

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



Рабочая программа дисциплины (модуля),
как компонент образовательной программы
высшего образования - программы специалитета
по специальности
23.05.03 Подвижной состав железных дорог,
утвержденной первым проректором РУТ (МИИТ)
Тимониным В.С.

РАБОЧАЯ ПРОГРАММА ДИСЦИПЛИНЫ (МОДУЛЯ)

Информатика и основы искусственного интеллекта

Специальность:	23.05.03 Подвижной состав железных дорог
Специализация:	Высокоскоростной наземный транспорт
Форма обучения:	Очная

Рабочая программа дисциплины (модуля) в виде
электронного документа выгружена из единой
корпоративной информационной системы управления
университетом и соответствует оригиналу

Простая электронная подпись, выданная РУТ (МИИТ)
ID подписи: 5214
Подписал: заведующий кафедрой Пудовиков Олег
Евгеньевич
Дата: 24.10.2025

1. Общие сведения о дисциплине (модуле).

Целями освоения учебной дисциплины "Информатика и основы искусственного интеллекта" являются:

- изучить структуру и общие свойства информации;
- изучить способы представления хранения, обработки и передачи информации с помощью технических средств;
- сформировать у обучающихся информационную культуру;
- создать основу для использования современных средств вычислительной техники и стандартных пакетов прикладных программ при изучении обучающимися профессиональных дисциплин.

Задачами освоения учебной дисциплины "Информатика" являются:

- освоение практических навыков алгоритмизации и программирования;
- освоение работы персональным компьютером на пользовательском уровне;
- освоение работы с типовыми программными средами.

2. Планируемые результаты обучения по дисциплине (модулю).

Перечень формируемых результатов освоения образовательной программы (компетенций) в результате обучения по дисциплине (модулю):

ОПК-2 - Способен понимать принципы работы современных информационных технологий и использовать их для решения задач профессиональной деятельности.

Обучение по дисциплине (модулю) предполагает, что по его результатам обучающийся будет:

Знать:

устройство персонального компьютера, периферийных устройств, назначение и принципы работы стандартных программных сред; состояние и направления развития вычислительной техники и программного обеспечения.

Уметь:

правильно оценивать возможности персонального компьютера, выбирать программные среды для решения задач профессиональной деятельности; работать на персональном компьютере, использовать стандартные и специализированные программные среды.

Владеть:

методами обработки информации для решения задач, оценки и представления полученных результатов с использованием возможностей современных технических средств

3. Объем дисциплины (модуля).

3.1. Общая трудоемкость дисциплины (модуля).

Общая трудоемкость дисциплины (модуля) составляет 7 з.е. (252 академических часа(ов)).

3.2. Объем дисциплины (модуля) в форме контактной работы обучающихся с педагогическими работниками и (или) лицами, привлекаемыми к реализации образовательной программы на иных условиях, при проведении учебных занятий:

Тип учебных занятий	Количество часов		
	Всего	Семестр	
		№1	№2
Контактная работа при проведении учебных занятий (всего):	160	80	80
В том числе:			
Занятия лекционного типа	64	32	32
Занятия семинарского типа	96	48	48

3.3. Объем дисциплины (модуля) в форме самостоятельной работы обучающихся, а также в форме контактной работы обучающихся с педагогическими работниками и (или) лицами, привлекаемыми к реализации образовательной программы на иных условиях, при проведении промежуточной аттестации составляет 92 академических часа (ов).

3.4. При обучении по индивидуальному учебному плану, в том числе при ускоренном обучении, объем дисциплины (модуля) может быть реализован полностью в форме самостоятельной работы обучающихся, а также в форме контактной работы обучающихся с педагогическими работниками и (или) лицами, привлекаемыми к реализации образовательной программы на иных условиях, при проведении промежуточной аттестации.

4. Содержание дисциплины (модуля).

4.1. Занятия лекционного типа.

