

**МИНОБРНАУКИ РОССИИ**

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

**Кафедра информационной безопасности и систем связи**

**Авторы-составители: Никитина Елена Юрьевна  
Айдаров Юрий Рафаэлевич  
Мустакимова Яна Романовна**

Рабочая программа дисциплины  
**МЕТОДЫ ПРОГРАММИРОВАНИЯ I**  
Код УМК 88889

Утверждено  
Протокол №1  
от «31» августа 2020 г.

Пермь, 2020

## **1. Наименование дисциплины**

Методы программирования I

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

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

Направление: **09.03.02** Информационные системы и технологии  
направленность Безопасность информационных систем

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

В результате освоения дисциплины **Методы программирования I** у обучающегося должны быть сформированы следующие компетенции:

**09.03.02** Информационные системы и технологии (направленность : Безопасность информационных систем)

**ПК.2** Способность выполнять работы по созданию (модификации) и сопровождению информационных систем, автоматизирующих задачи организационного управления и бизнес-процессы

#### **Индикаторы**

**ПК.2.2** Применяет навыки использования библиотек для разработки программных комплексов, в том числе при коллективной разработке

**ПК.3** Способность применять инструментальные средства разработки для создания защищенных программных средств и систем; работать с программными средствами прикладного, системного и специализированного назначения для обеспечения информационной безопасности

#### **Индикаторы**

**ПК.3.1** Использует современные языки, методы, технологии, системы и инструментальные средства программирования для разработки защищенных программных средств и систем прикладного и специализированного назначения

**ПК.3.2** Применяет опыт разработки приложений и программных прототипов решений прикладных задач на основе современных языков, методов и технологий программирования для защиты программ и данных

#### 4. Объем и содержание дисциплины

<b>Направления подготовки</b>	09.03.02 Информационные системы и технологии (направленность: Безопасность информационных систем)
<b>форма обучения</b>	очная
<b>№№ триместров, выделенных для изучения дисциплины</b>	2
<b>Объем дисциплины (з.е.)</b>	4
<b>Объем дисциплины (ак.час.)</b>	144
<b>Контактная работа с преподавателем (ак.час.), в том числе:</b>	56
<b>Проведение лекционных занятий</b>	28
<b>Проведение лабораторных работ, занятий по иностранному языку</b>	28
<b>Самостоятельная работа (ак.час.)</b>	88
<b>Формы текущего контроля</b>	Итоговое контрольное мероприятие (1) Письменное контрольное мероприятие (8)
<b>Формы промежуточной аттестации</b>	Экзамен (2 триместр)

## **5. Аннотированное описание содержания разделов и тем дисциплины**

### **Системы счисления**

Понятие системы счисления. Классификация систем счисления. Позиционные и непозиционные системы счисления. Характеристики позиционных систем счисления: базис, алфавит, основание. Формы записи чисел в системах счисления: свернутая, развернутая.

Алгоритмы перевода целой и дробной частей между 10-чной системой счисления и  $p$ -чной системой счисления.

Правила проведения арифметических операций в позиционных системах счисления.

Смешанные системы счисления. Алгоритмы перевода чисел между смешанными системами счисления.

### **Элементы теории алгоритмов**

#### **Понятие алгоритма. Свойства алгоритмов. Способы записи алгоритмов**

Понятие алгоритма.

Свойства алгоритма: массовость, дискретность, результативность, конечность, понятность, точность, детерминированность, эффективность.

Способы записи алгоритма: словесная, формульная, табличная, блок-схема, псевдокод, язык программирования. Требования к записи алгоритма в соответствующей форме. Стандарт и диалект языков программирования.

#### **Машина Тьюринга**

Понятие машины Тьюринга. Составные части машины Тьюринга.

Формы записи программы для машины Тьюринга: последовательная, табличная, графовая.

Правила выполнения программы для машины Тьюринга.

Дополнительные условия на содержание программы для машины Тьюринга.

Композиции машин Тьюринга: последовательное соединение, итерация.

