

**МИНОБРНАУКИ РОССИИ**

**Федеральное государственное автономное образовательное  
учреждение высшего образования "Пермский  
государственный национальный исследовательский  
университет"**

**Кафедра радиоэлектроники и защиты информации**

Авторы-составители: **Карпов Сергей Борисович**  
**Лунегов Игорь Владимирович**

Рабочая программа дисциплины

**ОБЪЕКТНО-ОРИЕНТИРОВАННОЕ ПРОГРАММИРОВАНИЕ**

Код УМК 68690

Утверждено  
Протокол №4  
от «24» июня 2021 г.

Пермь, 2021

## **1. Наименование дисциплины**

Объектно-ориентированное программирование

## **2. Место дисциплины в структуре образовательной программы**

Дисциплина входит в обязательную часть Блока « Б.1 » образовательной программы по направлениям подготовки (специальностям):

Направление подготовки: **03.03.03** Радиофизика  
направленность Электроника, микро- и наноэлектроника

### **3. Планируемые результаты обучения по дисциплине**

В результате освоения дисциплины **Объектно-ориентированное программирование** у обучающегося должны быть сформированы следующие компетенции:

**03.03.03** Радиофизика (направленность : Электроника, микро- и наноэлектроника)

**ПК.3** Способен выбирать и моделировать архитектурные решения для реализации программной системы

#### **Индикаторы**

**ПК.3.1** Определяет архитектуры отдельных компонентов программной среды

**ПК.3.2** Разрабатывает архитектуру программного обеспечения программной системы

#### 4. Объем и содержание дисциплины

<b>Направления подготовки</b>	03.03.03 Радиофизика (направленность: Электроника, микро- и наноэлектроника)
<b>форма обучения</b>	очная
<b>№№ триместров, выделенных для изучения дисциплины</b>	4
<b>Объем дисциплины (з.е.)</b>	3
<b>Объем дисциплины (ак.час.)</b>	108
<b>Контактная работа с преподавателем (ак.час.), в том числе:</b>	42
<b>Проведение лекционных занятий</b>	14
<b>Проведение лабораторных работ, занятий по иностранному языку</b>	28
<b>Самостоятельная работа (ак.час.)</b>	66
<b>Формы текущего контроля</b>	Защищаемое контрольное мероприятие (3) Итоговое контрольное мероприятие (1)
<b>Формы промежуточной аттестации</b>	Зачет (4 триместр)

## 5. Аннотированное описание содержания разделов и тем дисциплины

### **Объектно-ориентированное программирование. Первый семестр**

Основные принципы ООП: инкапсуляция, наследование, полиморфизм.

Изучение основ ООП на примере языка C++

#### **Введение**

Процедурное и объектно-ориентированное программирование (ООП). Основные принципы ООП: инкапсуляция, наследование, полиморфизм. Язык описания UML. Объектно-ориентированный язык C++. Отличия языков C и C++: комментарии, константы, встраиваемые функции, объявление переменных, операторы new и delete, перегрузка функций, передача аргументов по умолчанию, приведение типов. Ссылки в C++. Использование ссылок в качестве аргументов и возвращаемого значения функций.

#### **Введение в классы**

Определение класса. Инкапсуляция. Члены класса. Член-данные и член-функции. Диаграмма класса. Открытые и закрытые члены класса. Спецификаторы доступа. Классы и структуры. Встраиваемые член-функции. Оператор расширения области видимости. Объявление объекта класса. Создание массивов объектов. Предварительное объявление объекта. Доступ к член-данным и член-функциям.

#### **Конструкторы и деструкторы**

Инициализация объектов. Конструкторы с параметрами. Конструкторы по умолчанию. Конструкторы копий. Деструкторы. Копирование объектов класса. Использование объектов класса в качестве аргументов и возвращаемого значения функций. Конструктор в качестве преобразователя типов. Объекты класса – члены другого класса. Список перегрузки инициализации.

#### **Перегрузка операций**

Операции в качестве функций. Правила перегрузки операций. Перегрузка бинарных и унарных операторов. Перегруженные операторы – члены класса и дружественные операторы. Ссылки в качестве аргументов и возвращаемого значения операторов-функций. Перегрузка оператора присваивания. Оператор вызова функции. Операции преобразования типов.

