

Семестр 1

Лекции:

1. Парадигмы программирования. Ссылочная модель данных и типизация в Python. 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()`. Оператор ветвления `if`. Переменные-счётчики. Целочисленное деление и взятие остатка, их отличие в C++ и Python. Вложенные ветвления. Каскадные ветвления `if-elif-else`. Цикл `while`. Инструкции `break` и `continue`.

2. Архитектура ПК. Регистр команд, хранение команд в памяти. Ход исполнения программы: от кода к транзистору. Высокоуровневые и низкоуровневые языки, работа с памятью и процессором. Алгебра логики. Тип `bool`. Логические операции. Битовые операции `&`, `|`, `^`. Позиционные системы счисления и литералы целых чисел в Python.

3. Интерактивный режим и работа в консоли. Обработка потока чисел с терминальным элементом. Поиск числа в потоке. Фильтрация потока чисел. Переменные-флаги. Максимальное число в потоке. Местоположение максимума. Количество равных максимуму. Поиск трёх максимумов за один проход. Однопроходные алгоритмы: сумма, произведение. Анализ цифр числа в произвольной системе счисления. Генерирующие алгоритмы: арифметическая, геометрическая прогрессии, арифметико-геометрическая прогрессия.

4. 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. Работа со строками в Python. Неизменяемость строки. Наивный поиск подстроки в строке. Методы строк `find`, `rfind`, `count`, `replace`. Методы `split` и `join`. Разбиение на подстроки, объединение. Срезы строк.

5. Массивы и списки. 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. Концепция `deepcopy`. Циклический сдвиг в массиве. Обращение массива.

6. Изменяемость списка `list` в Python. 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.

7. Словари и множества в Python. Частотный анализ текста. Логические операции с множествами. Подходы к тестированию и дебаггину программы. `Except` и `assert` в


Python. Использование Python для математических вычислений. Матричные вычисления в Numpy.

8. Рекурсия. Принцип «Разделяй и властвуй»: поиск медианы. 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, бинарный поиск.


9. Вычисление чисел Фибоначчи и проблема перевычислений. Рекурсия с кэшированием. Динамическое программирование. Задачи о Кузнечике. Восстановление пути минимальной стоимости. Двумерное динамическое программирование. Вычисление расстояния Левенштейна. Наибольшая общая подпоследовательность. Наибольшая возрастающая подпоследовательность.


10. Концепции функционального и императивного программирования. Мотивация ООП, три парадигмы ООП. Реализация ООП в Python: инкапсуляция, полиморфизм.

11. Наследование. Иерархии наследования, особенности наследования в Python.

 12. Очереди: FIFO и LIFO, реализация как классов. Пирамида (heap). Пирамида как очередь FIFO с приоритетами. Стек как очередь LIFO. Дек. Проверка корректности скобочной последовательности. Прямая и обратная польская нотация.

13. Оценка сложности алгоритмов. Простые сортировки (пузырьковая, вставками, выбором). Устойчивость сортировок.

 14. Теорема о сложности сортировок на сравнениях. Сортировка слиянием, сортировка Хоара. Сортировка подсчетом.

 15. Хеширование. Хеш-функции. Алгоритм Рабина-Карпа. Хеш-таблицы, открытая и закрытая. Энтропия Шеннона. RSA, ECC, SHA-256.

Семинары:

1. Распределительный контекст.
2. Основы работы в консоли. Калькулятор. Вариант с запуском из интерактивного режима, запуском из консоли с консольными аргументами, считыванием-записью файла. Параллельно – знакомство с числовыми типами данных, списками и math.
3. Git и Github. Обработка текста (частотный анализ, фильтрация etc). Параллельно – знакомство со строками и списками, устройством файловой системы.
4. Обработка лабораторной работы. Параллельно – первое знакомство с функциями, списками и numpy
5. Построение графиков, работа с БД. Знакомство с matplotlib, pandas

6. Контекст: логические функции, системы счисления, ветвления и циклы, НОД и НОК. По теории: декомпозиция алгоритмов. Написание и использование функций. Параметры функций.
7. Простейший графический интерфейс в tkinter. Калькулятор с графическим интерфейсом, интерактивный обработчик лаб. Декораторы функций.
8. Введение в ООП. Модель солнечной системы.
9. Контекст: работа со списками, строками, словарями и множествами.
10. Написание простой игры (пушка, амеба, "жизнь"). Декораторы в классах. Интерактивный вывод данных
11. Реализация пирамиды с балансировкой. Перевод выражений из инфиксной в прямую и обратную польскую нотации. Стековый калькулятор.
12. Разбор задач на ДП (восстановление пути, вариации задачи о рюкзаке)
13. Контекст: рекурсия, ДП, классы и сортировки.
14. Криптографический анализ. Шифр Цезаря. Частотный анализ. Шифр Виженера. Реализация простых хеш функций для целых чисел, чисел с плавающей запятой, строк.

Семестр 2

Лекции:

1. Повторение сортировок. Сортировка кучей. Биномиальная и фибоначчиева куча.
2. Амортизационный анализ, асимптотика стандартных операций со структурами данных.
3. Дерево отрезков, дерево Фенвика.
4. Двоичные деревья поиска. Асимптотика основных операций. Балансировка деревьев. АВЛ-дерево и красно-чёрное дерево. Декартово дерево.
5. Детерминированная и недетерминированная машина Тьюринга. Алгоритмически простые и сложные задачи (классы P и NP). Классы NP-complete и NP-hard. Строковые алгоритмы.
6. Еще строковые алгоритмы.
7. 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.
8. 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.

9. Обход графа в глубину. Выделение компонент связности (обходом в глубину). Выделение компонент сильной связности орграфа (алгоритм Косарайю). Проверка двудольности графа. Проверка графа на ацикличность и нахождение цикла. Топологическая сортировка. Поиск мостов и точек сочленения.

10. Обход графа в ширину. Очередь при обходе в ширину и её асимптотика. Выделение компонент связности (обходом в ширину). Нахождение кратчайшего цикла в невзвешенном графе.

11. Поиск кратчайшего пути. Алгоритм Дейкстры поиска кратчайшего пути. Алгоритмы Флойда-Уоршелла и Беллмана-Форда.

12. Остовные деревья. Алгоритм Прима. Алгоритм Краскала.

13. Паросочетания. Алгоритм Куна, Теорема Кенига, Теорема Холла.

14. Потоки и сети в графах.

15. Асинхронное программирование и многопоточность в Python.

Семинары:

Семинары во 2 семестре предполагают разбор рассматриваемых на лекции алгоритмов в задачах, представленных в “листочках” к семинару, и 4 конкурса с плавающим временем сдачи, код-ревью и жестким антиплагиатом:

Конкурс 1 – темы 1-4

Конкурс 2 – темы 6-8

Конкурс 3 – темы 9-10

Конкурс 4 – темы 11-13