#### **Алгоритм Маркова**

Понятие алгоритма Маркова. Составные части алгоритма Маркова.

Формулы подстановки: простая, завершающая. Схема алгоритма Маркова.

Правила выполнения алгоритма Маркова.

Композиции алгоритмов Маркова.

### **Методика решения задач на ЭВМ**

#### **Этапы решения задачи на ЭВМ. Этапы прохождения задания через ЭВМ**

Этапы решения задач любой сложности любой предметной области с помощью ЭВМ. Особенности этапов.

Этапы прохождения задания через ЭВМ. Современная реализация этапов прохождения в виде интегрированных систем программирования.

#### **Последовательное построение алгоритма**

Метод пошаговой детализации для последовательного построения алгоритма решения задачи. Варианты метода: "сверху-вниз" и "снизу-вверх". Применение псевдокода для последовательного построения алгоритма.

#### **Правила организации рекуррентных вычислений**

Понятие рекуррентных вычислений. Рекуррентная формула. Порядок рекуррентной формулы. Виды рекуррентных формул.

Технические особенности построения программ, содержащих рекуррентные вычисления.

## **Правила организации рекурсивных вычислений**

Понятие рекурсии. Виды рекурсии: прямая, косвенная.

Особенности выполнения программ, содержащих рекурсию. Рекурсивный стек. Прямой и обратный ход рекурсии.

Технические особенности построения программ, содержащих рекурсию.

Затраты ЭВМ на выполнение рекурсивных программ. Сравнение рекурсии и итерации.

## **Алгоритмы с возвратами**

Понятие алгоритма (задачи) с возвратами. Понятие эвристики. Бэктрекинг.

Технические особенности реализации программ, содержащих алгоритмы с возвратами.

## **Оценка сложности алгоритмов**

### **Оценка сложности по управляющему графу**

Понятие сложности алгоритма. Виды сложности: сложность по данным, сложность по времени выполнения.

Правила вычисления сложности по данным.

Построение управляющего графа алгоритма.

Правила вычисления сложности по времени выполнения для структур: следование, ветвление, повторение. Методы "пессимиста" и "среднего".

### **Оценка сложности рекурсивных алгоритмов**

Выделение составных частей рекурсивного алгоритма для вычисления сложности по времени выполнения. Построение рекуррентного уравнения функции сложности рекурсивного алгоритма.

Решение рекуррентного уравнения. Понятие  $O(n)$  и  $o(n)$ .

### **Классы сложности задач**

Понятие задачи. Общие и частные задачи. Разрешимые и неразрешимые задачи. Оптимизация задачи.

Особенности вычисления сложности задач.

Классы сложности: P, EXP, NP.

## **Сортировка и поиск**

### **Параметры оценки сортировок**

Понятие сортировки. Ключевая и информационная части сортируемого элемента.

Основные параметры сортировки: по времени выполнения, по объему памяти, по распределению элементов, по значению элементов.

Дополнительные параметры для определения алгоритма сортировки: размер данных, характеристики ключевой части сортируемого элемента, объем информационной части сортируемого элемента, программные связи, характеристики ЭВМ для реализации сортировки.

### **Классификация сортировок**

Общая классификация сортировок: внутренние и внешние сортировки.

Классификация внутренних сортировок: вставками, выбором, обменом, подсчетом. Общие алгоритмы классов.

### **Внутренние методы сортировки**

Общие алгоритмы классов внутренних сортировок.

Улучшенные алгоритмы сортировок: бинарные вставки, шейкерная, челночная, парным обменом, поразрядная, квадратичным выбором, Бэтчера, быстрая сортировка Хоара. Оценка сложности

внутренних сортировок.

### **Внешние методы сортировки**

Сортировки фон Неймана (трехленточная сортировка файлов), двухпутевое слияние.

Оценка сложности внешних сортировок. Особенности реализации.

### **Хеширование**

Понятие хеширования. Хэш-функция, требования к ее построению, особенности построения хэш-функций.