#### **Наследование и полиморфизм**

Базовый и производный классы. Спецификаторы доступа. Ключевое слово protected. Множественное наследование. Конструкторы и деструкторы при наследовании. Виртуальные базовые классы. Динамический полиморфизм. Доступ к объектам производных классов через указатель на базовый класс. Виртуальные функции. Полиморфные классы. Чистые виртуальные функции. Абстрактные базовые классы. Виртуальные деструкторы.

#### **Шаблоны**

Функции-шаблоны. Порожденные функции. Классы-шаблоны. Типизируемые параметры.

#### **Исключения**

Обработка исключительных ситуаций. Генерация и захват исключений. Ключевые слова try, catch, throw.

#### **Основы языка Java**

Обработка исключительных ситуаций. Генерация и захват исключений. Ключевые слова try, catch, throw.

#### **Диаграммы UML**

Использование диаграмм UML при разработке объектно-ориентированных программ.



## **6. Методические указания для обучающихся по освоению дисциплины**

Освоение дисциплины требует систематического изучения всех тем в той последовательности, в какой они указаны в рабочей программе.

Основными видами учебной работы являются аудиторские занятия. Их цель - расширить базовые знания обучающихся по осваиваемой дисциплине и систему теоретических ориентиров для последующего более глубокого освоения программного материала в ходе самостоятельной работы. Обучающемуся важно помнить, что контактная работа с преподавателем эффективно помогает ему овладеть программным материалом благодаря расстановке необходимых акцентов и удержанию внимания интонационными модуляциями голоса, а также подключением аудио-визуального механизма восприятия информации.

Самостоятельная работа преследует следующие цели:

- закрепление и совершенствование теоретических знаний, полученных на лекционных занятиях;
- формирование навыков подготовки текстовой составляющей информации учебного и научного назначения для размещения в различных информационных системах;
- совершенствование навыков поиска научных публикаций и образовательных ресурсов, размещенных в сети Интернет;
- самоконтроль освоения программного материала.

Обучающемуся необходимо помнить, что результаты самостоятельной работы контролируются преподавателем во время проведения мероприятий текущего контроля и учитываются при промежуточной аттестации.

Обучающимся с ОВЗ и инвалидов предоставляется возможность выбора форм проведения мероприятий текущего контроля, альтернативных формам, предусмотренным рабочей программой дисциплины. Предусматривается возможность увеличения в пределах 1 академического часа времени, отводимого на выполнение контрольных мероприятий.

Процедура оценивания результатов обучения инвалидов и лиц с ограниченными возможностями здоровья по дисциплине предусматривает предоставление информации в формах, адаптированных к ограничениям их здоровья и восприятия информации.

При проведении текущего контроля применяются оценочные средства, обеспечивающие передачу информации, от обучающегося к преподавателю, с учетом психофизиологических особенностей здоровья обучающихся.

## **7. Перечень учебно-методического обеспечения для самостоятельной работы обучающихся по дисциплине**

При самостоятельной работе обучающимся следует использовать:

- конспекты лекций;
- литературу из перечня основной и дополнительной учебной литературы, необходимой для освоения дисциплины (модуля);
- текст лекций на электронных носителях;
- ресурсы информационно-телекоммуникационной сети "Интернет", необходимые для освоения дисциплины;
- лицензионное и свободно распространяемое программное обеспечение из перечня информационных технологий, используемых при осуществлении образовательного процесса по дисциплине;
- методические указания для обучающихся по освоению дисциплины.

## 8. Перечень основной и дополнительной учебной литературы

### Основная:

1. Букунов С.В. Основы программирования на языке С++ [Электронный ресурс] : учебное пособие / С.В. Букунов. — Электрон. текстовые данные. — СПб. : Санкт-Петербургский государственный архитектурно-строительный университет, ЭБС АСВ, 2015. — 201 с. — 978-5-9227-0619-3. — Режим доступа: <http://www.iprbookshop.ru/63631.html> <http://www.iprbookshop.ru/63631.html>
2. Мейер, Б. Объектно-ориентированное программирование и программная инженерия / Б. Мейер. — 3-е изд. — Москва : Интернет-Университет Информационных Технологий (ИНТУИТ), Ай Пи Эр Медиа, 2019. — 285 с. — ISBN 978-5-4486-0513-0. — Текст : электронный // Электронно-библиотечная система IPR BOOKS : [сайт]. <http://www.iprbookshop.ru/79706>

