



МИНИСТЕРСТВО НАУКИ И ВЫСШЕГО ОБРАЗОВАНИЯ РОССИЙСКОЙ
ФЕДЕРАЦИИ
Федеральное государственное автономное образовательное учреждение
высшего образования
«Дальневосточный федеральный университет»
(ДФУ)

ПОЛИТЕХНИЧЕСКИЙ ИНСТИТУТ (ШКОЛА)

«СОГЛАСОВАНО»


Руководитель ОП


(подпись) Бочарова А.А.
«29» января 2021 г. (Ф.И.О. рук. ОП)



«УТВЕРЖДАЮ»

Директор департамента компьютерно-
интегрированных производственных систем


(подпись) Змеу К.В.
«29» января 2021 г. (Ф.И.О. дир. деп.)

РАБОЧАЯ ПРОГРАММА ДИСЦИПЛИНЫ

Информационные технологии в прикладной механике

Направление подготовки: **15.04.03 Прикладная механика**

Магистерская программа Вычислительная механика и компьютерный инжиниринг

Форма подготовки (очная)

курс 1 семестр 1
лекции 18 час.
практические занятия 18 час.
лабораторные работы 18 час.
в том числе с использованием МАО лек. 12 /пр. 8 /лаб. 8 час.
всего часов аудиторной нагрузки 54 час.
в том числе с использованием МАО 16 час.
самостоятельная работа 54 час.
в том числе на подготовку к экзамену - 27 час.
контрольные работы (количество)
курсовая работа / курсовой проект не предусмотрены
зачет семестр
экзамен 1 семестр

Рабочая программа составлена в соответствии с требованиями Федерального государственного образовательного стандарта по направлению подготовки 15.04.03 Прикладная механика утвержденного приказом Министерства образования и науки РФ от 07.07.2015 №12-13-1282

Рабочая программа обсуждена на заседании отделения Машиностроения, морской техники и транспорта протокол № 5 от «29» января 2021г.
Директор отделения: к.т.н. доцент Гривиниченко М.В.

Составитель: к.т.н. доцент Г.П. Озерова

Цели и задачи освоения дисциплины

Учебная дисциплина «Информационные технологии в прикладной механике» предназначена для магистров 1 курса, входит в вариативную часть блока 1 «Дисциплины (модули)». Знания, умения и навыки, полученные после ее изучения, будут использоваться в различных дисциплинах, где требуется умение работать с компьютером и владение современными информационными технологиями, а также при написании курсовых работ и проектов, выпускной квалификационной работы.

Цель: формирование теоретических и практических навыков по разработке надежного, качественного программного обеспечения для задач прикладной механики с применением современных технологий программирования.

Задачи:

- Сформировать умение ставить задачу для решения ее на компьютере, а также реализовать ее средствами языков программирования.
- Изучить основы структурного программирования, типы данных и конструкции языка высокого уровня.
- Сформировать умение реализовывать задачи прикладной механики средствами языка программирования.
- Дать методику, позволяющую свободно изучать и применять новые языки программирования.

В результате изучения данной дисциплины у обучающихся формируются следующие общекультурные компетенции (элементы компетенций):

Код и формулировка компетенции	Этапы формирования компетенции	
ПК-6 Способность самостоятельно овладеть языками программирования и разрабатывать оригинальные пакеты прикладных программ и проводить с их помощью расчеты машин и приборов на динамику и прочность, устойчивость, надежность, трение и износ для специализированных задач прикладной механики	Знать	<ul style="list-style-type: none"> - этапы разработки пакетов прикладных программ - общие принципы разработки прикладных программ - синтаксис, особенности программирования и стандартные библиотеки выбранного языка программирования
	Уметь	<ul style="list-style-type: none"> - применять вычислительную технику и программные средства для решения практических задач - разрабатывать алгоритмы и выполнять их программную реализацию для решения типовых задач прикладной механики
	Владеть	<ul style="list-style-type: none"> - современными интегрированными средами разработки прикладных программ - навыками разработки пакетов прикладных программ на алгоритмических языках - навыками тестирования, отладки и верификации программ
ПК-10 Способность принимать непосредственное участие в учебной и учебно-методической работе кафедр и других учебных подразделений по профилю направления, участвовать в разработке программ учебных дисциплин и курсов	Знать	<ul style="list-style-type: none"> - современные подходы преподавания лекционных курсов, проведения практических учебных занятий, лабораторных работ; - платформы для организации online обучения и основные компоненты online курсов
	Уметь	<ul style="list-style-type: none"> - применять современные подходы для преподавания лекционных курсов, проведения практических занятий и лабораторных работ с использованием online технологий ; - создавать элементы учебных курсов на online платформах;
	Владеть	<ul style="list-style-type: none"> - методами создания элементов учебных курсов на online платформах с использованием технологии программирования