Понятие коллизии. Рехеширование. Виды рехеширования: линейное, случайное, квадратичное, метод цепочек.

### **Тестирование и отладка программ**

Понятие тестирования. Принципы тестирования.

Методы тестирования: инспекция исходного текста, сквозной просмотр, проверка за столом, "черный ящик", "белый ящик", пошаговое тестирование.

Критерии завершения тестирования.

Понятие отладки. Принципы отладки. Метод грубой силы, метод индукции, метод дедукции, отладка методом тестирования.

### **Применение методики решения задач на ЭВМ для построения решений простейших типовых задач**

Применение методики решения задач с помощью ЭВМ для видов задач: простейшие, содержащие основные управляющие структуры, содержащие рекуррентные соотношения, содержащие подпрограммы, содержащие обработку массивов, содержащие обработку файлов, содержащие рекурсию, содержащие алгоритмы с возвратами.

### **Итоговое контрольное мероприятие**

Итоговая комплексная контрольная работа. Студенты должны продемонстрировать умение реализовать простейшие методы сортировки, решать типовые задачи, содержащие основные управляющие структуры данных, рекуррентные соотношения, рекурсию.

## **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. Аляев Ю. А., Гладков В. П., Козлов О. А. Практикум по алгоритмизации и программированию на языке Паскаль: учеб. пособие для высш. и сред. учеб. заведений/Ю.А. Аляев, В. П. Гладков, О. А. Козлов.-М.: Финансы и статистика, 2007, ISBN 978-5-279-02771-2.-528.-Библиогр.: с. 524-527
2. Программирование на языке высокого уровня C/C++ [Электронный ресурс]: конспект лекций/ — Электрон. текстовые данные.— М.: Московский государственный строительный университет, Ай Пи Эр Медиа, ЭБС АСВ, 2016.— 140 с.— Режим доступа: <http://www.iprbookshop.ru/48037>.— ЭБС «IPRbooks» <http://www.iprbookshop.ru/48037.html>
3. Аляев Ю. А., Козлов О. А. Алгоритмизация и языки программирования Pascal, C++, Visual Basic: учебно-справочное пособие для курсантов воен. учеб. заведений и училищ, студентов техн. вузов, учащихся спец. калссов школ/Ю. А. Аляев, О. А. Козлов.-М.: Финансы и статистика, 2007, ISBN 978-5-279-02294-6.-320.-Библиогр.: с. 318-319
4. Королев Л. Н., Миков А. И. Информатика: введение в компьютерные науки: [учебник для вузов]/Л. Н. Королев, А. И. Миков.-Москва: Высшая школа, 2012, ISBN 978-5-4372-0020-9.-3661.-Библиогр.: с. 346-347

### Дополнительная:

1. Кнут Д. Э. Искусство программирования Т. 2. Получисленные алгоритмы/Д. Э. Кнут ; ред. Ю. В. Козаченко ; transl.: В. Т. Тертышный, И. В. Красиков.-3-е изд..-М.; СПб.; Киев: Вильямс, 2004, ISBN 5-8459-0081-6.-832
2. Вирт Н. Алгоритмы и структуры данных/Н. Вирт ; пер. Д. Б. Подшивалов.-Москва: Мир, 1989, ISBN 5-03-001045-9.-360.
3. Вирт Н. Программирование на языке модуля-2/Н. Вирт ; пер.: В. А. Серебряков, В. М. Ходукин ; ред. В. М. Курочкина.-Москва: Мир, 1987.-222.
4. Лорин Г. Сортировка и системы сортировки/Г. Лорин ; пер. Р. Л. Смелянский.-М.: Наука, 1983.-384.
5. Кнут Д. Э. Искусство программирования Т. 3. Сортировка и поиск/Д. Э. Кнут ; ред. Ю. В. Козаченко ; transl.: В. Т. Тертышный, И. В. Красиков.-2-е изд..-М.; СПб.; Киев: Вильямс, 2004, ISBN 5-8459-0082-4.-832
6. Липский В. Комбинаторика для программистов/В. Липский ; пер.: В. А. Евстигнеев, О. А. Логинова ; ред. А. П. Ершов.-Москва: Мир, 1988.-213.
7. Кнут Д. Э. Искусство программирования Т. 1. Основные алгоритмы/Д. Э. Кнут ; ред. Ю. В. Козаченко ; transl.: С. Г. Тригуб, Ю. Г. Гордиенко, И. В. Красикова.-3-е изд..-М.; СПб.; Киев: Вильямс, 2005, ISBN 5-8459-0080-8.-720

