

СОВРЕМЕННЫЕ ЯЗЫКИ ПРОГРАММИРОВАНИЯ

1. Цель освоения дисциплины

Сформировать систему компетенций бакалавра на основе изучения наиболее современных языков программирования и практики использования полученных теоретических знаний для решения задач профессиональной деятельности.

2. Место дисциплины в структуре ОПОП

Дисциплина «Современные языки программирования» относится к базовой части блока дисциплин.

Для освоения дисциплины «Современные языки программирования» обучающиеся используют знания, умения, способы деятельности и установки, сформированные в ходе изучения дисциплин «Графика», «Дискретная математика», «ИКТ и медиаинформационная грамотность», «История науки и техники», «Математика», «Машиностроительное черчение», «Основы материаловедения», «Основы стандартизации, метрологии и сертификации», «Основы цифровой экономики», «Программирование», «Техническая эстетика и дизайн», «Технологии обработки конструкционных материалов», «Философия», «3D-моделирование и печать», «Введение в информатику», «Веб-дизайн и разработка интернет-приложений», «Компьютерная графика и мультимедиа технологии», «Образовательная робототехника», «Обустройство и дизайн дома», «Технологический практикум по обработке конструкционных материалов», «Технологический практикум по обработке тканей и пищевых продуктов», прохождения практик «Производственная (исследовательская) практика», «Учебная (технологическая) практика».

Освоение данной дисциплины является необходимой основой для последующего изучения дисциплин «Администрирование компьютерных систем», «Архитектура компьютера», «Информационные системы», «Методика обучения информатике», «Основы искусственного интеллекта», «Основы исследований в технологическом образовании», «Основы творческо-конструкторской деятельности», «Перспективные материалы и технологии», «Профориентационная работа в старших классах», «Теоретические основы информатики», «Теория вероятностей и математическая статистика», «Технологии современного производства», «Технологические и транспортные машины», «3D-моделирование и прототипирование в технологическом образовании», «Декоративно-оформительское искусство», «Информационные технологии в управлении образованием», «Практикум решения задач по информатике», «Ремонт и эксплуатация дома», «Робототехнические системы в быту», «Технологический практикум по обработке конструкционных материалов», «Технологический практикум по обработке тканей и пищевых продуктов», «Экологические основы производства и защита окружающей среды», прохождения практик «Научно-исследовательская работа», «Производственная (педагогическая) практика (Информатика)», «Производственная (педагогическая) практика (Технология)», «Производственная (преддипломная) практика», «Учебная (производственно-технологическая) практика».

3. Планируемые результаты обучения

В результате освоения дисциплины выпускник должен обладать следующими компетенциями:

- способен осуществлять поиск, критический анализ и синтез информации, применять системный подход для решения поставленных задач (УК-1);
- способен применять предметные знания в образовательном процессе (ПК-3).

В результате изучения дисциплины обучающийся должен:

знать

- особенности реализации объектно-ориентированного подхода в языке Python;
- концепцию, основные понятия и конструкции функционального программирования;
- особенности реализации функционального подхода в языке Python;

уметь

- разрабатывать объектно-ориентированные приложения на языке Python;
- разрабатывать приложения на языке Python в соответствии с функциональным подходом;

владеть

- навыками объектно-ориентированного программирования на языке Python;
- опытом разработки приложений на языке Python в соответствии с функциональным подходом.

4. Общая трудоёмкость дисциплины и её распределение

количество зачётных единиц – 2,

общая трудоёмкость дисциплины в часах – 72 ч. (в т.ч. аудиторных часов – 28 ч., СРС – 40 ч.),

распределение по семестрам – 5,

форма и место отчётности – аттестация с оценкой (5 семестр).

5. Краткое содержание дисциплины

Объектно-ориентированное программирование на языке Python.

Классы. Экземпляры классов. Атрибуты. Методы. Скрытие данных. Свойства.

Полиморфизм. Перегрузка операций. Наследование. Множественное наследование.

Абстрактные методы. Метаклассы.

Функциональное программирование на языке Python.

Парадигма функционального программирования. Функции в Python, типы аргументов.

Способы вызова функций. Рекурсия. Лямбда-выражения (анонимные функции). Функции

как параметры и как результат. Функции для обработки последовательностей. Списковые включения. Итераторы. Генераторы. Генераторные выражения.

6. Разработчик

Маркович Ольга Сергеевна, кандидат педагогических наук, доцент кафедры информатики и методики преподавания информатики ФГБОУ ВО «ВГСПУ»,

Усольцев Вадим Леонидович, кандидат физико-математических наук, доцент кафедры информатики и методики преподавания информатики ФГБОУ ВО «ВГСПУ».