

Документ подписан простой электронной подписью  
Информация о владельце:  
ФИО: Смирнов Сергей Николаевич  
Должность: врио ректора  
Дата подписания: 08.11.2023 16:53:56  
Уникальный программный ключ:  
69e375c64f7e975d4e8830e7b4fcc2ad1bf35f08

Министерство науки и высшего образования  
Российской Федерации

Федеральное государственное бюджетное  
образовательное учреждение высшего образования  
«Тверской государственный университет»

УТВЕРЖДАЮ  
Руководитель ООП  
*С.М. Дудаков*  
30 марта 2023 г.



Рабочая программа дисциплины (с аннотацией)  
**Теоретические основы информатики**

Направление подготовки  
09.03.03 — ПРИКЛАДНАЯ ИНФОРМАТИКА

Профиль подготовки  
ПРИКЛАДНАЯ ИНФОРМАТИКА В ЭКОНОМИКЕ

для студентов 1 курса  
ФОРМА ОБУЧЕНИЯ — очная

Составитель(и):

- к.ф.-м.н. Карлов Б.Н.

Тверь — 2023

# I. Аннотация

## 1. Цель и задачи дисциплины:

Главная цель курса — развить у студентов алгоритмическое мышление, дать общее представление обучающимся о методологии программирования, о различных парадигмах программирования (структурированные программы, программы с метками, функциональные программы), о методах верификации и оценки сложности алгоритмов.

## 2. Место дисциплины в структуре ООП

Дисциплина входит в раздел «Информатика и коммуникационно-информационные технологии» обязательной части блока 1.

**Предварительные знания и навыки.** Знание школьных курсов математики и информатики.

**Дальнейшее использование.** Полученные знания используются в последующем при изучении предметов: «Дискретная математика», «Методы программирования», «Математическая логика и теория алгоритмов» и других.

## 3. Объем дисциплины: 4 зач. ед., 144 акад. ч., в том числе:

**контактная аудиторная работа** лекций 45 ч., практических занятий 15 ч.,  
**контактная внеаудиторная работа** контроль самостоятельной работы 10 ч., в том числе курсовая (расчетно-графическая) работа 10 ч.;  
**самостоятельная работа** 74 ч., в том числе контроль 36 ч.

## 4. Перечень планируемых результатов обучения по дисциплине, соотнесенных с планируемыми результатами освоения образовательной программы:

Планируемые результаты освоения образовательной программы (формируемые компетенции)	Планируемые результаты обучения по дисциплине
ОПК-4, Способен участвовать в разработке стандартов, норм и правил, а также технической документации, связанной с профессиональной деятельностью	ОПК-4.1, Знает основные стандарты оформления технической документации на различных стадиях жизненного цикла информационной системы ОПК-4.2, Применяет стандарты оформления технической документации на различных стадиях жизненного цикла информационной системы ОПК-4.3, Владеет навыками составления тех-

Планируемые результаты освоения образовательной программы (формируемые компетенции)	Планируемые результаты обучения по дисциплине
ОПК-7, Способен разрабатывать алгоритмы и программы, пригодные для практического применения	<p>нической документации на различных этапах жизненного цикла информационной системы</p> <p>ОПК-7.1, Знает основные языки программирования и работы с базами данных, операционные системы и оболочки, современные программные среды разработки информационных систем и технологий</p> <p>ОПК-7.2, Применяет языки программирования и работы с базами данных, современные программные среды разработки информационных систем и технологий для автоматизации бизнес-процессов, решения прикладных задач различных классов, ведения баз данных и информационных хранилищ</p> <p>ОПК-7.3, Владеет навыками программирования, отладки и тестирования прототипов программно-технических комплексов задач</p>

**5. Форма промежуточной аттестации и семестр прохождения:**

экзамен в 1 семестре

**6. Язык преподавания:**

русский

## II. Содержание дисциплины, структурированное по темам (разделам) с указанием отведенного на них количества академических часов и видов учебных занятий