№ п/п	Тематика лекционных занятий / краткое содержание
1	Алгоритмы Рассматриваемые вопросы: - структура и типы алгоритмов; - правила построения последовательности решения задачи
2	Графическое представление алгоритмов. Рассматриваемые вопросы: - символы графического представления алгоритмов; - правила формирования и представления блок-схем алгоритмов
3	Программная среда TurboBASIC. Интерфейс программной среды. Рассматриваемые вопросы: - режим написания и редактирования программы.
4	Алгоритмический язык BASIC. Операторы присваивания, правила применения операторов присваивания. Рассматриваемые вопросы: - типы переменных в BASIC; - операторы присваивания явно и неявно заданных исходных данных.
5	Язык программирования Python. Операторы присваивания, правила применения операторов присваивания. Рассматриваемые вопросы: - типы переменных в BASIC; - операторы присваивания; - форматы операторов присваивания.
6	Алгоритмический язык BASIC. Язык программирования Python. Операторы вывода результатов Рассматриваемые вопросы: - представление результатов на экране; - операторы вывода результатов
7	Алгоритмический язык BASIC. Язык программирования Python. Операторы условного перехода. Рассматриваемые вопросы: - реализация различных видов оператора IF
8	Алгоритмический язык BASIC. Язык программирования Python. Операторы циклических действий. Рассматриваемые вопросы: - реализация циклов с помощью операторов If, While, For; - вложенные циклы
9	Алгоритмический язык BASIC. Массивы чисел. Язык программирования Python. Массивы чисел. Списки. Рассматриваемые вопросы: - формирование массивов, способы ввода данных в массивы; - операции в массивах, вывод массивов.
10	Алгоритмический язык BASIC. Язык программирования Python. Графические операторы. Рассматриваемые вопросы: - графический вывод результатов.
11	Примеры решения типовых задач в BASIC. Рассматриваемые вопросы: - решение задач.

№ п/п	Тематика лекционных занятий / краткое содержание
12	Программная среда MathCad. Интерфейс программной среды. Рассматриваемые вопросы: - структура документа MathCad; - основные способы использования.
13	Программная среда MathCad. Сложные математические вычисления в MathCad. Рассматриваемые вопросы: - понятие и типы переменных в MathCad.
14	Программная среда MathCad. Программирование в MathCad. Реализация алгоритма условного перехода в MathCad. Циклические действия/ Рассматриваемые вопросы: - способы использования оператора If. - реализация в программной среде Mathcad алгоритмов циклических действий.
15	Программная среда MathCad. Вывод результатов в MathCad. Рассматриваемые вопросы: - численный и графический способы вывода.
16	Примеры решения типовых задач в MathCad. Рассматриваемые вопросы: - решение задач.

4.2. Занятия семинарского типа.

Лабораторные работы

№ п/п	Наименование лабораторных работ / краткое содержание
1	Алгоритмы. Рассматриваемые вопросы: - структура и типы алгоритмов; - правила построения последовательности решения задачи
2	Графическое представление алгоритмов Рассматриваемые вопросы: - символы графического представления алгоритмов; - правила формирования и представления блок-схем алгоритмов
3	Программная среда TurboBASIC. Интерфейс программной среды. Рассматриваемые вопросы: - режим написания и редактирования программы.
4	Алгоритмический язык BASIC. Операторы присваивания, правила применения операторов присваивания. Рассматриваемые вопросы: - типы переменных в BASIC; - операторы присваивания явно и неявно заданных исходных данных.
5	Язык программирования Python. Операторы присваивания, правила применения операторов присваивания. Рассматриваемые вопросы: - типы переменных в BASIC; - операторы присваивания; - форматы операторов присваивания.

