

Документ подписан простой электронной подписью

Информация о владельце:

ФИО: Максимов Алексей Борисович

Должность: директор департамента по образовательной политике

Дата подписания: 31.08.2019 14:44:49

Уникальный программный ключ:

8db180d1a3f02ac9e60521a567x44109c1800ab

МИНИСТЕРСТВО НАУКИ И ВЫСШЕГО ОБРАЗОВАНИЯ РОССИЙСКОЙ ФЕДЕРАЦИИ

Федеральное государственное бюджетное образовательное учреждение высшего образования

«МОСКОВСКИЙ ПОЛИТЕХНИЧЕСКИЙ УНИВЕРСИТЕТ»

УТВЕРЖДАЮ



Рабочая программа дисциплины

«Математическая логика и теория алгоритмов»

Направление подготовки:

09.03.01 Информатика и вычислительная техника

Образовательная программа (профиль):

«Программное обеспечение информационных систем»

Год начала обучения:

2019.

Уровень образования:

бакалавриат.

Квалификация (степень) выпускника:

Бакалавр.

Форма обучения:

заочная.

Москва, 2019

Программа дисциплины «Математическая логика и теория алгоритмов» составлена в соответствии с Федеральным государственным образовательным стандартом высшего образования по направлению подготовки бакалавров 09.03.01 «**Информатика и вычислительная техника**».

Программу составил:
доцент, к.ф.-м.н.



/Н. И. Царькова/

Программа утверждена на заседании кафедры «Прикладная информатика» 28 августа 2019 г., протокол № 1.

Заведующий кафедрой
доцент, к.э.н.



/С.В. Суворов/

1. Цели освоения дисциплины

Основные цели дисциплины «Математическая логика и теория алгоритмов»:

- знакомство студентов с основами их будущей специальности, систематизация математических знаний и навыков, полученных в школе, помощь в изучении профилирующих дисциплин первого и второго семестра;
- ознакомление с такими разделами математики как алгебра высказываний, теория множеств, теория предикатов, теория неориентированных и ориентированных графов, теория кодирования, которые являются основой многих других математических и специальных дисциплин.

Основные задачи дисциплины «Математическая логика и теория алгоритмов»:

- помощь в разрешении проблем в области математики и информатики, возникающих у студентов-первокурсников в связи с различными стилями обучения в школе и вузе.

2. Место дисциплины в структуре ООП бакалавриата

Дисциплина «Математическая логика и теория алгоритмов» относится к числу профессиональных учебных дисциплин обязательной части базового цикла основной образовательной программы бакалавриата.

Курс базируется на знаниях, полученных студентами в средней школе. Знания, навыки и умения, приобретенные в результате прохождения курса, необходимы при изучении таких дисциплин как: «Информатика» «Программирование» «Базы данных» «ЭВМ и периферийные устройства» «Защита информации» «Методы хранения и обработки информации» «Структуры и алгоритмы компьютерной обработки данных» «Теория вычислительных процессов и структур»

Знания, навыки и умения, приобретенные в результате прохождения курса, также могут быть востребованы студентами при подготовке выпускной квалификационной работы бакалавра.

3. Перечень планируемых результатов обучения по дисциплине (модулю), соотнесенные с планируемыми результатами освоения образовательной программы

В результате освоения дисциплины у обучающихся формируются следующие компетенции, и ими должны быть достигнуты следующие результаты обучения (как этап формирования соответствующих компетенций):

Код компетенции	В результате освоения образовательной программы обучающийся должен обладать	Перечень планируемых результатов обучения по дисциплине
ПК-1	Обладать способностью разрабатывать модели компонентов информационных систем, включая модели баз данных и модели интерфейсов «человек – электронно-вычислительная машина».	знать: понятия множества и отношения, сравнение скоростей роста функций; метод математической индукции; основы логики высказываний и исчисления предикатов; основные определения и понятия теория неориентированных и ориентированных графов; основы линейной алгебры и аналитической геометрии; базисные методы обработки информации; уметь: выполнять различные операции над

		<p>множествами, сравнивать мощности множеств и скорости роста функций; записывать утверждения в виде предикатов; применять метод математической индукции;</p> <p>владеть: навыками решения задач, связанных с понятием мощности множеств, способами оценки скоростей роста функций</p>
ОПК-2	Обладать способностью осваивать методики использования программных средств для решения практических задач.	<p>знать: понятия алгоритма и алгоритмического языка; способы построения программ; основные концепции объектно-ориентированного программирования и базовые конструкции языка Ruby;</p> <p>уметь: применять базисные методы обработки информации при построении программ; понимать и использовать на практике основные концепции объектно-ориентированного программирования и базовые конструкции языка Ruby;</p> <p>владеть: программами, предназначенными для символьных вычислений</p>