Для студентов очной формы обучения

Учебная программа — наименование разделов и тем	Всего (час.)	Контактная работа (час.)				Сам. раб., в т.ч. контроль (час.)	
		Лекции		Практ. занятия / Лаб. работы			Контроль сам. раб., в т.ч. курсовая работа
		Всего	В т.ч. практ. подг.	Всего	В т.ч. практ. подг.		
1	2	3	4	5	6	7	8
Общие сведения об алгоритмах и исполнителях	6	2		1/0		0	3
Структурное программирование	42	16		5/0		0	21
Программы с метками	32	9		3/0		0	20
Функциональное программирование	32	9		3/0		10	10
Сложность алгоритмов	32	9		3/0		0	20
Итого	144	45	0	15/0	0/0	10	74

### Учебная программа дисциплины

#### 1. Общие сведения об алгоритмах и исполнителях

- История представлений об алгоритмах и исполнителях
- Основные сведения об алгоритмах и исполнителях. Языки программирования
- Классификация языков программирования и алгоритмов. Примеры алгоритмов
- Общие элементы языков программирования

#### 2. Структурное программирование

- Основные конструкции структурного программирования: присваивание, следование, ветвление, цикл. Синтаксис структурированных программ
- Состояния. Семантика структурированных программ. Примеры
- Преобразования структурных программ
- Основные свойства структурированных программ, замена переменных

#### 3. Программы с метками

- Синтаксис и семантика программ с метками
- Свойства программ с метками. Программы с метками как модель ЭВМ

- Эквивалентность программ с метками и структурных программ

#### 4. Функциональное программирование

- Подпрограммы, синтаксис и семантика
- Рекурсивные алгоритмы, функциональные программы
- Эквивалентность функциональных и структурированных программ

#### 5. Сложность алгоритмов

- Меры сложности алгоритмов, общие свойства
- Основные ресурсы: время и память. Эвристический анализ ресурсоемкости.
- Максимальные и средние оценки ресурсоемкости. Анализ ресурсоемкости алгоритмов.
- Эвристические приемы оценки времени и памяти

### III. Образовательные технологии

Учебная программа — наименование разделов и тем	Вид занятия	Образовательные технологии
Общие сведения об алгоритмах и исполнителях	лекции, практические занятия	изложение теоретического материала
Структурное программирование	лекции, практические занятия	изложение теоретического материала
Программы с метками	лекции, практические занятия	изложение теоретического материала
Функциональное программирование	лекции, практические занятия	изложение теоретического материала, выполнение РГР
Сложность алгоритмов	лекции, практические занятия	изложение теоретического материала

### IV. Оценочные материалы для проведения текущей и промежуточной аттестации

#### Типовые контрольные задания и/или критерии для проверки индикатора ОПК-4.1

Требования к обучающемуся	Типовые контрольные задания для оценки знаний, умений, навыков	Показатели и критерии оценивания, шкала оценивания
Знать синтаксис и семантику структурированных программ	Примеры вопросов к экзамену: <ul style="list-style-type: none"> <li>• Выражение. Основные конструкции структурированных программ: присваивание, следование, полное и неполное ветвление, цикл.</li> </ul>	оценка 3 — знает неформальное определение синтаксиса и семантики структурированных программ,

Требования к обучающемуся	Типовые контрольные задания для оценки знаний, умений, навыков	Показатели и критерии оценивания, шкала оценивания
	<ul style="list-style-type: none"> <li>Синтаксические свойства структурированных программ. Лемма о сбалансированности. Лемма о суффиксе программы.</li> <li>Состояние. Значение выражения на состоянии. Семантика структурированной программы. Условие заикливание. Корректность семантики следования. Эквивалентность алгоритмов.</li> </ul>	оценка 4 — знает формальное определение синтаксиса и семантики структурированных программ, некоторые свойства программ, оценка 5 — кроме того знает доказательства свойств программ

### Типовые контрольные задания и/или критерии для проверки индикатора ОПК-4.2

Требования к обучающемуся	Типовые контрольные задания для оценки знаний, умений, навыков	Показатели и критерии оценивания, шкала оценивания
Знать синтаксис и семантику программ с метками	<p>Примеры вопросов к экзамену:</p> <ul style="list-style-type: none"> <li>Оператор присваивания. Оператор ветвления. Программа с метками.</li> <li>Расширенное состояние. Семантика программ с метками. Условие заикливания.</li> <li>Теорема о построении программы с метками по структурированной программе.</li> <li>Теорема о построении структурированной программы по программе с метками.</li> </ul>	оценка 3 — знает определение синтаксиса и семантики программ с метками, оценка 4 — кроме того знает схемы доказательства теорем об эквивалентности, оценка 5 — кроме того может формально доказать эти теоремы

