

**МИНОБРНАУКИ РОССИИ**

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

**Кафедра математического обеспечения вычислительных систем**

Авторы-составители: **Городилов Алексей Юрьевич**

Рабочая программа дисциплины

**СОВРЕМЕННЫЕ ЯЗЫКИ И ТЕХНОЛОГИИ ПРОГРАММИРОВАНИЯ**

Код УМК 90766

Утверждено  
Протокол №9  
от «24» мая 2019 г.

Пермь, 2019

## **1. Наименование дисциплины**

Современные языки и технологии программирования

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

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

Направление: **01.03.02** Прикладная математика и информатика  
направленность Программа широкого профиля

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

В результате освоения дисциплины **Современные языки и технологии программирования** у обучающегося должны быть сформированы следующие компетенции:

**01.03.02** Прикладная математика и информатика (направленность : Программа широкого профиля)

**ОПК.2** Способен решать задачи профессиональной деятельности с использованием существующих информационно-коммуникационных технологий и с учетом основных требований информационной безопасности

#### **Индикаторы**

**ОПК.2.1** Применяет знания основных положений и концепций в области программирования, архитектуру языков программирования, основную терминологию и базовые алгоритмы, основные требования информационной безопасности

**ОПК.2.2** Анализирует типовые языки программирования, составляет программы

**ОПК.2.3** Применяет на практике опыт решения задач с использованием базовых алгоритмов, анализа типов коммуникаций и интеграции различных типов программного обеспечения

**ОПК.3** Способен использовать и адаптировать существующие математические методы и системы программирования для разработки и реализации алгоритмов решения прикладных задач

#### **Индикаторы**

**ОПК.3.2** Разрабатывает и реализует алгоритм решения прикладной задачи

**ОПК.3.3** Демонстрирует практический опыт решения прикладных задач с использованием систем программирования и специализированного программного обеспечения

#### 4. Объем и содержание дисциплины

<b>Направления подготовки</b>	01.03.02 Прикладная математика и информатика (направленность: Программа широкого профиля)
<b>форма обучения</b>	очная
<b>№№ триместров, выделенных для изучения дисциплины</b>	5
<b>Объем дисциплины (з.е.)</b>	6
<b>Объем дисциплины (ак.час.)</b>	216
<b>Контактная работа с преподавателем (ак.час.), в том числе:</b>	84
<b>Проведение лекционных занятий</b>	28
<b>Проведение практических занятий, семинаров</b>	14
<b>Проведение лабораторных работ, занятий по иностранному языку</b>	42
<b>Самостоятельная работа (ак.час.)</b>	132
<b>Формы текущего контроля</b>	Входное тестирование (1) Защищаемое контрольное мероприятие (3) Итоговое контрольное мероприятие (1) Письменное контрольное мероприятие (2)
<b>Формы промежуточной аттестации</b>	Экзамен (5 триместр)

## 5. Аннотированное описание содержания разделов и тем дисциплины

### Современные языки и технологии программирования

#### Входной контроль

Входной контроль проверяет знание основ языков программирования C#, F# и умение составлять программы на них, полученные при изучении таких предметов, как "Алгоритмизация и программирование" и "Языки программирования".

#### Декларативное программирование. Языки семейства Lisp, Haskell

Особенности декларативного программирования. Преимущества и недостатки. Свойство контекстной независимости.

Язык программирования Lisp. Области применения. Атомы. Списки. Представление знаний с использованием списков.

Лямбда-исчисление. Функции в языке Lisp. Базовые функции. Примеры использования базовых функций. Функция DEFUN. Передача параметров и область их действия.

Аппликативный стиль программирования. Функция EVAL.

Использование рекурсии для организации повторяющихся вычислений. Простая рекурсия.

Накапливающие параметры. Параллельное ветвление рекурсии. Программирование вложенных циклов. Функции более высокого порядка.

#### Императивное программирование. C#, Java, Scala

Императивное программирование. Основные отличия от декларативного стиля.

Язык программирования Java. Особенности JAVA. Идентификаторы. Примитивные типы. Лексемы.

Операции, управляющие конструкции. Массивы. Классы и их структура. Поля, методы. Наследование.

Статические элементы. Структура java- программы.

Автоматическое управление памятью, сборка мусора.

Абстрактные классы. Интерфейсы и их множественное наследование. Полиморфизм. Модификаторы доступа. Инкапсуляция.

Аннотации как способ введения метаданных.

Рефлексия (Reflection).

Обобщенное программирование. Generic-типы в Java. Стандартные коллекции.

Библиотека тестирования JUnit

Ввод-вывод. Сериализация.