4. Структура и содержание дисциплины

Общая трудоемкость дисциплины составляет **3** зачетных единиц, т.е. **108** академических часа, из них **96** часов – самостоятельная работа студентов. Все они осваиваются обучающимися в третьем семестре

Форма контроля – экзамен

Содержание разделов дисциплины

Первый семестр

Введение

Характеристика знаний и навыков, которыми должен обладать бакалавр информатики и вычислительной техники после 4-х лет обучения. Обзор основных математических фактов, которые должен знать студент-первокурсник после школы. Систематизация необходимых математических знаний.

Избранные вопросы математики

Множества и операции над ними. Сравнение мощностей множеств. Сравнение скоростей роста функций. Высказывания и предикаты. Законы эквивалентности. Элементы линейной алгебры и аналитической геометрии. Метод математической индукции.

Избранные вопросы информатики

Базисные методы обработки информации. Грамматики. Регулярные выражения.

Избранные вопросы программирования

Понятия алгоритма и алгоритмического языка. Способы построения программ. Основные концепции объектно-ориентированного программирования. Базовые конструкции языка Ruby. Операции над строками и массивами.

Второй семестр

Элементы теории графов

Графы и терминология. Ориентированные графы, сети. Представление графов в ЭВМ. Орграфы и бинарные отношения. Раскраска графов. Алгоритмы раскрашивания.

Деревья

Ориентированные, упорядоченные и бинарные деревья. Представление деревьев в ЭВМ. Деревья сортировки. Кратчайший остов.

Теория кодирования

Алфавитное кодирование. Кодирование с минимальной избыточностью. Помехоустойчивое кодирование. Коды обнаружения и исправления ошибок. Коды Хемминга. Сжатие данных. Шифрование.

Структура и содержание дисциплины представлены в приложении 1 к рабочей программе.

5. Образовательные технологии

Методика преподавания дисциплины «Математическая логика и теория алгоритмов» и реализация компетентностного подхода в изложении и восприятии материала предусматривает использование следующих активных и интерактивных форм проведения групповых, индивидуальных, аудиторных занятий в сочетании с внеаудиторной работой с целью формирования и развития профессиональных навыков обучающихся:

- чтение лекций;
- проведение лабораторных работ;
- проведение практических занятий;
- проведение регулярных устных опросов.

Тематика практических занятий:

- «Графы и терминология. Смежность. Связность. Изоморфизм графов. Ориентированные графы, сети. Представление графов в ЭВМ. Матрица смежности. Матрица инцидентий. Списки смежности. Массив дуг. Обходы графов. Орграфы и бинарные отношения. Раскраска графов. Алгоритмы раскрашивания»;
- «Ориентированные, упорядоченные и бинарные деревья. Представление деревьев в ЭВМ. Деревья сортировки. Кратчайший остов. Схема алгоритма построения кратчайшего остова. Алгоритм Краскала»;
- «Алфавитное кодирование. Кодирование с минимальной избыточностью. Помехоустойчивое кодирование. Коды обнаружения и исправления ошибок. Коды Хемминга. Сжатие данных. Шифрование».

Названия лабораторных работ:

- «Язык математики. Систематизация необходимых математических знаний. Построение таблиц истинности. Вывод законов логических эквивалентностей»;

- «Множества и операции над ними. Сравнение мощностей множеств. Сравнение скоростей роста функций. Высказывания и предикаты. Законы эквивалентности. Элементы линейной алгебры и аналитической геометрии. Метод математической индукции»;
- «Базисные методы обработки информации. Преобразование итерационных алгоритмов в рекурсивные и рекурсивных в итерационные. Построение индуктивных функций. Грамматика и правила вывода. Построение регулярных выражений»;
- «Понятия алгоритма и алгоритмического языка. Способы построения программ. Основные концепции объектно-ориентированного программирования. Базовые конструкции языка Ruby. Операции над строками и массивами»;
- «Графы и терминология. Смежность. Связность. Изоморфизм графов. Ориентированные графы, сети. Представление графов в ЭВМ. Матрица смежности. Матрица инцидентий. Списки смежности. Массив дуг. Обходы графов. Орграфы и бинарные отношения. Раскраска графов. Алгоритмы раскрашивания»;
- «Ориентированные, упорядоченные и бинарные деревья. Представление деревьев в ЭВМ. Деревья сортировки. Кратчайший остов. Схема алгоритма построения кратчайшего остова. Алгоритм Краскала»;
- «Алфавитное кодирование. Кодирование с минимальной избыточностью. Помехоустойчивое кодирование. Коды обнаружения и исправления ошибок. Коды Хемминга. Сжатие данных. Шифрование».