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

<http://www.psu.ru/elektronnye-resursy-dlya-psu> Электронные ресурсы для ПГНИУ

<http://www.mathnet.ru/> Общероссийский математический портал

<http://window.edu.ru/> Единое окно доступа к образовательным ресурсам

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

Образовательный процесс по дисциплине **Методы программирования I** 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:

- приложение позволяющее просматривать и воспроизводить медиаконтент PDF-файлов «AdobeAcrobatReader DC»;
- 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**).

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 лабораторных работ требуется компьютерный класс. Состав оборудования определен в Паспорте компьютерного класса.

Для групповых (индивидуальных) консультаций - аудитория, оснащенная презентационной техникой (проектор, экран, компьютер/ноутбук) с соответствующим программным обеспечением, 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  
Методы программирования I**

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

**ПК.3**

**Способность применять инструментальные средства разработки для создания защищенных программных средств и систем; работать с программными средствами прикладного, системного и специализированного назначения для обеспечения информационной безопасности**

<b>Компетенция (индикатор)</b>	<b>Планируемые результаты обучения</b>	<b>Критерии оценивания результатов обучения</b>
<p><b>ПК.3.1</b> Использует современные языки, методы, технологии, системы и инструментальные средства программирования для разработки защищенных программных средств и систем прикладного и специализированного назначения</p>	<p>Знать современные языки, методы, технологии, системы и инструментальные средства программирования. Уметь использовать современные языки, методы, технологии, системы и инструментальные средства программирования для разработки защищенных программных средств и систем прикладного и специализированного назначения.</p>	<p align="center"><b>Неудовлетворител</b> Не способен использовать современные языки, методы, технологии, системы и инструментальные средства программирования для разработки защищенных программных средств и систем прикладного и специализированного назначения.</p> <p align="center"><b>Удовлетворительн</b> Способен со значительными затруднениями использовать современные языки, методы, технологии, системы и инструментальные средства программирования для разработки защищенных программных средств и систем прикладного и специализированного назначения.</p> <p align="center"><b>Хорошо</b> Способен с незначительными затруднениями использовать современные языки, методы, технологии, системы и инструментальные средства программирования для разработки защищенных программных средств и систем прикладного и специализированного назначения.</p> <p align="center"><b>Отлично</b> Способен без затруднений использовать современные языки, методы, технологии, системы и инструментальные средства программирования для разработки защищенных программных средств и систем прикладного и специализированного назначения.</p>
<p><b>ПК.3.2</b> Применяет опыт</p>	<p>Знать современные языки, методы и технологии</p>	<p align="center"><b>Неудовлетворител</b> Не способен применять опыт разработки</p>

Компетенция (индикатор)	Планируемые результаты обучения	Критерии оценивания результатов обучения
разработки приложений и программных прототипов решений прикладных задач на основе современных языков, методов и технологий программирования для защиты программ и данных	<p>программирования. Уметь использовать современные языки, методы и технологии программирования для защиты программ и данных. Владеть навыками разработки приложений и программных прототипов решений прикладных задач на основе современных языков, методов и технологий программирования для защиты программ и данных.</p>	<p><b>Неудовлетворител</b> приложений и программных прототипов решений прикладных задач на основе современных языков, методов и технологий программирования для защиты программ и данных.</p> <p><b>Удовлетворительн</b> Способен со значительными затруднениями применять опыт разработки приложений и программных прототипов решений прикладных задач на основе современных языков, методов и технологий программирования для защиты программ и данных.</p> <p><b>Хорошо</b> Способен с незначительными затруднениями применять опыт разработки приложений и программных прототипов решений прикладных задач на основе современных языков, методов и технологий программирования для защиты программ и данных.</p> <p><b>Отлично</b> Способен без затруднений применять опыт разработки приложений и программных прототипов решений прикладных задач на основе современных языков, методов и технологий программирования для защиты программ и данных.</p>