Мультипарадигмальный язык программирования Scala.

#### Интеграция программных модулей на различных языках программирования

Интеграция приложения на платформе .NET с офисными пакетами, представляющими API в виде подключаемых библиотек. Взаимодействие программы на языке C# с текстовым редактором и электронными таблицами. Особенности, возникающие с использованием сторонних приложений. Неуправляемые ресурсы.

Создание отдельных программных модулей на языках C++, C#, F# и их интеграция как приложений платформы .NET. Интеграция технологий .NET и Java. Поддержка стандартов.

#### Некоторые современные технологии программирования с использованием языка C++

Язык программирования C++. Основные особенности и отличия от языков C и C#.

Особенности работы с памятью. Статические и динамические объекты.

Множественное наследование.

Перегрузка операций.

Автоматное программирование. Сравнение программ в разных стилях на языке C++.

### **Сравнение различных парадигм и языков программирования**

Преимущества и недостатки различных парадигм программирования (императивная и декларативная; процедурная, объектно-ориентированная, функциональная; аппликативное, автоматное, символьное программирование). Основные особенности современных языков программирования. Разработка программ для решения одной задачи на различных языках программирования. Сравнение программ по трудозатратам на разработку, тестирование и отладку, по объему кода, по скорости работы завершеного приложения, оценка сложности масштабирования и модификации. Обоснование выбора языка и среды разработки для поставленной прикладной задачи.

### **Современные языки программирования для мобильных платформ**

Обзор современных языков программирования для мобильных платформ: Objective-C, Java, C, C++, C#

### **Итоговая контрольная работа**

Итоговая контрольная работа состоит из двух частей. Первая часть проводится в форме письменной контрольной работы во время заключительного лекционного занятия и включает в себя ряд задач по изученным языкам и технологиям программирования. Контрольная работа позволяет закрепить и обобщить полученные знания, проверить уровень освоения компетенции.

Вторая часть - это письменный ответ на теоретический вопрос. Вопрос предполагает самостоятельную работу студента по поиску нужной информации о новейших технологиях разработки программ, современных стандартах языков программирования.

## **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. Рогозин, О. В. Функциональное и рекурсивно-логическое программирование : учебное пособие / О. В. Рогозин. — Москва : Евразийский открытый институт, 2009. — 139 с. — ISBN 978-5-374-00182-2. — Текст : электронный // Электронно-библиотечная система IPR BOOKS : [сайт]. <http://www.iprbookshop.ru/11119>
2. Кауфман, В. Ш. Языки программирования. Концепции и принципы / В. Ш. Кауфман. — 2-е изд. — Саратов : Профобразование, 2019. — 464 с. — ISBN 978-5-4488-0137-2. — Текст : электронный // Электронно-библиотечная система IPR BOOKS : [сайт]. <http://www.iprbookshop.ru/88014.html>

### Дополнительная:

1. Подбельский В. В. Язык Си++: учеб. пособие для вузов/В. В. Подбельский.-М.:Финансы и статистика,2008, ISBN 978-5-279-02204-5.-560.-Библиогр.: с. 538-540
2. Буч Гради Объектно-ориентированный анализ и проектирование с примерами приложений на С++:Пер.с англ./Под ред.И.Романовского,Ф.Андреева.-М.,СПб.:Бином,Невский диалект,2001, ISBN 5-7989-0067-3.-560.
3. Залогова Л. А. Языки программирования. Основы объектно-ориентированного программирования на базе языка С#: учебное пособие для вузов/Л. А. Залогова.-Пермь,2017, ISBN 978-5-7944-2879-7.-169.
4. Вязовик Н. А.,Олифер Н. А. Программирование на Java:Курс лекций для студентов вузов, обучающихся по спец. 351400 "Прикл. информатика"/Интернет-Ун-т Информ. Технологий.- Москва:Интернет-Университет информационных технологий,2003, ISBN 5-9556-0006-X.-592.
5. Семенова Е. Т. Учебное пособие по курсу "Языки программирования". Язык программирования LISP 1.5/Е. Т. Семенова ; ред. Д. А. Поспелов.-М.:МЭИ,1977.-86.-Библиогр.: с. 82

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

<http://progopedia.ru/> Энциклопедия языков программирования

<http://www.helloworld.ru/> Документация и книги по программированию

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

Образовательный процесс по дисциплине **Современные языки и технологии программирования** предполагает использование следующего программного обеспечения и информационных справочных систем:

- доступ в режиме on-line в Электронную библиотечную систему (ЭБС);
- доступ в электронную информационно-образовательную среду университета.