Удельный вес занятий, проводимых в интерактивных формах, определен главной целью образовательной программы, особенностью контингента обучающихся и содержанием дисциплины «Математическая логика и теория алгоритмов» и в целом по дисциплине составляет 25% аудиторных занятий. Занятия лекционного типа составляют 35,7% от объема аудиторных занятий.

6. Оценочные средства для текущего контроля успеваемости, промежуточной аттестации по итогам освоения дисциплины и учебно-методическое обеспечение самостоятельной работы студентов

В процессе обучения используются следующие оценочные формы самостоятельной работы студентов, оценочные средства текущего контроля успеваемости и промежуточных аттестаций:

В первом семестре:

- проверка домашних заданий;
- проверка готовности студентов к проведению лабораторных и практических работ;
- проверка выполненных лабораторных и практических работ;
- проведение экзамена.

Во втором семестре:

- проверка домашних заданий;
- проверка готовности студентов к проведению лабораторных и практических работ;
- проверка выполненных лабораторных и практических работ;
- проведение экзамена.

Примерные вопросы к экзамену приведены в приложении 2.

6.1. Фонд оценочных средств для проведения промежуточной аттестации обучающихся по дисциплине (модулю)

6.1.1. Перечень компетенций с указанием этапов их формирования в процессе освоения образовательной программы

В результате освоения дисциплины (модуля) формируются следующие компетенции:

Код компетенции	В результате освоения образовательной программы обучающийся должен обладать
ПК-1	Обладать способностью разрабатывать модели компонентов информационных систем, включая модели баз данных и модели интерфейсов «человек – электронно-вычислительная машина».
ОПК-2	Способностью осваивать методики использования программных средств для решения практических задач.

В процессе освоения образовательной программы данные компетенции, в том числе их отдельные компоненты, формируются поэтапно в ходе освоения обучающимися дисциплин (модулей), практик в соответствии с учебным планом и календарным графиком учебного процесса.

6.1.2. Описание показателей и критериев оценивания компетенций, формируемых по итогам освоения дисциплины (модуля), описание шкал оценивания

Показателем оценивания компетенций на различных этапах их формирования является достижение обучающимися планируемых результатов обучения по дисциплине (модулю).

ПК-1 – обладать способностью разрабатывать модели компонентов информационных систем, включая модели баз данных и модели интерфейсов «человек – электронно-вычислительная машина»				
Показатель	Критерии оценивания			
	2	3	4	5
знать: понятия множества и отношения, сравнение скоростей роста функций; метод математической индукции; основы логики высказываний и исчисления предикатов; основные определения и понятия теории неориентированных и ориентированных графов; основы линейной алгебры и аналитической	Обучающийся не знает понятия множества и отношения, методы сравнения скоростей роста функций.	Обучающийся знает метод математической индукции; основы логики высказываний и исчисления предикатов.	Обучающийся знает основные определения и понятия теории неориентированных и ориентированных графов.	Обучающийся знает основы линейной алгебры и аналитической геометрии; базисные методы обработки информации.