### Типовые контрольные задания и/или критерии для проверки индикатора ОПК-4.3

Требования к обучающемуся	Типовые контрольные задания для оценки знаний, умений, навыков	Показатели и критерии оценивания, шкала оценивания
Знать понятие подпрограммы, основные свойства подпрограмм	<p>Примеры вопросов к экзамену:</p> <ul style="list-style-type: none"> <li>Подпрограмма. Вызов подпрограммы. Выражение. Значение выражения. Семантика программы, содержащей подпрограммы.</li> <li>Граф зависимости. Рекурсия. Список вложенных вызовов. Множество вложенных вызовов. Дерево вложенных вызовов. Лес вложенных вызовов.</li> <li>Аппликативная программа. Функциональная программа.</li> <li>Теорема о построении функциональной программы. Теорема об удалении подпрограмм.</li> </ul>	оценка 3 — знает понятие подпрограммы, семантику выражений с вызовами подпрограмм, знает понятие рекурсии, оценка 4 — кроме того знает понятия списка, множества, дерева и леса вложенных вызовов, схемы доказательства теорем об эквивалентности структурированных и функциональных программ, оценка 5 — кроме того знает до-

Требования к обучающемуся	Типовые контрольные задания для оценки знаний, умений, навыков	Показатели и критерии оценивания, шкала оценивания
		казательства теорем об эквивалентности

## Типовые контрольные задания и/или критерии для проверки индикатора ОПК-7.1

Требования к обучающемуся	Типовые контрольные задания для оценки знаний, умений, навыков	Показатели и критерии оценивания, шкала оценивания
Уметь писать структурированные программы и доказывать их правильность, понимать написанные программы	<p>Примеры задач для контрольных работ:</p> <ul style="list-style-type: none"> <li>• Определить, каким будет состояние, полученное применением следующей программы к состоянию <math>\{(x, a), (y, b)\}</math>. Обосновать. <pre> u=0; while 0 &lt; x do   if x % 10 = 1 then     u = s(u);   end;   x = x / 100; end; </pre> </li> <li>• Написать алгоритм и доказать его правильность. Вход: число <math>x</math>, результат: количество разрядов (начиная с младшего), на которых и в десятичном, и в восьмеричном представлении числа <math>x</math> стоят одни и те же цифры. Использовать «&lt;», «=», «%», «/», <math>s</math>.</li> </ul>	оценка 3 — умеет писать структурированные программы и понимать написанные программы, может неформально описать, как работает программа, оценка 4 — кроме того может сформулировать утверждения о работе программы, оценка 5 — кроме того может формально доказать правильность сформулированных утверждений по определению семантики
Знать основные свойства структурированных программ	<p>Примеры вопросов к экзамену:</p> <ul style="list-style-type: none"> <li>• Лемма о сохранении значения. Лемма о неиспользуемых переменных.</li> <li>• Замена переменных в выражении. Замена переменных в программе. Лемма о замене переменных в выражении. Лемма о замене переменных в программе.</li> <li>• Простые программы. Леммы об упрощении присваиваний, ветвлений, циклов. Теорема о простой программе.</li> <li>• Теорема о подстановке.</li> </ul>	оценка 3 — знает понятия замены переменных, простой программы, подстановки, оценка 4 — кроме того знает общие схемы доказательства перечисленных утверждений, оценка 5 — кроме того знает формальные доказательства перечисленных утверждений

## Типовые контрольные задания и/или критерии для проверки индикатора ОПК-7.2

Требования к обучающемуся	Типовые контрольные задания для оценки знаний, умений, навыков	Показатели и критерии оценивания, шкала оценивания
Уметь самостоятельно писать программы и выполнять их верификацию	<p>Примеры тем для расчетно-графических работ:</p> <ul style="list-style-type: none"> <li>• Напишите структурированную программу и докажите её правильность по определению семантики. Найти количество чисел на интервале <math>[x, y)</math>, в шестнадцатеричной записи которых после цифры 3 не стоит цифра 4. Использовать <math>s</math> — приращение единицы, +, -, ×, : — целочисленное деление, % —</li> </ul>	оценка 3 — умеет писать программы разных типов, может неформально описать их работу, оценка 4 — кроме того уме-