### Дополнительная:

1. Тузовский, А. Ф. Объектно-ориентированное программирование : учебное пособие для прикладного бакалавриата / А. Ф. Тузовский. — Москва : Издательство Юрайт, 2019. — 206 с. — (Университеты России). — ISBN 978-5-534-00849-4. — Текст : электронный // ЭБС Юрайт [сайт]. <https://www.urait.ru/bcode/434045>
2. Подбельский В. В. Язык Си++: учебное пособие для вузов/ В. В. Подбельский.-Москва: Финансы и статистика, 2000, ISBN 5-279-02204-7.-560.-Библиогр.: с. 538-539
3. Казанский, А. А. Объектно-ориентированный анализ и программирование на Visual Basic 2013 : учебник для среднего профессионального образования / А. А. Казанский. — Москва : Издательство Юрайт, 2020. — 290 с. — (Профессиональное образование). — ISBN 978-5-534-03833-0. — Текст : электронный // ЭБС Юрайт [сайт]. <https://urait.ru/bcode/452453>



## **9. Перечень ресурсов сети Интернет, необходимых для освоения дисциплины**

<http://in.psu.ru/elis/> электронная библиотека ELiS

[https://skillbox.ru/media/code/oop\\_chast\\_1\\_chno\\_takoe\\_klassy\\_i\\_obekty/](https://skillbox.ru/media/code/oop_chast_1_chno_takoe_klassy_i_obekty/) ОБЪЕКТНО-ОРИЕНТИРОВАННОЕ ПРОГРАММИРОВАНИЕ.

## **10. Перечень информационных технологий, используемых при осуществлении образовательного процесса по дисциплине**

Образовательный процесс по дисциплине **Объектно-ориентированное программирование** предполагает использование следующего программного обеспечения и информационных справочных систем:

- 1) презентационные материалы (слайды по темам лекционных и практических занятий);
  - 2) доступ в режиме on-line в Электронную библиотечную систему (ЭБС);
  - 3) доступ в электронную информационно-образовательную среду университета;
  - 4) интернет-сервисы и электронные ресурсы (поисковые системы, электронная почта);
- Перечень необходимого лицензионного и (или) свободно распространяемого программного обеспечения

1. Проигрыватели виртуальных машин VirtualBox и VMWare Player (VMware Workstation).. Пакеты офисных программ (тестовые процессоры, табличные редакторы, программы для создания презентаций и др.).
3. С++ Builder или C#, MS Visual Studio с фреймворком .net минимум версии 4.0
4. Операционная система ALT Linux;
5. Офисный пакет приложений «LibreOffice».

При освоении материала и выполнения заданий по дисциплине рекомендуется использование материалов, размещенных в Личных кабинетах обучающихся ЕТИС ПГНИУ ([student.psu.ru](http://student.psu.ru)).

При организации дистанционной работы и проведении занятий в режиме онлайн могут использоваться:

система видеоконференцсвязи на основе платформы BigBlueButton (<https://bigbluebutton.org/>).

система LMS Moodle (<http://e-learn.psu.ru/>), которая поддерживает возможность использования текстовых материалов и презентаций, аудио- и видеоконтент, а так же тесты, проверяемые задания, задания для совместной работы.

система тестирования Indigo (<https://indigotech.ru/>).

## **11. Описание материально-технической базы, необходимой для осуществления образовательного процесса по дисциплине**

Для проведения лекционных занятий:

Аудитория, оснащенная презентационной техникой (проектор, экран, компьютер/ноутбук) с соответствующим программным обеспечением, меловой (и) или маркерной доской.

Для проведения лабораторных занятий – Компьютерный класс, оснащенный персональными ЭВМ и соответствующим программным обеспечением. Состав оборудования определен в Паспорте Компьютерного класса.

Аудитории для проведения текущего контроля;

Компьютерный класс, оснащенный персональными ЭВМ и соответствующим программным обеспечением. Состав оборудования определен в Паспорте компьютерного класса.

Аудитории для групповых (индивидуальных) консультаций;

Аудитория, оснащенная презентационной техникой (проектор, экран, компьютер/ноутбук) с соответствующим программным обеспечением, меловой (и) или маркерной доской.

Аудитория для самостоятельной работы:

Аудитория оснащенная компьютерной техникой с возможностью подключения к сети «Интернет», обеспеченная доступом в электронную информационно-образовательную среду университета.