геометрии; базисные методы обработки информации				
уметь: выполнять различные операции над множествами, сравнивать мощности множеств и скорости роста функций; записывать утверждения в виде предикатов; применять метод математической индукции	Обучающийся не умеет выполнять различные операции над множествами.	Обучающийся умеет выполнять различные операции над множествами.	Обучающийся умеет сравнивать мощности множеств и скорости роста функций.	Обучающийся умеет записывать утверждения в виде предикатов; применять метод математической индукции.
владеть: навыками решения задач, связанных с понятием мощности множеств, способами оценки скоростей роста функций	Обучающийся не владеет навыками решения задач, связанных с понятием мощности множеств.	Обучающийся владеет навыками решения задач, связанных с понятием мощности множеств.	Обучающийся владеет способами оценки скоростей роста функций.	Обучающийся владеет навыками решения задач, связанных с понятием мощности множеств, способами оценки скоростей роста функций.
ОПК-2 – обладать способностью осваивать методики использования программных средств для решения практических задач				
знать: понятия алгоритма и алгоритмического языка; способы построения программ; основные концепции объектно- ориентированного программирования и базовые конструкции языка Ruby	Обучающийся не знает понятия алгоритма и алгоритмического языка.	Обучающийся знает понятия алгоритма и алгоритмического языка.	Обучающийся знает способы построения программ.	Обучающийся знает основные концепции объектно- ориентированного программирования и базовые конструкции языка Ruby.
уметь: применять базисные методы обработки информации при построении программ; понимать и использовать на практике основные концепции объектно- ориентированного	Обучающийся не умеет применять базисные методы обработки информации при построении программ.	Обучающийся умеет применять базисные методы обработки информации при построении программ.	Обучающийся умеет понимать на практике основные концепции объектно- ориентированного программирования и базовые конструкции языка Ruby.	Обучающийся умеет использовать на практике основные концепции объектно- ориентированного программирования и базовые конструкции языка Ruby.

программирования и базовые конструкции языка Ruby				
владеть: программами, предназначенными для символьных вычислений	Обучающийся не владеет программами, предназначенными для символьных вычислений.	Обучающийся владеет приемами работы посредством интерфейса с программами, предназначенными для символьных вычислений.	Обучающийся владеет приемами работы посредством интерфейса с программами, предназначенными для символьных вычислений.	Обучающийся уверенно владеет программами, предназначенными для символьных вычислений.

Шкалы оценивания результатов промежуточной аттестации и их описание:

Формы промежуточной аттестации: экзамены

Промежуточная аттестация обучающихся в форме экзамена проводится по результатам выполнения всех видов учебной работы, предусмотренных учебным планом по данной дисциплине, при этом учитываются результаты текущего контроля успеваемости в течение семестра. Оценка степени достижения обучающимися планируемых результатов обучения по дисциплине проводится преподавателем, ведущим занятия по дисциплине методом экспертной оценки. По итогам промежуточной аттестации по дисциплине выставляется оценка «отлично», «хорошо», «удовлетворительно» или «неудовлетворительно».

К промежуточной аттестации допускаются только студенты, выполнившие все виды учебной работы, предусмотренные рабочей программой по дисциплине «Математическая логика и теория алгоритмов» (выполнили практические и лабораторные работы).

Шкала оценивания	Описание
Отлично	Выполнены все виды учебной работы, предусмотренные учебным планом. Студент демонстрирует соответствие знаний, умений, навыков приведенным в таблицах показателей, оперирует приобретенными знаниями, умениями, навыками, применяет их в ситуациях повышенной сложности. При этом могут быть допущены незначительные ошибки, неточности, затруднения при аналитических операциях, переносе знаний и умений на новые, нестандартные ситуации.
Хорошо	Выполнены все виды учебной работы, предусмотренные учебным планом. Студент в основном демонстрирует соответствие знаний, умений, навыков приведенным в таблицах показателей, оперирует приобретенными знаниями, умениями, навыками, применяет их в ситуациях повышенной сложности. При этом могут быть допущены некоторые ошибки, неточности, затруднения при аналитических операциях, переносе знаний и умений на новые, нестандартные ситуации.
Удовлетворительно	Студент демонстрирует удовлетворительное соответствие знаний, умений, навыков приведенным в таблицах показателей, допускаются умеренные ошибки, проявляется неполное наличие знаний, умений, навыков по ряду показателей, студент испытывает затруднения при оперировании знаниями и умениями при их переносе на новые ситуации.
Неудовлетворительно	Не выполнен один или более видов учебной работы, предусмотренных учебным планом. Студент демонстрирует неполное соответствие знаний, умений, навыков приведенным в таблицах показателей, допускаются

	значительные ошибки, проявляется отсутствие знаний, умений, навыков по ряду показателей, студент испытывает значительные затруднения при оперировании знаниями и умениями при их переносе на новые ситуации.
--	--

Фонд оценочных средств представлен приложении 2 к рабочей программе.