Требования к обучающемуся	Типовые контрольные задания для оценки знаний, умений, навыков	Показатели и критерии оценивания, шкала оценивания
	<p>остаток от деления, <math>&lt;</math>, <math>=</math>.</p> <ul style="list-style-type: none"> <li>● Написать программу с метками, обосновать правильность и преобразовать ее в структурированную программу. Найти НОД всех цифр числа <math>x</math>. Использовать <math>s</math> — прибавление 1, <math>=</math>, <math>&lt;</math>, <math>+</math>, <math>-</math>.</li> <li>● Написать функциональную программу и доказать ее правильность по индукции. Написать структурированную программу и определить максимальное время и память в зависимости от длины входных данных. Найти <math>x</math>-е по счету простое число. Использовать <math>s</math> — прибавление 1, <math>=</math>, <math>&lt;</math>, <math>+</math>, <math>-</math>, <math>\times</math>, <math>/</math>, <math>\%</math>.</li> </ul>	<p>ет формулировать утверждения о работе программ, оценка 5 — кроме того может доказать сформулированные утверждения,</p>
<p>Уметь строить программы с метками, преобразовывать программы с метками в структурированные программы и обратно</p>	<p>Примеры задач для контрольных работ:</p> <ul style="list-style-type: none"> <li>● Написать программу с метками и доказать правильность по определению семантики программ с метками. Определить, является ли заданное число <math>x</math> квадратом целого числа. Использовать <math>s</math>, <math>&lt;</math>, <math>=</math>, <math>/</math>, <math>\%</math>.</li> <li>● Напишите структурированную программу для вычисления функции Эйлера, используя операции <math>s</math>, <math>&lt;</math>, <math>=</math>, <math>+</math>, <math>-</math>, <math>\times</math>, <math>/</math>, <math>\%</math>. Преобразуйте ее в эквивалентную программу с метками.</li> <li>● Напишите программу с метками для проверки числа на простоту, используя операции <math>s</math>, <math>+</math>, <math>-</math>, <math>&lt;</math>, <math>=</math>. Преобразуйте ее в структурированную программу, вычисляющую ту же функцию.</li> </ul>	<p>оценка 3 — умеет писать программы с метками, может неформально описать как работает программа, оценка 4 — кроме того умеет преобразовывать программы с метками в структурированные и обратно, может сформулировать утверждения о работе программы, оценка 5 — кроме того умеет доказывать правильность сформулированных утверждений по определению семантики</p>
<p>Уметь писать функциональные программы, преобразовывать их в структурированные и обратно</p>	<p>Примеры задач для контрольных работ:</p> <ul style="list-style-type: none"> <li>● Написать функциональную программу и доказать правильность по определению семантики функциональных программ. Найти количество чисел Фибоначчи меньших заданного числа <math>x</math>. Использовать <math>s</math>, <math>&lt;</math>, <math>=</math>, <math>+</math>.</li> <li>● Напишите функциональную программу, которая находит наименьшее основание системы счисления, в которой запись числа <math>x</math> не содержит нулей, используя операции <math>s</math>, <math>+</math>, <math>-</math>, <math>\times</math>, <math>/</math>, <math>\%</math>, <math>&lt;</math>, <math>=</math>. Преобразуйте ее в структурированную программу.</li> <li>● Напишите структурированную программу для вычисления функции <math>x!</math>, используя операции <math>s</math>, <math>+</math>, <math>-</math>, <math>&lt;</math>, <math>=</math>. Преобразуйте ее в функциональную программу.</li> </ul>	<p>оценка 3 — умеет писать программы с метками, может неформально описать как работает программа, оценка 4 — кроме того умеет преобразовывать функциональные программы в структурированные и обратно, может сформулировать утверждения о работе программы, оценка 5 — кроме того умеет доказывать правильность сформулированных утверждений по определению семантики</p>