Необходимое лицензионное и (или) свободно распространяемое программное обеспечение:

- Microsoft Office Standard
- Microsoft Visual Studio
- Среда разработки для языка Haskell (Haskell Stack)
- транслятор экрана VNC-viewer

При освоении материала и выполнении 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 лекционных занятий требуется аудитория, оснащенная презентационной техникой (проектор, экран, компьютер/ноутбук) с соответствующим программным обеспечением, меловой (и) или маркерной доской.

Для проведения практических занятий - аудитория, оснащенная презентационной техникой (проектор, экран, компьютер/ноутбук) с соответствующим программным обеспечением, меловой (и) или маркерной доской.

Для проведения лабораторных занятий - меловая и (или) маркерная доска, компьютерный класс (аппаратное и программное обеспечение определено в Паспортах компьютерных классов)

Для групповых (индивидуальных) консультаций - аудитория, оснащенная меловой (и) или маркерной доской.

Для проведения текущего контроля - аудитория, оснащенная меловой (и) или маркерной доской.

Самостоятельная работа студентов: аудитория, оснащенная компьютерной техникой с возможностью подключения к сети «Интернет», с обеспеченным доступом в электронную информационно-образовательную среду университета, помещения Научной библиотеки ПГНИУ.

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  
Современные языки и технологии программирования**

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

**ОПК.2**

**Способен решать задачи профессиональной деятельности с использованием существующих информационно-коммуникационных технологий и с учетом основных требований информационной безопасности**

<b>Компетенция (индикатор)</b>	<b>Планируемые результаты обучения</b>	<b>Критерии оценивания результатов обучения</b>
<p><b>ОПК.2.1</b> Применяет знания основных положений и концепций в области программирования, архитектуру языков программирования, основную терминологию и базовые алгоритмы, основные требования информационной безопасности</p>	<p>Знать принципиальные отличия, особенности, преимущества и недостатки различных парадигм программирования, современные технологии программирования, области применения языков Lisp, Haskell, Java, C#, C++; уметь составлять программы в декларативном и императивном стилях, применять аппликативный, автоматный стили программирования, использовать рефлексю, обобщенное программирование в объектно-ориентированных языках, интегрировать в одном приложении программные модули, реализованные на различных языках.</p>	<p align="center"><b>Неудовлетворител</b></p> <p>Не знает основные особенности, преимущества и недостатки различных парадигм программирования, современные технологии программирования, не имеет представления об аппликативном, автоматном стилях программирования; не умеет составлять программы в декларативном и императивном стилях, использовать рефлексю, обобщенное программирование в объектно-ориентированных языках, интегрировать в одном приложении программные модули, реализованные на различных языках.</p> <p align="center"><b>Удовлетворительн</b></p> <p>Общие, но не структурированные знания основных особенностей, преимуществ и недостатков различных парадигм программирования, современных технологий программирования, аппликативного, автоматного стилей программирования; умеет составлять программы в декларативном и императивном стилях по известным алгоритмам, использовать рефлексю, обобщенное программирование в объектно-ориентированных языках, интегрировать в одном приложении программные модули, реализованные на различных языках.</p> <p align="center"><b>Хорошо</b></p> <p>Сформированные знания основных отличий, особенностей, преимуществ и недостатков различных парадигм программирования, современных технологий программирования, областей применения различных языков;</p>

Компетенция (индикатор)	Планируемые результаты обучения	Критерии оценивания результатов обучения
		<p style="text-align: center;"><b>Хорошо</b></p> <p>умеет составлять программы в декларативном и императивном стилях, применять аппликативный, автоматный стили программирования, использовать рефлексия, обобщенное программирование в объектно-ориентированных языках, интегрировать в одном приложении программные модули, реализованные на различных языках, сравнивать программы в разных стилях.</p> <p style="text-align: center;"><b>Отлично</b></p> <p>Сформированные систематические знания отличий, особенностей, преимуществ и недостатков различных парадигм программирования, современных технологий программирования, областей применения языков Lisp, Haskell, Java, C#, C++; в совершенстве умеет составлять программы в декларативном и императивном стилях, применять аппликативный, автоматный стили программирования, использовать рефлексия, обобщенное программирование в объектно-ориентированных языках, интегрировать в одном приложении программные модули, реализованные на различных языках.</p>
<p><b>ОПК.2.2</b> Анализирует типовые языки программирования, составляет программы</p>	<p>Уметь составлять программы в декларативном и императивном стилях, применять аппликативный, автоматный стили программирования, использовать рефлексия, обобщенное программирование в объектно-ориентированных языках, интегрировать в одном приложении программные модули, реализованные на различных языках; владеть навыками составления программ на языках программирования Lisp, Haskell, Java, C#, C++, сравнения программ в разных стилях.</p>	<p style="text-align: center;"><b>Неудовлетворител</b></p> <p>Не умеет составлять программы в декларативном и императивном стилях, использовать рефлексия, обобщенное программирование в объектно-ориентированных языках, интегрировать в одном приложении программные модули, реализованные на различных языках; отсутствие навыков составления программ на языках программирования Lisp, Haskell, Java, C#, C++.</p> <p style="text-align: center;"><b>Удовлетворительн</b></p> <p>Умеет составлять программы в декларативном и императивном стилях по известным алгоритмам, использовать рефлексия, обобщенное программирование в объектно-ориентированных языках, интегрировать в одном приложении</p>