7. Учебно-методическое и информационное обеспечение дисциплины

Основная литература:

1. Чичкарев Е. А. Компьютерная математика с Maxima. [Электронный ресурс] – Электрон. дан. – Москва: Национальный Открытый Университет «ИНТУИТ», 2016. – 459 с. – Режим доступа: <http://www.knigafund.ru/books/176073/read#page1>

Дополнительная литература:

1. Аляев Ю. А., Тюрин С. Ф. Дискретная математика : практическая дискретная математика и математическая логика: учебное пособие. [Электронный ресурс] – Электрон. дан. – Москва: Финансы и статистика, 2012. – 383 с. – Режим доступа: <http://www.knigafund.ru/books/178656/read#page1>

Программное обеспечение и интернет-ресурсы:

1. Свободное программное обеспечение, входящее в базовую поставку ОС Linux: Браузер Mozilla Firefox, интерпретатор языка Ruby, текстовый редактор Atom, пакет символьной алгебры Maxima.
2. Офисные приложения LibreOffice для Linux (свободное ПО)
3. Офисные приложения Microsoft Office 2013(или ниже) - Microsoft Open License. Лицензия № 61984042

8. Материально-техническое обеспечение дисциплины

Компьютерные классы с ОС Linux в аудиториях: ав1201, ав1202, оснащенные: Компьютеры, столы, стулья, аудиторная доска, проектор. Рабочее место преподавателя: компьютер, стол, стул

9. Методические рекомендации для самостоятельной работы студентов

Изучение дисциплины «Математическая логика и теория алгоритмов» осуществляется в строгом соответствии с целевой установкой рабочей программы по дисциплине. При самостоятельной работе студентам рекомендуется в первую очередь прорабатывать лекционные материалы, дополняя их сведениями из тематических литературы и информационных ресурсов. Теоретические знания закрепляются посредством выполнения лабораторных работ и решения практических задач в рамках аудиторных занятий, к которым требуется своевременная самостоятельная подготовка. Для углубления получаемых знаний и выработки исследовательских навыков студенту предлагается выполнить ряд домашних заданий и изучить отдельные темы. Важным элементом освоения студентом дисциплины является его стремление к систематизации знаний, получаемых по всем видам данной дисциплины, а также выстраивание логических связей между данной дисциплиной и дисциплинами изученными ранее. При возникновении у студента вопросов локального характера по материалам дисциплины преподавателем дистанционно, с помощью современных средств телекоммуникации, оказывается консультационная помощь.

10. Методические указания для преподавателя

Проведение занятий по дисциплине «Математическая логика и теория алгоритмов» осуществляется в строгом соответствии с целевой установкой и в тесной взаимосвязи с учебным планом. Основой теоретической подготовки студентов являются лекции. При рассмотрении учебных материалов рекомендуется делать акцент на практические примеры, демонстрировать их реальную работу с помощью проектора.

В процессе самостоятельной работы студенты закрепляют и углубляют знания, полученные во время аудиторных занятий, дорабатывают конспекты лекций, готовятся к экзамену, а также самостоятельно изучают отдельные темы учебной программы.

Важным обстоятельством является привлечение внимания студентов к обсуждаемой проблеме, стимулирование интереса к ней и организация активного обсуждения, как структуры проблемы, так и составляющих ее наиболее актуальных тем. Для повышения эффективности проведения занятия требуется предварительная подготовка всех его участников. В этой связи рекомендуется заблаговременно (не менее, чем за неделю) оповестить студентов о теме занятия, дать перечень литературы по теме.

При проведении практического занятия преподаватель выполняет, в основном, функции ведущего – направляет студентов в правильное русло решения задач, рассматривает оптимальность предложенных решений, корректирует возможные ошибки.

Активная работа студента на практическом занятии учитывается при определении итоговой оценки его знаний по дисциплине на экзамене.

Самостоятельная работа по дисциплине «Математическая логика и теория алгоритмов» предполагает: выполнение студентами домашних заданий. Домашние задания являются, как правило, продолжением практических занятий и содействуют овладению практическими навыками по основным разделам дисциплины. Самостоятельная работа студентов предполагает изучение теоретического и практического материала по актуальным вопросам дисциплины. Рекомендуется самостоятельное изучение учебной и научной литературы, использование справочной литературы и др.

При выдаче заданий на самостоятельную работу используется дифференцированный подход к студентам. Перед выполнением студентами самостоятельной внеаудиторной работы преподаватель проводит инструктаж по выполнению задания, который включает: цель задания, его содержание, сроки выполнения, ориентировочный объем работы, основные требования к результатам работы, критерии оценки. В процессе инструктажа преподаватель предупреждает студентов о возможных типичных ошибках, встречающихся при выполнении задания. Инструктаж проводится преподавателем за счет объема времени, отведенного на изучение дисциплины.

