**Программирование на языке Python. Уровень 2. Продвинутый курс.**

**Продолжительность не менее 40 ак. час.**

**(программа идентична для дистанционного и очного обучения)**

|  |
| --- |
| **Модуль 1. Функциональное программирование** |
| * Функция как объект. * Анонимные функции. Сортировка коллекций по ключу * Применение функций all и any * Паттерн «Фабрика функций» * Функции высшего порядка (map, filter, reduce) * Модуль itertools и комбинаторика * Практические задачи |
| **Модуль 2. Замыкания и функции декораторы** |
| * Глобальный и локальный контекст * Замыкания функций * Декораторы функций * Встроенные декораторы * Функция-генератор. Оператор yield * Практические задачи |
| **Модуль 3. Алгоритмы**   * Жадные алгоритмы * Алгоритмы на графах * Алгоритмы поиска (BFS, DFS, Dijkstra) * Понятие сложности алгоритма * Практические задачи |
| **Модуль 4. Введение в ООП** |
| * Эволюция парадигм программирования и актуальность ООП * Класс и экземпляр класса * Методы классов * Конструктор класса * Инкапсуляция и доступ к содержимому класса * Практические задачи   **Модуль 5. Предопределённые магические (dunder) методы в ООП** |
| * Метод \_\_call\_\_. Функторы и классы-декораторы * Метод \_\_new\_\_. Паттерн синглетон * Паттерн matching * Слоты. Коллекция \_\_slots\_\_ * Паттерн моносостояние * Практические задачи   **Модуль 6. Магические методы и перегрузка операторов**   * Магические методы \_\_add\_\_, \_\_sub\_\_, \_\_mul\_\_ * Методы сравнений \_\_eq\_\_, \_\_ne\_\_, \_\_lt\_\_, \_\_gt\_ * Магические методы \_\_eq\_\_ и \_\_hash\_\_ * Магический метод \_\_bool \_\_ определения правдивости объектов * Магические методы \_\_getitem\_\_, \_\_setitem\_\_ и \_\_delitem\_\_ * Генераторы и итераторы. Магические методы \_\_iter\_\_ и \_\_next\_\_ * Практические задачи |
| **Модуль 7. Наследование** |
| * Понятия базового класса и производного класса. Утиная типизация * Наследование от встроенных типов * Атрибуты private и protected при наследовании. * Практические задачи   **Модуль 7. Полифорфизм и абстрактные классы**.   * Суть полиморфизма * Абстрактные методы * Класс Enum * Методы класса и статические методы * Практические задачи |
| **Модуль 8. Множественное наследование и абстракция** |
| * Множественное наследование * Миксины * Декрипторы классов * Абстрактные классы * Наследование исключений * Менеджеры контекстов. Оператор with * Практические задачи   **Модуль 9. Отношения между классами. Понятие об объектно-реляционном соответствии**   * Библиотечный модуль pickle. Применение * Концепция композиции и агрегации * Применение объектно-реляционного отображения. * Отношения между классами на примере работы c ORM SQLAlchemy * Практические задачи   **Модуль 10. Концепции ОО-проектирования. ОО-дизайн** |
| * Вложенные классы * Пользовательские мета-классы. Объект type * Декорирование классами * Применение dataclasses * Дескрипторы классов * Принципы проектирования Solid * Итоговая практическая работа |