



## 1. Цели и задачи дисциплины

### Цели преподавания дисциплины:

Цель дисциплины – овладение математическими средствами исследования практической приемлемости конкретных алгоритмов; обучение студентов основам объектно-ориентированного программирования в современных средах разработки ПО.

Основной *задачей* изучения курса является получение знаний и практических навыков в области разработки объектно-ориентированных программ.

## 2. Требования к результатам освоения дисциплины (табл. из п.1 РП)

Код и наименование компетенции	Код и наименование индикатора достижения компетенции	Планируемые результаты обучения
<b>УК-1</b> Способен осуществлять поиск, критический анализ и синтез информации, применять системный подход для решения поставленных задач	<b>ИУК-1.1.</b> Знает принципы сбора, отбора и обобщения информации. <b>ИУК-1.2.</b> Умеет соотносить разнородные явления и систематизировать их в рамках избранных видов профессиональной деятельности. <b>ИУК-1.3.</b> Имеет практический опыт работы с информационными объектами и сетью Интернет, опыт научного поиска, опыт библиографического разыскания, создания научных текстов.	<b>знать:</b> основные понятия, определения и факты теории сложности алгоритмов; <b>уметь:</b> основные методы теории сложности алгоритмов; оценивание сложности алгоритмов. <b>владеть (иметь навыки):</b> навыками решения практических задач анализа алгоритмов; методами анализа сложности алгоритмов.
<b>УК-2</b> Способен определять круг задач в рамках поставленной цели и выбирать оптимальные способы их решения, исходя из действующих правовых норм, имеющихся ресурсов и ограничений	<b>ИУК-2.1.</b> Знает необходимые для осуществления профессиональной деятельности правовые нормы. <b>ИУК-2.2.</b> Умеет определять круг задач в рамках избранных видов профессиональной деятельности, планировать собственную деятельность исходя из имеющихся ресурсов; соотносить главное и второстепенное, решать поставленные задачи в рамках избранных видов профессиональной деятельности. <b>ИУК-2.3.</b> Имеет практический опыт применения нормативной базы и решения задач в области избранных видов профессиональной деятельности.	<b>знать:</b> - общие принципы построения алгоритмов, основные алгоритмические конструкции; - понятие системы программирования; - основные элементы процедурного языка программирования, структуру программы, операторы и операции, управляющие структуры, структуры данных, файлы, классы памяти; - подпрограммы, составление библиотек программ; - объектно-ориентированную модель программирования, понятие классов и объектов, их свойств и методов. <b>уметь:</b> применять современные инструментальные средства для разработки компонентов аппаратно-программных комплексов и баз данных применять современные технологии программирования для разработки компонентов аппаратно-программных комплексов и баз данных <b>владеть (иметь навыки):</b> математическими средствами исследования практической приемлемости конкретных алгоритмов.
<b>ОПК-1</b> Способен применять фундаментальные знания, полученные	<b>ИОПК-1.1.</b> Знает основные положения и концепции в данной области, базовые теории и основы материала, теории	<b>знать:</b> основные программные среды и средства разработки программ на языке C++ <b>уметь:</b>

<p>в области математических и (или) естественных наук, и использовать их в профессиональной деятельности</p>	<p>коммуникации; знает основную терминологию.  <b>ИОПК-1.2.</b> Умеет осуществлять первичный сбор и анализ материала, интерпретировать различные объекты.  <b>ИОПК-1.3.</b> Имеет практический опыт работы по решению стандартных задач и применяет его в профессиональной деятельности.</p>	<p>обосновывать выбор программного обеспечения и разрабатывать концептуальную и логическую модель данных  <b>владеть</b> (иметь навыки):  математическими средствами исследования практической приемлемости конкретных алгоритмов.</p>
<p><b>ОПК-2</b> Способен применять компьютерные/суперкомпьютерные методы, современное программное обеспечение, в том числе отечественного происхождения, для решения задач профессиональной деятельности</p>	<p><b>ИОПК-2.1.</b> Знает основные положения и концепции в области программирования, архитектуру языков программирования, теории коммуникации, знает основную терминологию, знаком с содержанием Единого Реестра Российских программ.  <b>ИОПК-2.2.</b> Умеет анализировать типовые языки программирования, составлять программы.  <b>ИОПК-2.3.</b> Имеет практический опыт решения задач анализа, интеграции различных типов программного обеспечения, анализа типов коммуникаций.</p>	<p><b>знать:</b>  - общие принципы построения алгоритмов, основные алгоритмические конструкции;  - понятие системы программирования;  - основные элементы процедурного языка программирования, структуру программы, операторы и операции, управляющие структуры, структуры данных, файлы, классы памяти;  - подпрограммы, составление библиотек программ;  - объектно-ориентированную модель программирования, понятие классов и объектов, их свойств и методов.  <b>уметь:</b>  - использовать языки программирования;  - строить логически правильные и эффективные программы;  - использовать нисходящий метод для решения задач.  <b>владеть</b> (иметь навыки):  математическими средствами исследования практической приемлемости конкретных алгоритмов.</p>
<p><b>ОПК-3</b> Способен к разработке алгоритмических и программных решений в области системного и прикладного программирования, математических, информационных и имитационных моделей, созданию информационных ресурсов глобальных сетей, образовательного контента, прикладных баз данных, тестов и средств тестирования систем и средств на соответствие стандартам и исходным требованиям</p>	<p><b>ИОПК-3.1.</b> Знает методы теории алгоритмов, методы системного и прикладного программирования, основные положения и концепции в области математических, информационных и имитационных моделей,  <b>ИОПК-3.2.</b> Умеет соотносить знания в области программирования, интерпретацию прочитанного, определять и создавать информационные ресурсы глобальных сетей, образовательного контента, средств тестирования систем.  <b>ИОПК-3.3.</b> Имеет практический опыт применения разработки программного обеспечения.</p>	<p><b>знать:</b>  основные понятия, определения и факты теории сложности алгоритмов;  <b>уметь:</b>  основные методы теории сложности алгоритмов; оценивание сложности алгоритмов.  подтверждать корректность работы программной системы путем организации модульного тестирования и представления результатов тестов, разрабатывать и тестировать программные компоненты с использованием этих средств,  <b>владеть</b> (иметь навыки):  навыками решения практических задач анализа алгоритмов; методами анализа сложности алгоритмов.</p>

### 3. Общая трудоемкость дисциплины – 5 з.е., 180 часов

#### ***4. Содержание дисциплины (дидактические единицы)***

Введение в объектно-ориентированное программирование. Основные принципы объектно-ориентированного программирования. Написание многопоточных приложений. Современные технологии объектно-ориентированного программирования.