Текущий контроль осуществляется на практических занятиях, промежуточный контроль осуществляется на экзамене в письменной или устной форме.

Самостоятельная работа осуществляется индивидуально. Контроль самостоятельной работы организуется в двух формах:

- самоконтроль и самооценка студента;
- контроль со стороны преподавателей (текущий и промежуточный).

Критериями оценки результатов самостоятельной работы студента являются:

- уровень освоения студентом учебного материала;
- умения студента использовать теоретические знания при выполнении практических задач;
- сформированность умений;
- оформление материала в соответствии с требованиями.

**МИНИСТЕРСТВО ОБРАЗОВАНИЯ И НАУКИ
РОССИЙСКОЙ ФЕДЕРАЦИИ**

Федеральное государственное бюджетное образовательное учреждение
высшего образования
«МОСКОВСКИЙ ПОЛИТЕХНИЧЕСКИЙ УНИВЕРСИТЕТ»

Направление подготовки:

09.03.01 «Информатика и вычислительная техника»

Профиль подготовки

« Программное обеспечение информационных систем»

Форма обучения: заочная

Кафедра: Прикладная информатика

ФОНД ОЦЕНОЧНЫХ СРЕДСТВ ПО ДИСЦИПЛИНЕ

«Математическая логика и теория алгоритмов»

Состав:

1. Паспорт фонда оценочных средств
2. Перечень оценочных средств
3. Оценочные средства

Составитель:

доцент, Н И Царькова.

Москва 2019 г.

**Паспорт фонда оценочных средств по дисциплине «Математическая логика и теория алгоритмов» по направлению подготовки
09.03.01 «Информатика и вычислительная техника» (бакалавр)**

«Математическая логика и теория алгоритмов»					
ФГОС ВО 09.03.01 «Информатика и вычислительная техника» (уровень бакалавриата)					
В процессе освоения данной дисциплины студент формирует и демонстрирует следующие профессиональные и общепрофессиональные компетенции :					
Компетенции		Перечень компонентов	Технология формирования компетенций	Форма оценочного средства	Степени уровней освоения компетенций
Индекс	Формулировка				
ПК-1	способность разрабатывать модели компонентов информационных систем, включая модели баз данных и модели интерфейсов «человек – электронно-вычислительная машина»	<p>знать: основные принципы функционального программирования; основы лямбда-исчисления; основы теории комбинаторов; методы реализации функциональных языков; теорию рекурсивных функций и лямбда-исчисление А.Черча; различия между энергичными и ленивыми языками функционального программирования; соответствие между функциональными и императивными программами;</p> <p>уметь: применять полученные знания для решения конкретных прикладных задач; ориентироваться в современных языках функционального программирования, их возможностях; обосновывать выбор</p>	лекции, лабораторные работы, практические занятия	экзамен (Экз)	<p>пороговый уровень: владеет методами и приемами программирования с использованием функциональных языков;</p> <p>базовый уровень: умеет применять полученные знания для решения конкретных прикладных задач;</p> <p>повышенный уровень: умеет ориентироваться в современных языках функционального программирования, их возможностях; обосновывать выбор языка функционального программирования для решения конкретных задач</p>

		<p>языка функционального программирования для решения конкретных задач; обосновывать выбор представления и обработки данных для решения поставленной задачи;</p> <p>владеть: методами и приемами программирования с использованием функциональных языков; навыками разработки рекурсивных функций</p>			
ОПК-2	Обладать способностью осваивать методики использования программных средств для решения практических задач.	<p>знать: понятия алгоритма и алгоритмического языка; способы построения программ; основные концепции объектно-ориентированного программирования и базовые конструкции языка Ruby;</p> <p>уметь: применять базисные методы обработки информации при построении программ; понимать и использовать на практике основные концепции объектно-ориентированного программирования и базовые конструкции языка Ruby;</p> <p>владеть: программами, предназначенными для символьных вычислений</p>	лекции, лабораторные работы, практические занятия	экзамен (Экз)	<p>пороговый уровень: владеет программами, предназначенными для символьных вычислений;</p> <p>базовый уровень: знает основные концепции объектно-ориентированного программирования и базовые конструкции языка Ruby;</p> <p>повышенный уровень: умеет понимать и использовать на практике основные концепции объектно-ориентированного программирования и базовые конструкции языка Ruby</p>

Перечень оценочных средств по дисциплине «Математическая логика и теория алгоритмов» по направлению подготовки 09.03.01 «Информатика и вычислительная техника» (бакалавр)

№ ОС	Наименование оценочного средства	Краткая характеристика оценочного средства	Представление оценочного средства в ФОС
1	Экзамен (Экз)	Средство промежуточной аттестации студента, проводится в письменно-устной форме.	Перечень вопросов по темам (разделам) дисциплины.