Компетенция (индикатор)	Планируемые результаты обучения	Критерии оценивания результатов обучения
		<p style="text-align: center;"><b>Удовлетворительн</b></p> <p>программные модули, реализованные на различных языках; фрагментарное применение навыков составления программ на языках программирования Lisp, Haskell, Java, C#, C++.</p> <p style="text-align: center;"><b>Хорошо</b></p> <p>Умеет составлять программы в декларативном и императивном стилях, применять аппликативный, автоматный стили программирования, использовать рефлексия, обобщенное программирование в объектно-ориентированных языках, интегрировать в одном приложении программные модули, реализованные на различных языках, сравнивать программы в разных стилях; в целом успешно применяет навыки составления программ на языках программирования Lisp, Haskell, Java, C#, C++.</p> <p style="text-align: center;"><b>Отлично</b></p> <p>В совершенстве умеет составлять программы в декларативном и императивном стилях, применять аппликативный, автоматный стили программирования, использовать рефлексия, обобщенное программирование в объектно-ориентированных языках, интегрировать в одном приложении программные модули, реализованные на различных языках; успешно применяет навыки составления программ на языках программирования Lisp, Haskell, Java, C#, C++, сравнения программ в разных стилях.</p>
<p><b>ОПК.2.3</b> Применяет на практике опыт решения задач с использованием базовых алгоритмов, анализа типов коммуникаций и интеграции различных типов программного</p>	<p>Уметь составлять программы в декларативном и императивном стилях, интегрировать в одном приложении программные модули, реализованные на различных языках; владеть навыками составления программ на языках программирования Lisp,</p>	<p style="text-align: center;"><b>Неудовлетворител</b></p> <p>Не умеет составлять программы в декларативном и императивном стилях, интегрировать в одном приложении программные модули, реализованные на различных языках; отсутствие навыков составления программ на языках программирования Lisp, Haskell, Java, C#, C++.</p>

Компетенция (индикатор)	Планируемые результаты обучения	Критерии оценивания результатов обучения
обеспечения	Haskell, Java, C#, C++, сравнения программ в разных стилях.	<p style="text-align: center;"><b>Удовлетворительн</b></p> <p>Умеет составлять программы в декларативном и императивном стилях по известным алгоритмам, интегрировать в одном приложении программные модули, реализованные на различных языках; фрагментарное применение навыков составления программ на языках программирования Lisp, Haskell, Java, C#, C++.</p> <p style="text-align: center;"><b>Хорошо</b></p> <p>Умеет составлять программы в декларативном и императивном стилях, интегрировать в одном приложении программные модули, реализованные на различных языках, сравнивать программы в разных стилях; в целом успешно применяет навыки составления программ на языках программирования Lisp, Haskell, Java, C#, C++.</p> <p style="text-align: center;"><b>Отлично</b></p> <p>В совершенстве умеет составлять программы в декларативном и императивном стилях, интегрировать в одном приложении программные модули, реализованные на различных языках; успешно применяет навыки составления программ на языках программирования Lisp, Haskell, Java, C#, C++, сравнения программ в разных стилях.</p>

### ОПК.3

**Способен использовать и адаптировать существующие математические методы и системы программирования для разработки и реализации алгоритмов решения прикладных задач**

Компетенция (индикатор)	Планируемые результаты обучения	Критерии оценивания результатов обучения
<b>ОПК.3.2</b> Разрабатывает и реализует алгоритм решения прикладной задачи	Знать принципиальные отличия, особенности, преимущества и недостатки различных парадигм программирования, современные технологии программирования; уметь разрабатывать и	<p style="text-align: center;"><b>Неудовлетворител</b></p> <p>Не знает основные особенности, преимущества и недостатки различных парадигм программирования, современные технологии программирования, не имеет представления об аппликативном, автоматном стилях программирования; не умеет реализовывать алгоритмы решения</p>