Помещения научной библиотеки ПГНИУ для обеспечения самостоятельной работы обучающихся:

1. Научно-библиографический отдел, корп.1, ауд. 142. Оборудован 3 персональными компьютера с доступом к локальной и глобальной компьютерным сетям.

2. Читальный зал гуманитарной литературы, корп. 2, ауд. 418. Оборудован 7 персональными компьютерами с доступом к локальной и глобальной компьютерным сетям.

3. Читальный зал естественной литературы, корп.6, ауд. 107а. Оборудован 5 персональными компьютерами с доступом к локальной и глобальной компьютерным сетям.

4. Отдел иностранной литературы, корп.2 ауд. 207. Оборудован 1 персональным компьютером с доступом к локальной и глобальной компьютерным сетям.

5. Библиотека юридического факультета, корп.9, ауд. 4. Оборудована 11 персональными компьютерами с доступом к локальной и глобальной компьютерным сетям.

6. Читальный зал географического факультета, корп.8, ауд. 419. Оборудован 6 персональными компьютерами с доступом к локальной и глобальной компьютерным сетям.

Все компьютеры, установленные в помещениях научной библиотеки, оснащены следующим программным обеспечением:

Операционная система ALT Linux;

Офисный пакет Libreoffice.

Справочно-правовая система «КонсультантПлюс»

**Фонды оценочных средств для аттестации по дисциплине  
Объектно-ориентированное программирование**

**Планируемые результаты обучения по дисциплине для формирования компетенции.  
Индикаторы и критерии их оценивания**

**ПК.3**

**Способен выбирать и моделировать архитектурные решения для реализации программной системы**

Компетенция (индикатор)	Планируемые результаты обучения	Критерии оценивания результатов обучения
<p><b>ПК.3.1</b> Определяет архитектуры отдельных компонентов программной среды</p>	<p>Знать принципы объектно-ориентированного программирования; уметь применять эти принципы при разработке информационных систем. Уметь писать программы с использованием объектно-ориентированного подхода, в том числе создания классов, объектов с применением основных принципов ООП и использованием конструкторов и деструкторов. Иметь навыки написания программ с использованием объектно-ориентированного подхода, в том числе создания классов, объектов с применением основных принципов ООП и использованием конструкторов и деструкторов.</p>	<p align="center"><b>Неудовлетворител</b></p> <p>Не знает основные понятия объектно-ориентированного программирования и изучаемого языка программирования. Демонстрирует отсутствие навыков написания программ с использованием объектно-ориентированного подхода, в том числе создания классов, объектов с применением основных принципов ООП и использованием конструкторов и деструкторов</p> <p align="center"><b>Удовлетворительн</b></p> <p>Общие, но не структурированные знания основных понятий объектно-ориентированного программирования и изучаемого языка программирования. Демонстрирует частично сформированное умение написания программ с использованием объектно-ориентированного подхода, в том числе создания классов, объектов с применением основных принципов ООП и использованием конструкторов и деструкторов.</p> <p align="center"><b>Хорошо</b></p> <p>Сформированные, но содержащие отдельные пробелы знания основных понятий объектно-ориентированного программирования и изучаемого языка программирования. В целом успешные, но содержащие отдельные пробелы умения написания программ с использованием объектно-ориентированного подхода, в том числе создания классов, объектов с применением основных принципов ООП и использованием конструкторов и деструкторов</p>

Компетенция (индикатор)	Планируемые результаты обучения	Критерии оценивания результатов обучения
		<p style="text-align: center;"><b>Отлично</b></p> <p>Сформированные систематические знания основных понятий объектно-ориентированного программирования и изучаемого языка программирования. Успешное и систематическое применение навыков написания программ с использованием объектно-ориентированного подхода, в том числе создания классов, объектов с применением основных принципов ООП и использованием конструкторов и деструкторов.</p>
<p><b>ПК.3.2</b> Разрабатывает архитектуру программного обеспечения программной системы</p>	<p>знать принципы объектно-ориентированного программирования; уметь писать объектно-ориентированные программы на языке C++ в оболочке Microsoft Visual Studio; владеть навыками объектно-ориентированного программирования.</p>	<p style="text-align: center;"><b>Неудовлетворител</b></p> <p>не знает принципы объектно-ориентированного программирования; не умеет писать объектно-ориентированные программы на языке C++ в оболочке Microsoft Visual Studio; не владеет навыками объектно-ориентированного программирования.</p> <p style="text-align: center;"><b>Удовлетворительн</b></p> <p>частично знает принципы объектно-ориентированного программирования; умеет писать объектно-ориентированные программы на языке C++ в оболочке Microsoft Visual Studio; владеет навыками объектно-ориентированного программирования.</p> <p style="text-align: center;"><b>Хорошо</b></p> <p>по большей части знает принципы объектно-ориентированного программирования; умеет писать объектно-ориентированные программы на языке C++ в оболочке Microsoft Visual Studio; владеет навыками объектно-ориентированного программирования.</p> <p style="text-align: center;"><b>Отлично</b></p> <p>в полной мере знает принципы объектно-ориентированного программирования; умеет писать объектно-ориентированные программы на языке C++ в оболочке Microsoft Visual Studio; владеет на высоком уровне навыками объектно-ориентированного программирования.</p>