Оценочные средства для текущего контроля успеваемости, промежуточной аттестации по итогам освоения дисциплины и учебно-методическое обеспечение самостоятельной работы

Примерный перечень вопросов к экзамену по дисциплине «Математическая логика и теория алгоритмов» (ПК-1, ОПК-2):

1. Понятие множества. Основные операции над множествами, их свойства. Диаграммы Венна.
2. Понятие отображения. Основные свойства отображений.
3. Биекция. Мощность множества. Равномощные множества, счётные и несчётные множества.
4. Индуктивные умозаключения и их виды. Формальная аксиоматическая теория для арифметики натуральных чисел. Метод математической индукции.
5. Бинарные отношения, их свойства. Отношения частичного и полного порядка.
6. Сравнение асимптотического поведения функций.
7. Основные методы модуля Comparable и классов Numeric, Float и Integer.
8. Работа с регулярными выражениями (класс Regexp).
9. Опции интерпретатора Ruby. Однострочники.
10. Понятие неориентированного графа. Способы задания графа.
11. Матрицы смежности и инцидентности.
12. Деревья. Их свойства. Алгоритм Краскала построения дерева наименьшей длины.
13. Ориентированные деревья и их свойства. Бинарные деревья.
14. Эйлеров граф. Теорема Эйлера (критерии эйлеровости графа). Алгоритм нахождения эйлерова цикла в эйлеровом графе.
15. Теорема о числе маршрутов определенной длины в графе.
16. Сеть. Алгоритм Форда-Фалкерсона.
17. Кратчайший путь в орграфе. Алгоритм Дейкстры.
18. Остов минимального веса. Два алгоритма решения задачи.
19. Раскраски. Хроматический индекс и хроматическое число. Оценки. Редукция. Хроматический полином. Числа Стирлинга.
20. Понятие кодирование. Задачи теории кодирования и области ее применения. Алфавитное кодирование.
21. Сжатие по алгоритму Хаффмана.
22. Алгоритмы Фано и Хаффмена оптимального кодирования.
23. Понятие о шифросистемах с "открытым ключом", их возможностях и приложениях.
24. Шифр RSA. Алгоритм создания открытого и секретного ключей.
25. Множества. Формула количества подмножеств конечного множества.
26. Логическая символика. Предикаты.
27. Функция Эйлера. Мультипликативность функции Эйлера.
28. Малая теорема Ферма и проверка простоты числа.
29. Псевдопростые числа Ферма. Числа Кармайкла.
30. Функция Эйлера в RSA.
31. Вычисление обратного по умножению элемента по модулю с помощью функции Эйлера.

32. Китайская теорема об остатках.
33. Алгоритм Гарнера восстановления числа по наборам модулей и остатков.
34. Алгоритм Евклида. Расширенный алгоритм Евклида.
35. Коэффициенты Безу. Решение линейных диофантовых уравнений.
36. Поиск обратных элементов по модулю с помощью коэффициентов Безу.
37. Бинарный алгоритм вычисления НОД.
38. Системы счисления: позиционные, непозиционные и смешанные.
39. Сравнение по модулю. Операции со сравнениями.
40. Классы вычетов по модулю и их свойства.
41. Сравнения и признаки делимости.
42. Языки и грамматики.
43. Синтаксис языка предикатов. Расширение понятия предиката.
44. Семантика языка предикатов. Законы эквивалентности.
45. Понятие о вычислительной сложности алгоритма. O-символика.
46. Рекурсивные программы и математическая индукция.
47. Базисные схемы обработки информации. Рекурсия.
48. Базисные схемы обработки информации. Итерация.
49. Непрерывные (цепные) дроби. Разложение в цепную дробь.
50. Приближение вещественных чисел рациональными. Понятие о подходящей дроби.