Компетенция (индикатор)	Планируемые результаты обучения	Критерии оценивания результатов обучения
	<p>реализовывать алгоритмы решения прикладных задач в виде программ в декларативном и императивном стилях, применять аппликативный, автоматный стили программирования, использовать рефлексия, обобщенное программирование в объектно-ориентированных языках.</p>	<p><b>Неудовлетворител</b> прикладных задач в виде программ в декларативном и императивном стилях, использовать рефлексия, обобщенное программирование в объектно-ориентированных языках.</p> <p><b>Удовлетворительн</b> Общие, но не структурированные знания основных особенностей, преимуществ и недостатков различных парадигм программирования, современных технологий программирования, аппликативного, автоматного стили программирования; умеет реализовывать известные алгоритмы решения прикладных задач в виде программ в декларативном и императивном стилях, использовать рефлексия, обобщенное программирование в объектно-ориентированных языках.</p> <p><b>Хорошо</b> Сформированные знания основных отличий, особенностей, преимуществ и недостатков различных парадигм программирования, современных технологий программирования; умеет разрабатывать и реализовывать алгоритмы решения прикладных задач в виде программ в декларативном и императивном стилях, применять аппликативный, автоматный стили программирования, использовать рефлексия, обобщенное программирование в объектно-ориентированных языках.</p> <p><b>Отлично</b> Сформированные систематические знания отличий, особенностей, преимуществ и недостатков различных парадигм программирования, современных технологий программирования; в совершенстве умеет разрабатывать и реализовывать алгоритмы решения прикладных задач в виде программ в декларативном и императивном стилях, применять аппликативный, автоматный стили программирования, использовать</p>

Компетенция (индикатор)	Планируемые результаты обучения	Критерии оценивания результатов обучения
		<p align="center"><b>Отлично</b></p> <p>рефлексию, обобщенное программирование в объектно-ориентированных языках.</p>
<p><b>ОПК.3.3</b> Демонстрирует практический опыт решения прикладных задач с использованием систем программирования и специализированного программного обеспечения</p>	<p>Знать области применения языков Lisp, Haskell, Java, C#, C++; уметь составлять программы в декларативном и императивном стилях, интегрировать в одном приложении программные модули, реализованные на различных языках, сравнивать программы в разных стилях; владеть навыками составления программ на языках программирования Lisp, Haskell, Java, C#, C++, решающих прикладные задачи, используя для этого соответствующие системы программирования.</p>	<p align="center"><b>Неудовлетворител</b></p> <p>Не знает основных областей применения различных языков; не умеет составлять программы в декларативном и императивном стилях, интегрировать в одном приложении программные модули, реализованные на различных языках; отсутствие навыков составления программ на языках программирования Lisp, Haskell, Java, C#, C++.</p> <p align="center"><b>Удовлетворительн</b></p> <p>Общие, но не структурированные знания основных областей применения различных языков; умеет составлять программы в декларативном и императивном стилях по известным алгоритмам, интегрировать в одном приложении программные модули, реализованные на различных языках; фрагментарное применение навыков составления программ на языках программирования Lisp, Haskell, Java, C#, C++ с использованием соответствующих систем программирования.</p> <p align="center"><b>Хорошо</b></p> <p>Сформированные знания основных областей применения различных языков; умеет составлять программы в декларативном и императивном стилях, интегрировать в одном приложении программные модули, реализованные на различных языках, сравнивать программы в разных стилях; в целом успешно применяет навыки составления программ на языках программирования Lisp, Haskell, Java, C#, C++, решающих прикладные задачи, используя для этого соответствующие системы программирования.</p> <p align="center"><b>Отлично</b></p> <p>Сформированные систематические знания</p>

Компетенция (индикатор)	Планируемые результаты обучения	Критерии оценивания результатов обучения
		<p style="text-align: center;"><b>Отлично</b></p> <p>областей применения языков Lisp, Haskell, Java, C#, C++;</p> <p>в совершенстве умеет составлять программы в декларативном и императивном стилях, интегрировать в одном приложении программные модули, реализованные на различных языках, сравнивать программы в разных стилях;</p> <p>успешно применяет навыки составления программ на языках программирования Lisp, Haskell, Java, C#, C++, решающих прикладные задачи, используя для этого соответствующие системы программирования.</p>

## Оценочные средства текущего контроля и промежуточной аттестации

Схема доставки : Базовая

**Вид мероприятия промежуточной аттестации :** Экзамен

**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 44 до 60

«неудовлетворительно» / «незачтено» менее 44 балла

<b>Компетенция (индикатор)</b>	<b>Мероприятие текущего контроля</b>	<b>Контролируемые элементы результатов обучения</b>
<b>Входной контроль</b>	Входной контроль <b>Входное тестирование</b>	Знание языков программирования C#, F# и умение составлять программы на них.