## Типовые контрольные задания и/или критерии для проверки индикатора ОПК-7.3

Требования к обучающемуся	Типовые контрольные задания для оценки знаний, умений, навыков	Показатели и критерии оценивания, шкала оценивания
Знать понятия временной и емкостной сложности алгоритмов	<p>Примеры вопросов к экзамену:</p> <ul style="list-style-type: none"> <li>• Объем памяти для хранения чисел. Лемма о минимальном объеме.</li> <li>• Вычисление выражения, теста, программы. Время и память вычисления. Максимальное и среднее время, максимальная и средняя память.</li> <li>• Лемма о вычислении программ.</li> </ul>	оценка 3 — имеет представление о времени и памяти программы, оценка 4 — знает определения вычисления, временной и емкостной сложности, оценка 5 — кроме того знает свойства вычислений и может их доказать
Уметь оценивать сложность простых алгоритмов	<p>Примеры задач для контрольных работ:</p> <ul style="list-style-type: none"> <li>• Напишите структурированную программу для вычисления <math>n</math>-го числа Фибоначчи, используя операции <math>=, &lt;, s</math>. Оцените максимальное время и память (по порядку) в зависимости от размера входных данных.</li> <li>• Напишите функциональную программу для вычисления суммы всех факториалов на интервале <math>[x, y)</math>, используя операции <math>=, &lt;, s, +, -</math>. Оцените максимальное время и память (по порядку) в зависимости от размера входных данных.</li> </ul>	оценка 3 — умеет оценивать по порядку время и память простейших программ, оценка 4 — умеет оценивать время и память более сложных программ, оценка 5 — кроме того умеет получать более точные оценки времени и памяти

## V. Учебно-методическое и информационное обеспечение дисциплины

### 1. Рекомендованная литература

#### а) Основная литература

- [1] Дудаков, С. М. Математическое введение в информатику : учебник по дисциплине «Теоретические основы информатики» / С. М. Дудаков, Б. Н. Карлов. – Издание третье, исправленное и дополненное. – Тверь : Тверской государственный университет, 2020. – 320 с. – <https://elibrary.ru/vngtun>.
- [2] А. Шень. Программирование: теоремы и задачи (с1) 2-е изд., М.: МЦНМО, 2004, 296 с. — Режим доступа: <http://www.mccme.ru/free-books/shen/shen-progbook.pdf>
- [3] Кауфман, В. Ш. Языки программирования. Концепции и принципы [Электронный ресурс] / В. Ш. Кауфман. — М.: ДМК Пресс, 2010. — 464 с.: ил. — ISBN 978-5-94074-622-5. — Режим доступа: <http://znanium.com/bookread2.php?book=409077> (ЭБС znanium.com)

#### б) Дополнительная литература

- [4] Кудрявцев, К. Я. Функциональное программирование: конспект лекций : учебное пособие / К. Я. Кудрявцев. — Москва : НИЯУ МИФИ, 2020. — 112

с. — ISBN 978-5-7262-2672-9. — Текст : электронный // Лань : электронно-библиотечная система. — URL: <https://e.lanbook.com/book/175424>. — Режим доступа: для авториз. пользователей.

- [5] Прохорова О.В. Информатика [Электронный ресурс]: учебник/ Прохорова О.В. — Электрон. текстовые данные. — Самара: Самарский государственный архитектурно-строительный университет, ЭБС АСВ, 2013. — 106 с. — Режим доступа: <http://www.iprbookshop.ru/20465.html>. — ЭБС «IPRbooks»

## 2. Программное обеспечение

Наименование помещений	Программное обеспечение
Ауд. 201а (компьютерная лаборатория ПМиК) (170002, Тверская обл., г. Тверь, пер. Садовый, д. 35)	Перечень программного обеспечения (со свободными лицензиями): Linux Kubuntu, KDE, TeXLive, TeXStudio, LibreOffice, GIMP, Gwenview, ImageMagick, Okular, Skanlite, Google Chrome, KDE Connect, Konversation, KRDC, KTorrent, Thunderbird, Elisa, VLC media player, PulseAudio, KAppTemplate, KDevelop, pgAdmin4, PostgreSQL, Qt, QtCreator, R, RStudio, Visual Studio Code, Perl, Python, Ruby, clang, clang++, gcc, g++, nasm, flex, bison, Maxima, Octave, Dolphin, HTop, Konsole, KSystemLog, Xterm, Ark, Kate, KCalc, Krusader, Spectacle, Vim.