№ п/п	Наименование лабораторных работ / краткое содержание
6	Алгоритмический язык BASIC. Язык программирования Python. Операторы вывода результатов Рассматриваемые вопросы: - представление результатов на экране; - операторы вывода результатов
7	Алгоритмический язык BASIC. Язык программирования Python. Операторы условного перехода. Рассматриваемые вопросы: - реализация различных видов оператора IF.
8	Алгоритмический язык BASIC. Язык программирования Python. Операторы циклических действий. Рассматриваемые вопросы: - реализация циклов с помощью операторов If, While, For; - вложенные циклы
9	Алгоритмический язык BASIC. Массивы чисел. Язык программирования Python. Массивы чисел. Списки. Рассматриваемые вопросы: - формирование массивов, способы ввода данных в массивы; - операции в массивах, вывод массивов.
10	Алгоритмический язык BASIC. Язык программирования Python. Графические операторы. Рассматриваемые вопросы: - графический вывод результатов.
11	Примеры решения типовых задач в BASIC. Рассматриваемые вопросы: - решение задач.
12	Программная среда MathCad. Интерфейс программной среды. Рассматриваемые вопросы: - структура документа MathCad; - основные способы использования.
13	Программная среда MathCad. Сложные математические вычисления в MathCad. Рассматриваемые вопросы: - понятие и типы переменных в MathCad.
14	Программная среда MathCad. Программирование в MathCad. Реализация алгоритма условного перехода в MathCad. Циклические действия/ Рассматриваемые вопросы: - способы использования оператора If. - реализация в программной среде Mathcad алгоритмов циклических действий.
15	Программная среда MathCad. Вывод результатов в MathCad. Рассматриваемые вопросы: - численный и графический способы вывода.
16	Примеры решения типовых задач в MathCad. Рассматриваемые вопросы: - решение задач.

4.3. Самостоятельная работа обучающихся.

№ п/п	Вид самостоятельной работы
1	Подготовка к лабораторным занятиям
2	Выполнение расчетно-графической работы.
3	Подготовка к промежуточной аттестации.
4	Подготовка к текущему контролю.

4.4. Примерный перечень тем расчетно-графических работ

1. – сложные математические вычисления: подкоренные выражения, с использованием тригонометрических функций,
 - реализация алгоритма разветвлённого типа (алгоритма условного перехода или алгоритма циклических действий):
 - работа с массивами чисел: одномерные и двумерные массивы, определение максимума и минимума,
 - использование графических операторов для представления результатов вычислений: одиночный график в PYTHON

2. – сложные математические вычисления: сложные дробные выражения;
 - реализация алгоритма разветвлённого типа (алгоритма условного перехода или алгоритма циклических действий): цикл по количеству шагов;
 - работа с массивами чисел: определение чётности/кратности чисел,
 - использование графических операторов для представления результатов вычислений: одиночный график в PYTHON

3. – сложные математические вычисления: сложные дробные выражения;
 - реализация алгоритма разветвлённого типа (алгоритма условного перехода или алгоритма циклических действий): цикл по количеству шагов;
 - работа с массивами чисел: преобразование массивов из одномерного в двумерный, преобразование массивов из двумерного в одномерный,
 - использование графических операторов для представления результатов вычислений: одиночный график в PYTHON

4. – сложные математические вычисления: сложные дробные выражения;

- реализация алгоритма разветвлённого типа (алгоритма условного перехода или алгоритма циклических действий): цикл по условию выполнения;
- работа с массивами чисел: преобразование массивов из одномерного в двумерный, преобразование массивов из двумерного в одномерный,
- использование графических операторов для представления результатов вычислений: несколько графиков в собственных осях в PYTHON

5 – сложные математические вычисления: с использованием тригонометрических функций;

- реализация алгоритма разветвлённого типа (алгоритма условного перехода или алгоритма циклических действий): цикл по условию выполнения;
- работа с массивами чисел: преобразование массивов из одномерного в двумерный, преобразование массивов из двумерного в одномерный,
- использование графических операторов для представления результатов вычислений: несколько графиков в собственных осях в PYTHON

6 – сложные математические вычисления: с использованием тригонометрических функций;

- реализация алгоритма разветвлённого типа (алгоритма условного перехода или алгоритма циклических действий): цикл по условию выполнения;
- работа с массивами чисел: одномерные и двумерные массивы, определение максимума и минимумай,
- использование графических операторов для представления результатов вычислений: несколько графиков в собственных осях в PYTHON