## Оценочные средства текущего контроля и промежуточной аттестации

Схема доставки : СУОС

**Вид мероприятия промежуточной аттестации :** Зачет

**Способ проведения мероприятия промежуточной аттестации :** Оценка по дисциплине в рамках промежуточной аттестации определяется на основе баллов, набранных обучающимся на контрольных мероприятиях, проводимых в течение учебного периода.

**Максимальное количество баллов :** 100

### Конвертация баллов в отметки

«отлично» - от 81 до 100

«хорошо» - от 61 до 80

«удовлетворительно» - от 48 до 60

«неудовлетворительно» / «незачтено» менее 48 балла

Компетенция (индикатор)	Мероприятие текущего контроля	Контролируемые элементы результатов обучения
<b>ПК.3.2</b> Разрабатывает архитектуру программного обеспечения программной системы <b>ПК.3.1</b> Определяет архитектуры отдельных компонентов программной среды	Введение в классы <b>Защищаемое контрольное мероприятие</b>	Знать принципы объектно-ориентированного программирования, синтаксис объявления классов, разделение членов класса на закрытые и открытые, принципы объявления и использования конструкторов и деструкторов. Уметь применять эти знания при написании программ с объявлением классов, создавать объекты и организовывать их взаимодействие
<b>ПК.3.2</b> Разрабатывает архитектуру программного обеспечения программной системы <b>ПК.3.1</b> Определяет архитектуры отдельных компонентов программной среды	Наследование и полиморфизм <b>Защищаемое контрольное мероприятие</b>	Знать принципы наследования, синтаксис и особенности наследования, конструкторы и деструкторы при наследовании, понятие полиморфизма, виртуальных функций и абстрактных базовых классов. Уметь создавать иерархии классов, виртуальные функции и использовать знания при написании программ с классами
<b>ПК.3.2</b> Разрабатывает архитектуру программного обеспечения программной системы <b>ПК.3.1</b> Определяет архитектуры отдельных компонентов программной среды	Шаблоны <b>Защищаемое контрольное мероприятие</b>	Знать принципы создания шаблонов классов и уметь при менять их при написании объектно-ориентированных программ.

Компетенция (индикатор)	Мероприятие текущего контроля	Контролируемые элементы результатов обучения
<b>ПК.3.2</b> Разрабатывает архитектуру программного обеспечения программной системы <b>ПК.3.1</b> Определяет архитектуры отдельных компонентов программной среды	Диаграммы UML <b>Итоговое контрольное мероприятие</b>	знать принципы объектно-ориентированного программирования; уметь писать объектно-ориентированные программы на языке C++ в оболочке Bourland C++ Builder; владеть навыками объектно-ориентированного программирования

### Спецификация мероприятий текущего контроля

#### Введение в классы

Продолжительность проведения мероприятия промежуточной аттестации: **1 часа**

Условия проведения мероприятия: **в часы самостоятельной работы**

Максимальный балл, выставляемый за мероприятие промежуточной аттестации: **20**

Проходной балл: **10**

Показатели оценивания	Баллы
На основании полученного у преподавателя индивидуального задания разработаны классы и созданы программные объекты на основе данных классов. Предоставлен текст разработанной программы, отлаженная и корректно работающая программа.	20
На основании полученного у преподавателя индивидуального задания разработаны классы и созданы программные объекты на основе данных классов. Предоставлен текст разработанной программы, отлаженная и корректно работающая программа. В программе есть недочёты, но студент знает, как их исправить.	9
На основании полученного у преподавателя индивидуального задания не разработаны классы и не созданы программные объекты на основе данных классов. Не предоставлен текст разработанной программы, отлаженная и корректно работающая программа.	0