## 3. Современные профессиональные базы данных и информационные справочные системы

- [1] ЭБС «ZNANIUM.COM» <http://www.znanium.com>  
[2] ЭБС «Университетская библиотека онлайн» <https://biblioclub.ru>  
[3] ЭБС IPRbooks <http://www.iprbookshop.ru>  
[4] ЭБС «Лань» <http://e.lanbook.com>  
[5] ЭБС «Юрайт» <https://urait.ru>  
[6] ЭБС ТвГУ <http://megapro.tversu.ru/megapro/Web>  
[7] Научная электронная библиотека eLIBRARY.RU (подписка на журналы) [https://elibrary.ru/projects/subscription/rus\\_titles\\_open.asp](https://elibrary.ru/projects/subscription/rus_titles_open.asp)  
[8] Репозиторий ТвГУ <http://eprints.tversu.ru>

## 4. Перечень ресурсов информационно-телекоммуникационной сети «Интернет», необходимых для освоения дисциплины

- [1] Computer Programming Tutorial, [https://www.tutorialspoint.com/computer\\_programming/index.htm](https://www.tutorialspoint.com/computer_programming/index.htm)

## VI. Методические материалы для обучающихся по освоению дисциплины

Важной составляющей данного раздела РПД являются требования к рейтинг-контролю с указанием баллов, распределенных между модулями и видами работы обучающихся.

Максимальная сумма баллов по учебной дисциплине, заканчивающейся экзаменом, по итогам семестра составляет 60 баллов. Распределение баллов по модулям устанавливается преподавателем и может корректироваться.

Обучающемуся, набравшему 40–54 балла, при подведении итогов семестра (на последнем занятии по дисциплине) в рейтинговой ведомости учета успеваемости и зачетной книжке может быть выставлена оценка «удовлетворительно».

Обучающемуся, набравшему 55–57 баллов, при подведении итогов семестра (на последнем занятии по дисциплине) в графе рейтинговой ведомости учета успеваемости «Премияльные баллы» может быть добавлено 15 баллов и выставлена экзаменационная оценка «хорошо».

Обучающемуся, набравшему 58–60 баллов, при подведении итогов семестра (на последнем занятии по дисциплине) в графе рейтинговой ведомости учета успеваемости «Премияльные баллы» может быть добавлено 27 баллов и выставлена экзаменационная оценка «отлично». В каких-либо иных случаях добавление премияльных баллов не допускается.

Обучающийся, набравший до 39 баллов включительно, сдает экзамен.

### Примеры задач для подготовки к контрольным работам

1. Напишите структурированную программу, которая получает на вход натуральных числа  $x$  и  $y$  и проверяет условие  $x < y$ . Разрешается использовать только присваивание, прибавление единицы и проверку на равенство. Докажите правильность написанной программы.
2. Напишите структурированную программу, которая определяет, сколько чисел на интервале  $[x, y)$  состоят только из двоичных цифр в пятеричной системе счисления. Докажите правильность написанной программы по определению семантики. Разрешается использовать операции  $s$ ,  $+$ ,  $-$ ,  $\times$ ,  $/$ ,  $\%$ , сравнения  $<$  и  $=$ .
3. Докажите, что никакая структурированная программа без циклов, использующая только операцию прибавления единицы и сравнения  $<$ ,  $=$ , не может вычислять функцию  $x + y$ .
4. Напишите программу с метками для вычисления функции  $x^y$ . Разрешается использовать только сравнения  $=$ ,  $<$  и прибавление единицы. Докажите правильность написанной программы по определению семантики.
5. В теореме о структуризации была построена структурированная программа, содержащая две новые переменные. Модифицируйте доказательство теоремы, так чтобы использовалась только одна новая переменная.

6. Напишите функциональную программу для проверки числа на простоту. Разрешается использовать только сравнения  $<$ ,  $=$  и прибавление единицы. Докажите правильность написанной программы.
7. Доказать, что если функциональная программа не содержит рекурсии, то она эквивалентна программе без циклов, следований и подпрограмм.
8. Напишите структурированную программу, которая находит первую степень двойки большую  $x$ . Оцените ее время и память. Разрешается использовать операцию  $s$ , сравнения  $<$  и  $=$ .