## ПК.2

### Способность выполнять работы по созданию (модификации) и сопровождению информационных систем, автоматизирующих задачи организационного управления и бизнес-процессы

Компетенция (индикатор)	Планируемые результаты обучения	Критерии оценивания результатов обучения
<p><b>ПК.2.2</b> Применяет навыки использования библиотек для разработки программных комплексов, в том числе при коллективной</p>	<p>Знать набор основных библиотек для разработки программных комплексов. Владеть навыками использования библиотек для разработки программных комплексов, в том числе при коллективной разработке.</p>	<p><b>Неудовлетворител</b> Не способен применять навыки использования библиотек для разработки программных комплексов, в том числе при коллективной разработке.</p> <p><b>Удовлетворительн</b> Способен со значительными затруднениями применять навыки использования библиотек для разработки программных комплексов, в</p>

<b>Компетенция (индикатор)</b>	<b>Планируемые результаты обучения</b>	<b>Критерии оценивания результатов обучения</b>
разработке		<p style="text-align: center;"><b>Удовлетворительн</b></p> <p>том числе при коллективной разработке.</p> <p style="text-align: center;"><b>Хорошо</b></p> <p>Способен с незначительными затруднениями применять навыки использования библиотек для разработки программных комплексов, в том числе при коллективной разработке.</p> <p style="text-align: center;"><b>Отлично</b></p> <p>Способен без затруднений применять навыки использования библиотек для разработки программных комплексов, в том числе при коллективной разработке.</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 50 до 60

«неудовлетворительно» / «незачтено» менее 50 балла

Компетенция (индикатор)	Мероприятие текущего контроля	Контролируемые элементы результатов обучения
<b>ПК.3.1</b> Использует современные языки, методы, технологии, системы и инструментальные средства программирования для разработки защищенных программных средств и систем прикладного и специализированного назначения	Системы счисления <b>Письменное контрольное мероприятие</b>	Письменная работа, включающая в себя 5 примеров на простейшее преобразование информации
<b>ПК.3.2</b> Применяет опыт разработки приложений и программных прототипов решений прикладных задач на основе современных языков, методов и технологий программирования для защиты программ и данных	Алгоритм Маркова <b>Письменное контрольное мероприятие</b>	Письменная работа, включающая выполнение записи алгоритмов одним из возможных способов записи, а также выполнение алгоритма, представленного одним из возможных способов записи

<b>Компетенция (индикатор)</b>	<b>Мероприятие текущего контроля</b>	<b>Контролируемые элементы результатов обучения</b>
<p><b>ПК.2.2</b> Применяет навыки использования библиотек для разработки программных комплексов, в том числе при коллективной разработке</p> <p><b>ПК.3.2</b> Применяет опыт разработки приложений и программных прототипов решений прикладных задач на основе современных языков, методов и технологий программирования для защиты программ и данных</p>	<p>Правила организации рекуррентных вычислений</p> <p><b>Письменное контрольное мероприятие</b></p>	<p>Письменная работа, включающая в себя пример для применения правил организации рекуррентных вычислений, а также реализации записи разработанного алгоритма на одном из языков программирования (Pascal, C, любая версия)</p>
<p><b>ПК.2.2</b> Применяет навыки использования библиотек для разработки программных комплексов, в том числе при коллективной разработке</p> <p><b>ПК.3.2</b> Применяет опыт разработки приложений и программных прототипов решений прикладных задач на основе современных языков, методов и технологий программирования для защиты программ и данных</p>	<p>Правила организации рекурсивных вычислений</p> <p><b>Письменное контрольное мероприятие</b></p>	<p>Письменная работа, включающая в себя пример для применения правил организации рекурсивных вычислений, а также реализации записи разработанного алгоритма на одном из языков программирования (Pascal, C, любая версия)</p>
<p><b>ПК.2.2</b> Применяет навыки использования библиотек для разработки программных комплексов, в том числе при коллективной разработке</p> <p><b>ПК.3.2</b> Применяет опыт разработки приложений и программных прототипов решений прикладных задач на основе современных языков, методов и технологий программирования для защиты программ и данных</p>	<p>Классы сложности задач</p> <p><b>Письменное контрольное мероприятие</b></p>	<p>Письменная работа, включающая в себя 3 примера на применение правил вычисления сложности алгоритмов различного вида</p>