#### Наследование и полиморфизм

Продолжительность проведения мероприятия промежуточной аттестации: **1 часа**

Условия проведения мероприятия: **в часы самостоятельной работы**

Максимальный балл, выставляемый за мероприятие промежуточной аттестации: **30**

Проходной балл: **13**

Показатели оценивания	Баллы
Студент демонстрирует умение разрабатывать иерархии классов и программ, использующих её. В программе, выполненной на основе полученного от преподавателя индивидуального задания, реализованы следующие элементы: базовый и производный классы, спецификаторы доступа, ключевое слово protected, множественное наследование, конструкторы и деструкторы при наследовании, виртуальные базовые классы, динамический полиморфизм, доступ к объектам производных классов через указатель на базовый класс, виртуальные функции, полиморфные классы, чистые виртуальные функции, абстрактные базовые классы, виртуальные деструкторы. Представлен текст разработанной	30

программы, отлаженная и корректно работающая программа.	
Студент демонстрирует умение разрабатывать иерархии классов и программ, использующих её. В программе, выполненной на основе полученного от преподавателя индивидуального задания, реализованы следующие элементы, за исключением 1-2: базовый и производный классы, спецификаторы доступа, ключевое слово protected, множественное наследование, конструкторы и деструкторы при наследовании, виртуальные базовые классы, динамический полиморфизм, доступ к объектам производных классов через указатель на базовый класс, виртуальные функции, полиморфные классы, чистые виртуальные функции, абстрактные базовые классы, виртуальные деструкторы. Представлен текст разработанной программы, отлаженная и корректно работающая программа.	19
Студент демонстрирует умение разрабатывать иерархии классов и программ, использующих её. В программе, выполненной на основе полученного от преподавателя индивидуального задания, реализованы некоторые из следующих элементов: базовый и производный классы, спецификаторы доступа, ключевое слово protected, множественное наследование, конструкторы и деструкторы при наследовании, виртуальные базовые классы, динамический полиморфизм, доступ к объектам производных классов через указатель на базовый класс, виртуальные функции, полиморфные классы, чистые виртуальные функции, абстрактные базовые классы, виртуальные деструкторы. Представлен текст разработанной программы, программа работает с ошибками	9
Студент не демонстрирует умение разрабатывать иерархии классов и программ, использующих её. Не представлен текст разработанной программы.	0

## Шаблоны

Продолжительность проведения мероприятия промежуточной аттестации: **1 часа**

Условия проведения мероприятия: **в часы аудиторной работы**

Максимальный балл, выставляемый за мероприятие промежуточной аттестации: **20**

Проходной балл: **10**

Показатели оценивания	Баллы
Студент демонстрирует умение разрабатывать шаблоны классов и использовать шаблоны при разработке программного обеспечения на примере разработанных ранее классов. Представлен текст разработанной программы, отлаженная и корректно работающая программа.	20
Студент частично демонстрирует умение разрабатывать шаблоны классов и использовать шаблоны при разработке программного обеспечения на примере разработанных ранее классов, но разработанное программное имеет ряд недочётов. Представлен текст разработанной программы, отлаженная и корректно работающая программа.	16
Студент частично демонстрирует умение разрабатывать шаблоны классов и использовать шаблоны при разработке программного обеспечения на примере разработанных ранее классов. Представлен текст разработанной программы, программа работает с ошибками.	9
Студент не демонстрирует умение разрабатывать шаблоны классов и использовать шаблоны при разработке программного обеспечения на примере разработанных ранее классов. Не представлен текст разработанной программы.	0

## Диаграммы UML

Продолжительность проведения мероприятия промежуточной аттестации: **1 часа**

Условия проведения мероприятия: **в часы аудиторной работы**

Максимальный балл, выставляемый за мероприятие промежуточной аттестации: **30**

Проходной балл: **15**

<b>Показатели оценивания</b>	<b>Баллы</b>
Студент даёт полный развёрнутый ответ на два вопроса итогового контроля.	30
Студент даёт ответ на два вопроса итогового контроля.	24
Студент даёт частичный ответ на два вопроса итогового контроля.	13
Студент не может ответить на два вопроса итогового контроля.	0