Компетенция (индикатор)	Мероприятие текущего контроля	Контролируемые элементы результатов обучения
<p><b>ОПК.2.3</b> Применяет на практике опыт решения задач с использованием базовых алгоритмов, анализа типов коммуникаций и интеграции различных типов программного обеспечения</p> <p><b>ОПК.2.2</b> Анализирует типовые языки программирования, составляет программы</p> <p><b>ОПК.2.1</b> Применяет знания основных положений и концепций в области программирования, архитектуру языков программирования, основную терминологию и базовые алгоритмы, основные требования информационной безопасности</p> <p><b>ОПК.3.3</b> Демонстрирует практический опыт решения прикладных задач с использованием систем программирования и специализированного программного обеспечения</p> <p><b>ОПК.3.2</b> Разрабатывает и реализует алгоритм решения прикладной задачи</p>	<p>Декларативное программирование. Языки семейства Lisp, Haskell</p> <p><b>Защищаемое контрольное мероприятие</b></p>	<p>Знать области применения языков Lisp, Haskell; уметь составлять программы в декларативном стиле, применять аппликативный стиль программирования; владеть навыками составления программ на языках программирования Lisp, Haskell.</p>

Компетенция (индикатор)	Мероприятие текущего контроля	Контролируемые элементы результатов обучения
<p><b>ОПК.2.3</b> Применяет на практике опыт решения задач с использованием базовых алгоритмов, анализа типов коммуникаций и интеграции различных типов программного обеспечения</p> <p><b>ОПК.2.2</b> Анализирует типовые языки программирования, составляет программы</p> <p><b>ОПК.2.1</b> Применяет знания основных положений и концепций в области программирования, архитектуру языков программирования, основную терминологию и базовые алгоритмы, основные требования информационной безопасности</p> <p><b>ОПК.3.3</b> Демонстрирует практический опыт решения прикладных задач с использованием систем программирования и специализированного программного обеспечения</p> <p><b>ОПК.3.2</b> Разрабатывает и реализует алгоритм решения прикладной задачи</p>	<p>Императивное программирование. C#, Java, Scala</p> <p><b>Защищаемое контрольное мероприятие</b></p>	<p>Знать современные технологии программирования, области применения языков Java, C#; уметь составлять программы в императивном стиле, использовать рефлексию, обобщенное программирование в объектно-ориентированных языках; владеть навыками составления программ на языках программирования Java, C#.</p>

Компетенция (индикатор)	Мероприятие текущего контроля	Контролируемые элементы результатов обучения
<p><b>ОПК.2.1</b> Применяет знания основных положений и концепций в области программирования, архитектуру языков программирования, основную терминологию и базовые алгоритмы, основные требования информационной безопасности</p> <p><b>ОПК.2.2</b> Анализирует типовые языки программирования, составляет программы</p> <p><b>ОПК.2.3</b> Применяет на практике опыт решения задач с использованием базовых алгоритмов, анализа типов коммуникаций и интеграции различных типов программного обеспечения</p> <p><b>ОПК.3.3</b> Демонстрирует практический опыт решения прикладных задач с использованием систем программирования и специализированного программного обеспечения</p>	<p>Интеграция программных модулей на различных языках программирования</p> <p><b>Защищаемое контрольное мероприятие</b></p>	<p>Уметь интегрировать в одном приложении программные модули, реализованные на различных языках; владеть навыками составления программ, взаимодействующих со сторонними прикладными приложениями.</p>

Компетенция (индикатор)	Мероприятие текущего контроля	Контролируемые элементы результатов обучения
<p><b>ОПК.2.3</b> Применяет на практике опыт решения задач с использованием базовых алгоритмов, анализа типов коммуникаций и интеграции различных типов программного обеспечения</p> <p><b>ОПК.2.2</b> Анализирует типовые языки программирования, составляет программы</p> <p><b>ОПК.2.1</b> Применяет знания основных положений и концепций в области программирования, архитектуру языков программирования, основную терминологию и базовые алгоритмы, основные требования информационной безопасности</p> <p><b>ОПК.3.3</b> Демонстрирует практический опыт решения прикладных задач с использованием систем программирования и специализированного программного обеспечения</p> <p><b>ОПК.3.2</b> Разрабатывает и реализует алгоритм решения прикладной задачи</p>	<p>Некоторые современные технологии программирования с использованием языка C++</p> <p><b>Письменное контрольное мероприятие</b></p>	<p>Знать области применения языка C++; уметь составлять программы в императивном стиле, применять автоматный стиль программирования; владеть навыками составления программ на языке программирования C++.</p>

