

**МИНИСТЕРСТВО ТРАНСПОРТА РОССИЙСКОЙ ФЕДЕРАЦИИ  
ФЕДЕРАЛЬНОЕ ГОСУДАРСТВЕННОЕ БЮДЖЕТНОЕ ОБРАЗОВАТЕЛЬНОЕ  
УЧРЕЖДЕНИЕ ВЫСШЕГО ОБРАЗОВАНИЯ  
«РОССИЙСКИЙ УНИВЕРСИТЕТ ТРАНСПОРТА (МИИТ)»**

Кафедра «Экономическая информатика»

**АННОТАЦИЯ К РАБОЧЕЙ ПРОГРАММЕ ДИСЦИПЛИНЫ**

**«Теория сложности алгоритмов»**

Специальность:	<u>10.05.01 – Компьютерная безопасность</u>
Специализация:	<u>Информационная безопасность объектов информатизации на базе компьютерных систем</u>
Квалификация выпускника:	<u>Специалист по защите информации</u>
Форма обучения:	<u>очная</u>
Год начала подготовки	<u>2018</u>

## 1. Цели освоения учебной дисциплины

Целями освоения учебной дисциплины «Теория сложности алгоритмов» являются:

закладка математического фундамента как средства изучения окружающего мира для успешного освоения дисциплин естественнонаучного и профессионального циклов; получение студентами основ теоретических знаний и прикладных навыков применения математических методов и моделей; подготовка к использованию этих методов для разработки и принятия эффективных организационных и управленческих решений на транспорте; построение и оценка математических алгоритмов для решения задач; применение полученных теоретических знаний на практике; оценка сложности реализации алгоритмов решения ЭВМ; развитие логического мышления и повышение общего уровня культуры студентов.

## 2. Место учебной дисциплины в структуре ОП ВО

Учебная дисциплина "Теория сложности алгоритмов" относится к блоку 1 "Дисциплины (модули)" и входит в его базовую часть.

## 3. Планируемые результаты обучения по дисциплине (модулю), соотнесенные с планируемыми результатами освоения образовательной программы

Процесс изучения дисциплины направлен на формирование следующих компетенций:

ОПК-10	способностью к самостоятельному построению алгоритма, проведению его анализа и реализации в современных программных комплексах
--------	--

## 4. Общая трудоемкость дисциплины составляет

5 зачетных единиц (180 ак. ч.).

## 5. Образовательные технологии

Аудиторная работа сочетает лекции и практические занятия. Практические занятия проводятся в группах. Лекции проводятся в традиционной организационной форме, по типу управления познавательной деятельности являются классическо-лекционными (объяснительно-иллюстративными). Практические занятия организованы с использованием технологий развивающего обучения. Большинство практического курса выполняется в виде традиционных практических занятий (объяснительно-иллюстративное решение задач). Так же при обучении используются технологии, основанные на коллективных способах обучения, а также с использованием компьютерной тестирующей системы. Самостоятельная работа студента организована с использованием традиционных видов работы. К традиционным видам работы относятся отработка учебного материала и отработка отдельных тем по учебным пособиям, подготовка к текущему и промежуточному контролю. Оценка полученных знаний, умений и навыков основана на модульно-рейтинговой технологии. Весь курс разбит на разделы, представляющие собой логически завершенный объем учебной информации. Фонды оценочных средств освоенных компетенций включают как вопросы теоретического характера для оценки знаний, так и задания практического содержания (решение задач) для оценки умений и навыков. Теоретические знания проверяются путём применения таких организационных форм, как индивидуальные и групповые устные опросы, решение тестов с использованием компьютеров или на бумажных носителях..

## 6. Содержание дисциплины (модуля), структурированное по темам (разделам)

## РАЗДЕЛ 1

### Введение в сложность алгоритмов

#### 1. Детерминированная машина Тьюринга.

Схема машины Тьюринга: управляющее устройство, читающая/пишущая головка, лента. Определение ДМТ-программы. Примеры ДМТ-программ (программ для детерминированной машины Тьюринга).

#### 2. Задачи и алгоритмы.

Массовая и индивидуальная задача. Примеры. Определение алгоритма.

Сложность решения задачи в наихудшем и наилучшем случаях. Труднорешаемые задачи.

#### 3. Задачи распознавания, языки и кодирование.

Сведение любой массовой задачи к задаче распознавания. Соответствие между задачами распознавания и языками.

Язык, распознаваемой программой. Что значит, что программа решает задачу распознавания при кодировки.

## РАЗДЕЛ 2

### Класс языков и задач P

#### 4. M-ДМТ программа с полиномиальными временем работы

#### 5. Алгоритмы вычисления ( $X$ $n$ -ой степени) $\text{mod } m$ и $n!$

#### 6. Поиск в упорядоченном массиве.

#### 7. Сортировка. Сортировка слиянием.

#### 8. Полиномиальные алгоритмы для задач дискретной математики.

Минимальное остовное дерево. Кратчайший путь в графе. Алгоритм Прима.

9. Рекуррентные методы построения алгоритмов. Метод динамического программирования. Задача об оптимальном порядке умножения матриц. Алгоритм Карацубы для умножения чисел. Метод «разделяй и властвуй».

10. Алгоритмы решения логических задач. Задача об устойчивом бракосочетании. Задача об ханойских башнях. Задачи из дискретной математики.

Определение входной длины и вычисление количества операций, временной сложности.

тестирование, опрос

## РАЗДЕЛ 3

### Класс языков и задач NP

#### 11. Понятие полиномиальной проверяемости.

#### 12. Недетерминированная машина Тьюринга.

#### 13. M-НДМТ программа с полиномиальным временем работы.

#### 14. Алгоритмы с экспоненциальной временной сложностью.

#### 15. Соотношение классов P и NP.

Раздел 4 Класс NP-полных задач.

#### 16. Полиномиальная сводимость языков и её свойства.

Полиномиальная эквивалентность языков и задач.

#### 17. Теорема Кука и идея её доказательства.

#### 18. 6 основных NP-полных задач

Задачи KM, ГЦ, КЛИКА, ВЫП, 3-ВЫП, 3-С и их соотношения в системе сводимости.

19. Возможные решения проблемы  $P=NP$ ?

тестирование, опрос

Экзамен