<b>Компетенция (индикатор)</b>	<b>Мероприятие текущего контроля</b>	<b>Контролируемые элементы результатов обучения</b>
<p><b>ПК.2.2</b> Применяет навыки использования библиотек для разработки программных комплексов, в том числе при коллективной разработке</p> <p><b>ПК.3.1</b> Использует современные языки, методы, технологии, системы и инструментальные средства программирования для разработки защищенных программных средств и систем прикладного и специализированного назначения</p>	<p>Хеширование</p> <p><b>Письменное контрольное мероприятие</b></p>	<p>Письменная работа, выполненная в среде программирования одного из языков программирования (Pascal, C, любая версия), содержащая рабочую программу с одним из предложенных для реализации алгоритмов сортировки последовательности или хеширования.</p>
<p><b>ПК.2.2</b> Применяет навыки использования библиотек для разработки программных комплексов, в том числе при коллективной разработке</p> <p><b>ПК.3.1</b> Использует современные языки, методы, технологии, системы и инструментальные средства программирования для разработки защищенных программных средств и систем прикладного и специализированного назначения</p>	<p>Тестирование и отладка программ</p> <p><b>Письменное контрольное мероприятие</b></p>	<p>Письменная работа, включающая в себя расшифровку всей терминологической базы дисциплины</p>

<b>Компетенция (индикатор)</b>	<b>Мероприятие текущего контроля</b>	<b>Контролируемые элементы результатов обучения</b>
<p><b>ПК.2.2</b> Применяет навыки использования библиотек для разработки программных комплексов, в том числе при коллективной разработке</p> <p><b>ПК.3.1</b> Использует современные языки, методы, технологии, системы и инструментальные средства программирования для разработки защищенных программных средств и систем прикладного и специализированного назначения</p>	<p>Применение методики решения задач на ЭВМ для построения решений простейших типовых задач</p> <p><b>Письменное контрольное мероприятие</b></p>	<p>Письменная работа, содержащая реализацию решения одной из предложенных типовых задач, выполненную на одном из языков программирования (Pascal, C, любая версия)</p>
<p><b>ПК.2.2</b> Применяет навыки использования библиотек для разработки программных комплексов, в том числе при коллективной разработке</p> <p><b>ПК.3.2</b> Применяет опыт разработки приложений и программных прототипов решений прикладных задач на основе современных языков, методов и технологий программирования для защиты программ и данных</p> <p><b>ПК.3.1</b> Использует современные языки, методы, технологии, системы и инструментальные средства программирования для разработки защищенных программных средств и систем прикладного и специализированного назначения</p>	<p>Итоговое контрольное мероприятие</p> <p><b>Итоговое контрольное мероприятие</b></p>	<p>Письменная работа, включающая в себя ответ на теоретический вопрос, реализацию типового алгоритма обработки информации, а также составление алгоритмов для решения 2 простейших типовых задач и их реализацию на одном из языков программирования (Pascal, C, любая версия)</p>