7 – сложные математические вычисления: с использованием тригонометрических функций;

- реализация алгоритма разветвлённого типа (алгоритма условного перехода или алгоритма циклических действий): цикл по количеству шагов;
- работа с массивами чисел: , преобразование массивов из двумерного в одномерный,
- использование графических операторов для представления результатов вычислений: несколько графиков в собственных осях в PYTHON

8 – сложные математические вычисления: с использованием тригонометрических функций;

- реализация алгоритма разветвлённого типа (алгоритма условного перехода или алгоритма циклических действий): цикл по количеству шагов;
- работа с массивами чисел: определение чётности/кратности чисел,
- использование графических операторов для представления результатов вычислений: несколько графиков в собственных осях в PYTHON

9 – сложные математические вычисления: с использованием тригонометрических функций;

- реализация алгоритма разветвлённого типа (алгоритма условного перехода или алгоритма циклических действий): строчный и блочный форматы условия;
- работа с массивами чисел: определение чётности/кратности чисел,
- использование графических операторов для представления результатов вычислений: несколько графиков в собственных осях в PYTHON

10 – сложные математические вычисления: с использованием тригонометрических функций;

- реализация алгоритма разветвлённого типа (алгоритма условного перехода или алгоритма циклических действий): строчный и блочный форматы условия;
- работа с массивами чисел: определение максимума и минимума,
- использование графических операторов для представления результатов вычислений: несколько графиков в собственных осях в PYTHON.

1. – сложные математические вычисления: подкоренные выражения, с использованием тригонометрических функций,

- реализация алгоритма разветвлённого типа (алгоритма условного перехода или алгоритма циклических действий):
- работа с массивами чисел: одномерные и двумерные массивы, определение максимума и минимума,
- использование графических операторов для представления результатов вычислений: одиночный график в MATHCAD

2. – сложные математические вычисления: сложные дробные выражения;

– реализация алгоритма разветвлённого типа (алгоритма условного перехода или алгоритма циклических действий): цикл по количеству шагов;

– работа с массивами чисел: определение чётности/кратности чисел,

– использование графических операторов для представления результатов вычислений: одиночный график в МАТСАД

3. – сложные математические вычисления: сложные дробные выражения;

– реализация алгоритма разветвлённого типа (алгоритма условного перехода или алгоритма циклических действий): цикл по количеству шагов;

– работа с массивами чисел: преобразование массивов из одномерного в двумерный, преобразование массивов из двумерного в одномерный,

– использование графических операторов для представления результатов вычислений: одиночный график в МАТСАД

4. – сложные математические вычисления: сложные дробные выражения;

– реализация алгоритма разветвлённого типа (алгоритма условного перехода или алгоритма циклических действий): цикл по условию выполнения;

– работа с массивами чисел: преобразование массивов из одномерного в двумерный, преобразование массивов из двумерного в одномерный,

– использование графических операторов для представления результатов вычислений: несколько графиков в собственных осях в МАТСАД

5 – сложные математические вычисления: с использованием тригонометрических функций;

– реализация алгоритма разветвлённого типа (алгоритма условного перехода или алгоритма циклических действий): цикл по условию выполнения;

– работа с массивами чисел: преобразование массивов из одномерного в двумерный, преобразование массивов из двумерного в одномерный,

– использование графических операторов для представления результатов вычислений: несколько графиков в собственных осях в МАТСАД

6 – сложные математические вычисления: с использованием тригонометрических функций;

– реализация алгоритма разветвлённого типа (алгоритма условного перехода или алгоритма циклических действий): цикл по условию выполнения;

– работа с массивами чисел: одномерные и двумерные массивы, определение максимума и минимумай,

– использование графических операторов для представления результатов вычислений: несколько графиков в собственных осях в MATHCAD

7 – сложные математические вычисления: с использованием тригонометрических функций;