Компетенция (индикатор)	Мероприятие текущего контроля	Контролируемые элементы результатов обучения
<p><b>ОПК.2.1</b> Применяет знания основных положений и концепций в области программирования, архитектуру языков программирования, основную терминологию и базовые алгоритмы, основные требования информационной безопасности</p> <p><b>ОПК.2.2</b> Анализирует типовые языки программирования, составляет программы</p> <p><b>ОПК.2.3</b> Применяет на практике опыт решения задач с использованием базовых алгоритмов, анализа типов коммуникаций и интеграции различных типов программного обеспечения</p> <p><b>ОПК.3.3</b> Демонстрирует практический опыт решения прикладных задач с использованием систем программирования и специализированного программного обеспечения</p>	<p>Сравнение различных парадигм и языков программирования</p> <p><b>Письменное контрольное мероприятие</b></p>	<p>Уметь составлять программы в декларативном и императивном стилях; владеть навыками составления программ на языках программирования Lisp, Haskell, Java, C#, C++, сравнения программ в разных стилях.</p>

Компетенция (индикатор)	Мероприятие текущего контроля	Контролируемые элементы результатов обучения
<p><b>ОПК.2.1</b> Применяет знания основных положений и концепций в области программирования, архитектуру языков программирования, основную терминологию и базовые алгоритмы, основные требования информационной безопасности</p> <p><b>ОПК.3.3</b> Демонстрирует практический опыт решения прикладных задач с использованием систем программирования и специализированного программного обеспечения</p> <p><b>ОПК.3.2</b> Разрабатывает и реализует алгоритм решения прикладной задачи</p>	<p>Итоговая контрольная работа</p> <p><b>Итоговое контрольное мероприятие</b></p>	<p>Знать принципиальные отличия, особенности, преимущества и недостатки различных парадигм программирования, уметь осуществлять целенаправленный поиск информации о современных языках и технологиях программирования в сети Интернет и других источниках.</p>

### Спецификация мероприятий текущего контроля

#### Входной контроль

Продолжительность проведения мероприятия промежуточной аттестации: **2 часа**

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

Максимальный балл, выставляемый за мероприятие промежуточной аттестации: **0**

Проходной балл: **0**

Показатели оценивания	Баллы
Выполнено задание по разработке простого приложения на языке C#, состоящего из одного класса с заданными публичными методами и конструкторами.	3
Выполнено задание по разработке простой функции на языке F#, обрабатывающей поступающий на вход список пар значений.	3
Выполнено задание по разработке простого приложения на языке C#, при этом реализация эффективна, полностью соответствует заданному описанию, класс не содержит лишних элементов.	2
Выполнено задание по разработке простой функции на языке F#, при этом программа описана в функциональном стиле, грамотно оформлена, не содержит лишних функций.	2

#### Декларативное программирование. Языки семейства Lisp, Haskell

Продолжительность проведения мероприятия промежуточной аттестации: **2 часа**

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

Максимальный балл, выставяемый за мероприятие промежуточной аттестации: **12**

Проходной балл: **5**

Показатели оценивания	Баллы
Знает области применения языков Lisp, Haskell, умеет с их помощью решать прикладные задачи (5 задач по 1 баллу)	5
Умеет составлять программы в декларативном стиле: создавать и работать с собственными типами и классами типов	3
Владеет навыками составления программ на языках программирования Lisp, Haskell: знает базовые функции и умеет применять их для составления программ (2 задачи по 1 баллу).	2
Умеет применять аппликативный стиль программирования	2

### **Императивное программирование. C#, Java, Scala**

Продолжительность проведения мероприятия промежуточной аттестации: **2 часа**

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

Максимальный балл, выставяемый за мероприятие промежуточной аттестации: **16**

Проходной балл: **7**

Показатели оценивания	Баллы
Владеет навыками составления программ на языках программирования Java, C#: знает основные принципы ООП и умеет применять их для составления программ (правильно решена задача на составление программы, демонстрирующей основы объектно-ориентированного стиля)	5
Знает области применения языков Java, C#, умеет с их помощью решать прикладные задачи (правильно решена задача разработки программы в объектно-ориентированном стиле, решающей прикладную задачу)	4
Знает современные технологии программирования, умеет применять сериализацию для сохранения и загрузки объектов.	2
Умеет составлять программы в императивном стиле, использует возможности полиморфизма при составлении программ (реализованы дополнительные возможности в программе, демонстрирующей основы объектно-ориентированного программирования)	2
Умеет применять событийно-ориентированный подход к разработке программ в объектно-ориентированных языках (реализованы дополнительные возможности в программе, решающей прикладную задачу)	2
Умеет использовать рефлексии в объектно-ориентированных языках.	1