### Спецификация мероприятий текущего контроля

#### Системы счисления

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

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

Максимальный балл, выставяемый за мероприятие промежуточной аттестации: **7**

Проходной балл: **3.5**

Показатели оценивания	Баллы
Уметь выполнить простейшее преобразование информации между системами счисления	5
Уметь вычислить объем информации по заданным параметрам	2

### **Алгоритм Маркова**

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

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

Максимальный балл, выставяемый за мероприятие промежуточной аттестации: **4**

Проходной балл: **2**

Показатели оценивания	Баллы
Уметь выполнить алгоритм, записанный одним из возможных способов записи алгоритма	2
Уметь записать алгоритм одним из возможных способов записи алгоритма	2

### **Правила организации рекуррентных вычислений**

Продолжительность проведения мероприятия промежуточной аттестации: **1.5 часа**

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

Максимальный балл, выставяемый за мероприятие промежуточной аттестации: **5**

Проходной балл: **2.5**

Показатели оценивания	Баллы
Уметь записать оптимальный алгоритм для представленного примера	3
Уметь применить правила организации рекуррентных вычислений для представленного в задании примера и записать его на одном из языков программирования (C, Pascal, любая версия)	2

### **Правила организации рекурсивных вычислений**

Продолжительность проведения мероприятия промежуточной аттестации: **1.5 часа**

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

Максимальный балл, выставяемый за мероприятие промежуточной аттестации: **6**

Проходной балл: **3**

Показатели оценивания	Баллы
Уметь выполнить рекурсивный алгоритм и записать результаты его выполнения	3
Уметь применить правила организации рекурсивных вычислений для представленного в задании примера	3

### **Классы сложности задач**

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

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

Максимальный балл, выставяемый за мероприятие промежуточной аттестации: **8**

Проходной балл: **4**

Показатели оценивания	Баллы
Уметь вычислить сложность алгоритма для представленного примера	3
Уметь обосновать вычисленную сложность алгоритма	2
Уметь вычислить вероятность выполнения алгоритма по его различным ветвям	2
Уметь построить управляющий граф алгоритма	1

### Хеширование

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

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

Максимальный балл, выставляемый за мероприятие промежуточной аттестации: **10**

Проходной балл: **5**

Показатели оценивания	Баллы
Уметь выполнить реализацию алгоритма на одном из языков программирования (C, Pascal, любая версия)	5
Уметь выполнить разработку алгоритма сортировки или хеширования последовательности	5

### Тестирование и отладка программ

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

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

Максимальный балл, выставляемый за мероприятие промежуточной аттестации: **10**

Проходной балл: **5**

Показатели оценивания	Баллы
Уметь осуществить поиск информации по определенному вопросу методологии программирования с помощью любых имеющихся на рабочем месте информационных технологий	5
Знать основные понятия в области методологии программирования	5

### Применение методики решения задач на ЭВМ для построения решений простейших типовых задач

Продолжительность проведения мероприятия промежуточной аттестации: **3 часа**

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

Максимальный балл, выставляемый за мероприятие промежуточной аттестации: **10**

Проходной балл: **5**

Показатели оценивания	Баллы
Уметь произвести тестирование и отладку разработанной программы	4
Уметь разработать алгоритм решения типовой задачи	3
Уметь выполнить реализацию разработанного алгоритма на одном из языков программирования (C, Pascal, любая версия)	3

### Итоговое контрольное мероприятие

Продолжительность проведения мероприятия промежуточной аттестации: **3 часа**

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

Максимальный балл, выставляемый за мероприятие промежуточной аттестации: **40**

Проходной балл: **20**

<b>Показатели оценивания</b>	<b>Баллы</b>
Знать теоретические основы методологии программирования	10
Уметь выполнить реализацию разработанного алгоритма на одном из языков программирования (C, Pascal, любая версия)	10
Уметь выполнить разработку алгоритма для простейшей типовой задачи	10
Уметь выполнить реализацию типового алгоритма обработки информации	10