

**Учебная дисциплина «Объектно-ориентированное программирование»**

<p>Место дисциплины в структурной схеме образовательной программы</p>	<p align="center">Образовательная программа бакалавриата (I ступень высшего образования) Специальность: 6-05-0533-09 Прикладная математика Цикл специальных дисциплин: компонент УВО</p>
<p><b>Краткое содержание</b></p>	<p>Предмет курса и содержание дисциплины, ее связь с другими дисциплинами. Две парадигмы программирования. Основные направления в программировании. Возникновение объектно-ориентированного программирования (ООП). Базовые принципы ООП. Основные положения объектной модели ее преимущества. Абстрагирование. Модульность. Иерархия. Типизация. Базовые конструкции объектно-ориентированных программ: классы и объекты. Инициализация и разрушение объекта. Компоненты класса. Конструкторы и деструкторы. Перегрузка и переопределение методов класса. Принцип инкапсуляции. Область действия класса и доступ к компонентам класса. Управление доступом к компонентам класса. Организация внешнего доступа к локальным компонентам класса. Интерфейсные (дружественные) методы. Статические и константные компоненты. Указатели и ссылки. Операторы для динамического выделения и освобождения памяти. Статические и динамические объекты. Проху-классы. Базовые и производные классы. Основные правила построения производных классов. Конструкторы и деструкторы при наследовании. Композиция и наследование. Простое и множественное наследование. Переопределение членов базового класса в производном. Понятие раннего и позднего связывания. Использование виртуального механизма для реализации принципа полиморфизма. Виртуальные методы класса и механизм их использования. Абстрактные классы их назначение и свойства. Введение в параметризованные классы. Параметризованные классы и методы, их свойства. Совместное использование параметризации и принципов наследования. Организация внешнего доступа к компонентам параметризованных классов. Параметризованные классы и статические члены. Реализация smart-указателей, свойств в языке, механизма транзакций. Задание значений параметров класса по умолчанию. Основы обработки исключений. Генерация исключений. Перехватывание исключений. Повторная генерация исключения. Обработка неожиданных исключений. Генерация исключений в конструкторах. Исключения и наследование. Спецификация исключений. Иерархия исключений стандартной библиотеки. Потоки, общее понятие. Организация ввода из потока и вывод в поток. Контроль состояния потока и исправление ошибок. Неформатированный ввод-вывод. Манипуляторы потоков (стандартные и определяемые пользователем). Файлы и потоки их взаимосвязь. Файлы последовательного и произвольного доступа. Организация ввода и вывода объектов. Введение в стандартную библиотеку шаблонов (классов коллекций), основные понятия. Классы контейнеры и итераторы. Типы контейнерных классов, адаптеры контейнеров. Алгоритмы и их использование с контейнерными классами. Назначение паттернов проектирования. Основные их виды. Связь паттернов проектирования с ранее изученным материалом. Применение ООП в решении экономических задач, задачах обработки информации и при выполнении инженерных расчетов.</p>
<p><b>Формируемые компетенции, результаты обучения</b></p>	<p>Базовые профессиональные компетенции: <i>знать</i>: базовые понятия и синтаксис языка, технологию ООП и приемы разработки программ; методы определения и использования</p>

	<p>основных объектов и конструкций языка; технологию организации и использования иерархии классов, predetermined классов и типов данных, методы ограничения доступа и обработки исключительных ситуаций; методы параметризации классов и их использование для решения задач; методы применения шаблонов и контейнерных абстракций; работу с потоками и разработку многопоточных приложений; <b>уметь:</b> определять абстракции, модули, строить иерархию классов для реализации программ; использовать методы: типизации, инкапсуляции, наследования, полиморфизма для разработки программных продуктов; использовать возможности стандартных библиотек; использовать механизм исключений для создания устойчивых приложений; создавать свои и использовать предоставляемые стандартные библиотеки шаблонов сложных структур данных; использовать технологию ООП для разработки сложных программ и систем; <b>владеть:</b> методами и инструментальными средствами и системами разработки объектно-ориентированных программ; техникой создания объектно-ориентированных программных компонент и организацией их взаимодействия в программных проектах..</p>
<b>Пререквизиты</b>	«Алгоритмы и структуры данных», «Основы алгоритмизации и программирования», «Математика»
<b>Трудоемкость</b>	3 зачетные единицы, 108 академических часов, из них 68 аудиторных: 34 ч лекций и 34 ч лабораторных занятий.
<b>Семестр(ы), требования и формы текущей и промежуточной аттестации</b>	6-й семестр, зачет