  |  |  |  |
| • Языком программирования Python и методами создания программ с использованием стандартных библиотек; • средствами отладки программ на Python; • навыками программирования с использованием средств операционной системы для решения исследовательских задач; • основами работы с прикладными пакетами Python и принципами написания дополнительных модулей; • навыками освоения современных архитектур ЭВМ. | | | | | | | |  |
|  |  |  |  |  |  |  |  |  |
| **3. Перечень типовых (примерных) вопросов, заданий, тем для подготовки к текущему контролю** | | | | | | | |  |
|  |  |  |  |  |  |  |  |  |
| 1) Операторы if, elif, else. Цикл while, операторы break, continue, else. | | | | | | | |  |
| 2) Решето Эратосфена. Оценка временной сложности алгоритма. | | | | | | | |  |
| 3) Задача упорядочивания элементов в массиве. Оценка временной сложности задачи в общем случае. Проверка упорядоченности массива за O(N). | | | | | | | |  |
| 4) Рекурсия. Прямой и обратный ход рекурсии. Стек вызовов при рекурсии. Вычисление факториала. | | | | | | | |  |
| 5) Наибольшая возрастающая подпоследовательность. | | | | | | | |  |