### **Интеграция программных модулей на различных языках программирования**

Продолжительность проведения мероприятия промежуточной аттестации: **2 часа**

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

Максимальный балл, выставяемый за мероприятие промежуточной аттестации: **15**

Проходной балл: **7**

Показатели оценивания	Баллы
Создано приложение, решающее поставленную задачу, в котором интегрированы программные модули, реализованные на различных языках.	5
Создано приложение на языке C#, взаимодействующее с офисными приложениями	5

(читающее данные из электронных таблиц и записывающее данные в текстовый редактор), решающее прикладную задачу	
Создано приложение, решающее поставленную задачу, в котором интегрированы программные модули, реализованные на различных языках. Обеспечена возможность подключения других модулей с аналогичной функциональностью. Выбор конкретных языков для реализации отдельных модулей обоснован.	3
Создано приложение на языке C#, взаимодействующее с офисными приложениями, обеспечена возможность настройки имён файлов, характеристик используемых шрифтов и т.п.	2

### **Некоторые современные технологии программирования с использованием языка C++**

Продолжительность проведения мероприятия промежуточной аттестации: **2 часа**

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

Максимальный балл, выставляемый за мероприятие промежуточной аттестации: **12**

Проходной балл: **5**

<b>Показатели оценивания</b>	<b>Баллы</b>
Владеет навыками составления программ на языке программирования C++: знает отличительные возможности языка C++ (прямая работа с памятью, особенности работы с объектами, множественное наследование) и умеет применять их для составления программ (правильно решена задача по составлению программы на языке C++)	5
Умеет применять автоматный стиль программирования (правильно решена задача по составлению программы, реализующей автоматный стиль программирования)	4
Умеет составлять программы в императивном (в том числе процедурном) стиле на языке C++ (реализованы дополнительные возможности в программе на языке C++)	2
Знает области применения языка C++, умеет с его помощью решать прикладные задачи	1

### **Сравнение различных парадигм и языков программирования**

Продолжительность проведения мероприятия промежуточной аттестации: **2 часа**

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

Максимальный балл, выставляемый за мероприятие промежуточной аттестации: **25**

Проходной балл: **11**

<b>Показатели оценивания</b>	<b>Баллы</b>
Создана программа, демонстрирующая возможности языков программирования для создания и модификации офисных приложений. Составлен отчет, содержащий краткое описание разработанной программ и использованных алгоритмов.	10
Создана программа на одном из объектно-ориентированных языков программирования, решающая поставленную задачу. Программа протестирована. Даны комментарии и пояснения по назначению отдельных объектов, классов, методов, по алгоритму работы.	6
Создана программа на одном из функциональных языков программирования, решающая поставленную задачу. Программа протестирована. Даны комментарии и пояснения по назначению отдельных имен и функций, по алгоритму работы.	6
Созданы программы на объектно-ориентированном и функциональном языках программирования, решающие поставленную задачу. Составлен отчет, содержащий	2

сравнение программ по объему полученного кода, быстрдействию, скорости работы (объективно – на основе замеров времени работы на различных входных данных), трудоемкости разработки, тестирования и отладки (приблизительно, субъективно).	
Созданы программы на объектно-ориентированном и функциональном языках программирования, решающие поставленную задачу. Составлен отчет, содержащий краткое описание использованных алгоритмов.	1

### **Итоговая контрольная работа**

Продолжительность проведения мероприятия промежуточной аттестации: **2 часа**

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

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

Проходной балл: **9**

<b>Показатели оценивания</b>	<b>Баллы</b>
Умеет осуществлять целенаправленный поиск информации о современных языках и технологиях программирования. Ответ на теоретический вопрос о современном языке или технологии программирования.	10
Знает принципиальные отличия, особенности, преимущества и недостатки функциональных языков программирования.	3
Знает принципиальные отличия, особенности, преимущества и недостатки языка программирования C++.	3
Знает принципиальные отличия, особенности, преимущества и недостатки объектно-ориентированных языков программирования.	3
Знает современные парадигмы программирования.	1