– реализация алгоритма разветвлённого типа (алгоритма условного перехода или алгоритма циклических действий): цикл по количеству шагов;

– работа с массивами чисел: преобразование массивов из двумерного в одномерный,

– использование графических операторов для представления результатов вычислений: несколько графиков в собственных осях в MATHCAD

8 – сложные математические вычисления: с использованием тригонометрических функций;

– реализация алгоритма разветвлённого типа (алгоритма условного перехода или алгоритма циклических действий): цикл по количеству шагов;

– работа с массивами чисел: определение чётности/кратности чисел,

– использование графических операторов для представления результатов вычислений: несколько графиков в собственных осях в MATHCAD

9 – сложные математические вычисления: с использованием тригонометрических функций;

– реализация алгоритма разветвлённого типа (алгоритма условного перехода или алгоритма циклических действий): строчный и блочный форматы условия;

– работа с массивами чисел: определение чётности/кратности чисел,

– использование графических операторов для представления результатов вычислений: несколько графиков в собственных осях в MATHCAD

10 – сложные математические вычисления: с использованием тригонометрических функций;

– реализация алгоритма разветвлённого типа (алгоритма условного перехода или алгоритма циклических действий): строчный и блочный форматы условия;

– работа с массивами чисел: определение максимума и минимума,

– использование графических операторов для представления результатов вычислений: несколько графиков в собственных осях в MATHCAD.

5. Перечень изданий, которые рекомендуется использовать при освоении дисциплины (модуля).

№ п/п	Библиографическое описание	Место доступа
1	Рыбников, Е. К. Программирование в среде Turbo Basic и программном пакете Mathcad : учебное пособие по дисциплине «Информатика» / Е. К. Рыбников, Е. В. Сердобинцев, С. В. Володин. - Москва : РУТ (МИИТ), 2018. - 116 с	URL: https://znanium.com/catalog/product/1895306 (дата обращения: 27.04.2024)
2	Алексеев, А. П. Информатика 2015 : учебное пособие / А. П. Алексеев. — Москва : СОЛОН-Пресс, 2015. — 400 с. — ISBN 978-5-91359-158-6. — Текст : электронный // Лань : электронно-библиотечная система	URL: https://e.lanbook.com/book/64921 (дата обращения: 22.09.2025). — Режим доступа: для авториз. пользователей.

6. Перечень современных профессиональных баз данных и информационных справочных систем, которые могут использоваться при освоении дисциплины (модуля).

Научно-техническая библиотека РУТ (МИИТ) (<http://library.miit.ru>).

Сайт ОАО «РЖД» <http://rzd.ru/>.

Информационный портал Научная электронная библиотека eLIBRARY.RU (www.elibrary.ru)

Поисковые системы: Yandex, Mail.

7. Перечень лицензионного и свободно распространяемого программного обеспечения, в том числе отечественного производства, необходимого для освоения дисциплины (модуля).

Пакет офисных программ MSOffice.

Программная среда TurboBASIC.

Программная среда BASIC256.

Специализированная программа Mathcad.

8. Описание материально-технической базы, необходимой для осуществления образовательного процесса по дисциплине (модулю).

Рабочее место преподавателя: персональный компьютер с мультимедиа оборудованием, подключённый к сети Internet

Рабочие места студентов: персональные компьютеры, подключённые к сети интернет

Минимальные требования, предъявляемые к компьютерам: Pentium 4, ОЗУ 4 ГБ, HDD 100 ГБ, USB 2.0

9. Форма промежуточной аттестации:

Зачет в 1 семестре.

Экзамен во 2 семестре.

10. Оценочные материалы.

Оценочные материалы, применяемые при проведении промежуточной аттестации, разрабатываются в соответствии с локальным нормативным актом РУТ (МИИТ).

Авторы:

доцент, доцент, к.н. кафедры
«Электропоезда и локомотивы»

С.В. Володин

Согласовано:

Заведующий кафедрой ЭиЛ

О.Е. Пудовиков

Председатель учебно-методической
комиссии

С.В. Володин