## Требования к рейтинг контролю

**Расчетно-графическая работа 1.** Темы: структурированные программы. Пример задания:

Написать структурированную программу и доказать ее правильность по определению семантики структурированных программ. Найти число на интервале  $[x, y)$ , имеющее наибольшее количество делителей. Использовать  $s$  — прибавление 1,  $=$ ,  $<$ ,  $+$ ,  $-$ ,  $\times$ ,  $:$  — целочисленное деление,  $\%$  — остаток от деления.

За решение задачи выставляется максимум 10 баллов.

**Контрольная работа 1.** Темы: структурированные программы. Пример задания:

1. Определить, каким будет состояние, полученное применением следующей программы к состоянию  $\{(x, a), (u, b)\}$ . Обосновать.

```

u = 0;
while 0 < x do
  if x % 10 = 1 then
    u = s(u);
  end;
  x = x / 100;
end;

```

2. Написать алгоритм  $M_1$  и доказать его правильность. Вход: число  $x$ , результат: количество разрядов (начиная с младшего), на которых и в десятичном, и в восьмеричном представлении числа  $x$  стоят одни и те же цифры. Использовать « $<$ », « $=$ », « $\%$ », « $/$ », « $s$ ».

За решение каждой задачи выставляется максимум 5 баллов.

**Расчетно-графическая работа 2.** Темы: программы с метками. Пример задания:

Написать программу с метками и преобразовать ее в структурированную программу. Найти сумму чисел интервала  $[x; y)$ , в десятичной записи которых ровно половина цифр — нечетные. Использовать  $s$  — прибавление 1,  $=$ ,  $<$ ,  $+$ ,  $-$ ,  $\times$ ,  $:$  —

целочисленное деление, % — остаток от деления.

За решение задачи выставляется максимум 15 баллов.

**Контрольная работа 2.** Темы: программы с метками, функциональные программы. Пример задания:

1. Написать программу с метками и доказать правильность по определению семантики программ с метками. Определить, делится ли заданное число  $x$  на все числа, не превосходящие  $y$ . Использовать  $s, <, =, /, \times$ .
2. Написать функциональную программу и доказать правильность по определению семантики функциональных программ. Найти наибольшую степень двойки, меньшую заданного числа  $x$ . Использовать  $s, <, =, +$ .

За решение каждой задачи выставляется максимум 5 баллов.

**Расчетно-графическая работа 3.** Темы: функциональные программы, сложность вычислений. Пример задания:

Написать функциональную программу и доказать ее правильность по индукции, написать структурированную программу и определить максимальное время и память в зависимости от длины входных данных. Найти, сколько чисел меньших  $x$  имеют в своей десятичной записи только цифры 0 и 1. Использовать  $s$  — прибавление 1,  $=, <, +, -, \times, :$  — целочисленное деление, % — остаток от деления.

За решение задачи выставляется максимум 15 баллов.

**Общая сумма** В сумме за все задачи выставляет не более 60 баллов.

За ответ на экзамене выставляется максимум 40 баллов.

## VII. Материально-техническая база, необходимая для осуществления образовательного процесса по дисциплине

### Для аудиторной работы

Наименование помещений	Материально-техническое оснащение помещений
Ауд. 304 (170002, Тверская обл., г. Тверь, пер. Садовый, д. 35)	Набор учебной мебели, экран, комплект аудиотехники (радиосистема, стационарный микрофон с настольным держателем, усилитель, микшер, акустическая система), проектор, ноутбук.

### Для самостоятельной работы

Наименование помещений	Материально-техническое оснащение помещений
Ауд. 201а (компьютерная лаборатория ПМиК) (170002, Тверская обл., г. Тверь, пер. Садовый, д. 35)	Набор учебной мебели, доска маркерная, компьютер, сервер (системный блок), концентратор сетевой.

## VIII. Сведения об обновлении рабочей программы дисциплины

№ п/п	Обновленный раздел рабочей программы дисциплины	Описание внесённых изменений	Дата и протокол заседания кафедры, утвердившего изменения