

Аннотация рабочей программы дисциплины (модуля)

Б1.О.18 Языки и методы программирования

Цели дисциплины

Целями освоения дисциплины Языки и методы программирования является формирование профессиональных и общепрофессиональных компетенций будущих специалистов в области прикладной математики; ознакомление с общими принципами построения и использования языков программирования, а также развитие навыков проектирования и реализации алгоритмов решения практических задач на одном из объектно-ориентированных языков программирования.

Задачи дисциплины

Основными задачами изучения дисциплины являются:

- изучение принципов алгоритмизации и современных методов обработки информации с использованием языков программирования;
- ознакомление с техническими, алгоритмическими, программными и технологическими решениями, используемыми в данной области;
- выработка практических навыков аналитического и экспериментального исследования основных методов и средств, используемых в области, изучаемой в рамках данной дисциплины.

Формируемые компетенции и индикаторы их достижения по дисциплине

Коды компетенции	Содержание компетенции	Код и наименование индикатора достижения компетенции
ОПК-2	Способен использовать и адаптировать существующие математические методы и системы программирования для разработки и реализации алгоритмов решения прикладных задач	ОПК-2.1 Знать существующие математические методы и системы программирования для разработки и реализации алгоритмов решения прикладных задач. ОПК -2.2 Уметь: решать стандартные профессиональные задачи с применением математических методов и систем программирования ОПК-2.3 Иметь навыки разработки и реализации алгоритмов решения прикладных задач.
ОПК-4	Способен решать задачи профессиональной деятельности с использованием существующих информационно-коммуникационных технологий и с учетом основных требований информационной безопасности	ОПК-4.1 Знать существующие информационно-коммуникационные технологии и требования информационной безопасности. ОПК -4.2 Уметь: решать стандартные профессиональные задачи с использованием существующих информационно-коммуникационных технологий и с учетом основных требований информационной безопасности. ОПК-4.3 Иметь навыки разработки и реализации алгоритмов решения прикладных задач с использованием существующих информационно-коммуникационных технологий и с учетом основных требований информационной

		безопасности.
ПКС-2	Способен проектировать структуры данных	ПКС-2.1 Знать основные принципы проектирования структур данных. ПКС -2.2 Уметь использовать принципы проектирования структур данных. ПКС-2.3 Иметь навыки проектирования структур данных.
ПКС-4	Способен проектировать программные интерфейсы	ПКС-4.1 Знать основные принципы проектирования программных интерфейсов. ПКС -4.2 Уметь использовать принципы проектирования программных интерфейсов. ПКС-4.3 Иметь навыки проектирования программных интерфейсов.

Содержание разделов дисциплины

1 семестр

Тема 1. Основные понятия языков программирования

Введение в Delphi. Особенности и возможности языка Delphi. Базовые элементы языка: алфавит, словарь, константы, переменные. Иерархия типов данных. Выражения, операции, операнды. Структура программы (проекта): основные разделы, их назначение. Комментарии. Директивы компилятора.

Тема 2. Синтаксис, семантика, формальные способы описания языков программирования.

Простые типы данных: стандартные скалярные и пользовательские. Конструкции языков программирования. Операторы языка.

Тема 3. Типы данных, способы и механизмы управления данными.

Структурированные типы данных: массивы (статические, динамические, параметры-массивы. Алгоритмы информационного поиска и сортировки), строки (статические, динамические), множества, записи, файлы (типизированные, нетипизированные, текстовые). Способы описания, основные процедуры и функции обработки. Процедуры и функции, определяемые пользователем. Механизм передачи параметров. Рекурсивные подпрограммы.

Тема 4. Динамическая память и указатели. Динамические структуры данных.

Ссылочные типы и указатели. Представление динамических структур с помощью указателей.

Тема 5. Модульное программирование.

Общая структура модуля. Подпрограммы в модулях. Компиляция и использование модулей. Типы модулей. Создание приложений, содержащих несколько форм.

Тема 6. Создание динамических библиотек.

Особенности разработки динамических библиотек.

Тема 7. Основные понятия объектно-ориентированного программирования

Парадигмы объектно-ориентированного программирования: инкапсуляция, полиморфизм, наследование. Понятие класса, объекта. Поля, методы. Конструкторы, деструкторы, свойства. Представление объекта в памяти.

Тема 8. Обработка исключений

Использование классов общего назначения. Класс исключений. Защищаемые блоки. Создание собственных исключений.

2 семестр

Тема 1. Язык программирования C#.

Пространство имен. Типы данных. Операции языка. Типы как классы. Парадигмы

объектно-ориентированного программирования в C#. Особенности реализации.

Тема 2. Операторы языка C#.

Операторы помеченные (labeled-statement), декларирующие (declaration-statement), встроенные (embedded-statement). Особенности реализации операторов в языке C#.

Тема 3. Массивы в C#.

Массивы одномерные, многомерные. Массивы массивов. Прямоугольные массивы. Особенности реализации массивов в языке C#.

Тема 4. Строки в C#.

Строки как объекты класса string. Строка как контейнер. Применение строк в переключателях. Массивы строк. Операции над строками. Особенности реализации строк в языке C#.

Тема 5. Методы в C#.

Методы–процедуры и методы-функции. Соответствие фиксированных параметров и аргументов. Параметры с типами ссылок. Методы с переменным числом аргументов. Перегрузка методов. Рекурсивные методы. Особенности реализации методов в языке C#.

3 семестр

Тема 1. Классы как типы.

Объявление класса. Поля объектов. Методы объектов. Ссылка this. Конструкторы объектов класса (умолчания, общего вида, копирования, приведения). Деструкторы и финализаторы.

Тема 2. Средства взаимодействия с объектами.

Принцип инкапсуляции и методы объектов. Свойства классов. Автореализуемые свойства. Индексаторы.

Тема 3. Включение, вложение и наследование классов.

Включение объектов классов. Вложение классов. Наследование классов. Доступность членов класса при наследовании. Методы при наследовании. Абстрактные методы и абстрактные классы. Применение абстрактных классов. Опечатанные классы и методы.

Тема 4. Интерфейсы.

Интерфейс как механизм наследования специфицированной функциональности. Наследование специфицированной функциональности. Реализация интерфейсов. Интерфейс как тип. Интерфейсы и наследование.