

РАБОЧАЯ ПРОГРАММА УЧЕБНОЙ ДИСЦИПЛИНЫ (МОДУЛЯ) ПРОГРАММИРОВАНИЕ

1. Цель учебной дисциплины (модуля) – теоретическая подготовка в области основных терминов и понятий процедурной и объектно-ориентированной парадигм программирования и привитие навыков программирования на языках Pascal и Object Pascal.

2. Требования к уровню освоения учебной дисциплины (модуля).

Код и наименование компетенции	Код и наименование индикатора достижения компетенции	Планируемые результаты обучения
ПК-1 Способен осваивать и использовать теоретические знания и практические умения и навыки в предметной области при решении профессиональных задач	ИПК-1.1 Знает структуру, состав и дидактические единицы предметной области (преподаваемого предмета) ИПК-1.2 Умеет осуществлять отбор учебного содержания для его реализации в различных формах обучения в соответствии с требованиями ФГОС ОО ИПК-1.3 Демонстрирует умение разрабатывать различные формы учебных занятий, применять методы, приемы и технологии обучения, в том числе информационные	<p>знать:</p> <ul style="list-style-type: none"> - основные этапы производства программного продукта, принципы их построения; - приемы работы с инструментальными средствами, поддерживающими разработку программного обеспечения; - теоретические основы построения современных языков программирования, инструментальных и технологических средств разработки программного обеспечения; - способы записи алгоритмов. <p>уметь:</p> <ul style="list-style-type: none"> - конструировать программы с использованием различных алгоритмических языков; - применять современные методики и технологии для обеспечения качества учебно-воспитательного процесса. <p>владеть:</p> <ul style="list-style-type: none"> - навыками использования процедурного и объектно-ориентированного программирования в профессиональной деятельности.

3. Содержание учебной дисциплины (модуля).

Раздел 1. Понятие о программировании. Парадигмы программирования.

Алгоритмизация и программирования. Программа для ЭВМ, как средство реализации алгоритма. Высокоуровневое и низкоуровневое программирование. Парадигмы высокоуровневого программирования: процедурная, объектно-ориентированная, функциональная, рекурсивно-логическая. Понятие о структурном программировании.

Раздел 2. Введение в программирование на языке Паскаль.

Структура программы. Линейные программы. Алфавит языка. Простые и составные типы данных. Операторы ветвления и циклов. Процедуры и функции. Использование массивов. Одномерные и многомерные массивы. Сортировка массивов. Работа с векторами и матрицами.

Множества. Записи. Оператор присоединения. Записи с вариантами.
Понятие рекурсии. Рекурсивные алгоритмы.
Описания типов. Пользовательские типы. Интервальный и перечислимый типы.

Раздел 3. Основы объектно-ориентированного программирования.

Языки программирования и системы программирования. Понятие парадигмы. Процедурно-ориентированная парадигма, ее особенности, процедурные языки, принцип разделения данных и процедур, Паскаль – как язык структурного программирования. Понятие объекта, как структуры содержащей данные и процедуры – 2 подхода. Манифест ОО-систем. Свойства и методы объектов, классы и типы, свойства и методы класса и экземпляра, конструкторы и деструктуры. Инкапсуляция. Сообщения и события. Наследование, понятие абстрактных классов, иерархия или сеть классов. Обработка событий. Понятие визуального проектирования. Визуальное проектирование и ООП. Обзор основных объектно-ориентированных языков программирования: SmallTalk, C++, Java, Object Pascal.

Раздел 4. Object Pascal, как язык объектно-ориентированного программирования.

Понятие о визуальном проектировании.

Object Pascal, как результат эволюционных изменений языка Pascal. Типы данных языка Pascal, специфичные для Windows. Объекты и классы, ссылочная модель объекта, объявление класса, методы Create и Free – как конструктор и деструктор. Поддержка инкапсуляции – ключевые слова private, public и protected. Классы и модули. Ключевое слово Self. Методы и данные класса. Указатели на методы, как развитие идеи процедурного типа. Наследование, наследование и совместимость с типов. Виртуальные и динамические методы. Информация о типе на этапе выполнения. Отладка программы. Пошаговое выполнение, точки останова, просмотр значений переменных на этапе выполнения. Создание консольных приложений.

Проблема быстрой разработки интерфейса пользователя и попытки ее решения с помощью объектно-ориентированной идеологии (Turbo Vision, Windows GUI). Обзор современных средств визуального.

Рабочая программа учебной дисциплины составлена:

Стась А.Н., к.т.н., доцент, заведующий кафедрой информатики