Для формирования вышеуказанных компетенций в рамках дисциплины «Информационные технологии в прикладной механике» применяются следующие методы активного/ интерактивного обучения: игровое проектирование, групповая консультация.

I. СТРУКТУРА И СОДЕРЖАНИЕ ТЕОРЕТИЧЕСКОЙ ЧАСТИ КУРСА

Раздел I. Основы вычислений (6 час.)

Тема 1. Этапы реализации решения задач средствами языка программирования (1 час.)

Алгоритм и программа. Этапы разработки программы: проектирование, реализация, отладка, тестирование. Проектирование: определение входных и выходных данных, формальная постановка задачи, определения функциональных возможностей. Реализация: описание алгоритма, выбор типов данных. Отладка: способы отладки программы, типичные ошибки. Тестирование: методы тестирования.

Тема 2. Основы программирования на языке высокого уровня (2 час.)

Синтаксис и семантика программы. Структура программы на языке Python. Понятие синтаксиса и семантики. Представление данных в языках программирования. Типы данных: int, float, bool. Конструкции структурного программирования. Операторы присваивания, ввода, вывода, условные операторы. Простейшие задачи: геометрические и алгебраические вычисления. Библиотека math. Форматный вывод.

Тема 3. Функции (1 час.)

Функции, их назначение, синтаксис и семантика. Пошаговая разработка программ. Области видимости. Локальные и глобальные переменные. Формальные и фактические параметры. Методология разработки программ «сверху вниз» и «снизу вверх». Вычисление функций. Обработка исключительных ситуаций.

Тема 4. Списки и методы их обработки (1 час.)

Описание и назначение списков. Конструктор списков. Ввод и вывод списков. Циклы, их назначение, синтаксис и семантика. Обработка списков с использованием циклов. Вычисление табулированных функций. Функции min, max, sum. Задачи на списки.

Тема 5. Вычисления с комплексными числами (1 час.)

Описание комплексного числа. Основные операции с комплексными числами. Задачи на списки на применение комплексных чисел из механики жидкости и газа.

Раздел II. Графика и визуализация (4 час.)

Тема 1. Построение графиков (2 час.)

Библиотека matplotlib. Область рисования, основные элементы области. Управление выводом осей, подписей, легенды. Линии, способы их описания. Основные характеристики линий. Задача о построение графиков функций на заданном интервале.

Тема 2. Рисование геометрических фигур (1 час.)

Основы растровой графики. Рисование в относительных координатах. Библиотека для рисования геометрических фигур. Основные характеристики фигур, управление ими. Создание изображения из геометрических фигур.

Тема 3. Построение графиков поверхностей (1 час.)

Методы трехмерной графики. Математическая постановка задачи о визуализации потенциального течения идеальной жидкости. Реализация задачи

Раздел III. Матрицы и векторы (4 час.)

Тема 1. Массивы(2 час.)

Библиотека numpy, понятие массива. Описание массивов и операции с ними. Основные функции массивов. Математические операции с матрицами: сложение, вычитание, умножение, вычисление обратной матрицы, определителя. Решение систем линейных уравнений.

Тема 2. Интерполяция функций (1 час.)

Понятие тренда. Основные функции описания трендов. Прогнозирование временных рядов с помощью трендов. Задача о построении трендов и прогнозе значений.

Тема 3. Решение задачи теплопроводности методом конечных автоматов (2 час.)

Основные понятия клеточных автоматов. Математическая постановка задачи. Разработка алгоритма. Реализация задач о нагревании стержня и пластины.

Раздел IV. Символьные вычисления (4 час.)

Тема 1. Основы символьных вычислений (2 час.)

Библиотека sympy. Понятие символьных переменных. Методы решения уравнений. Символьные преобразования.

Тема 2. Исследование функций (1 час.)

Построение функции на графике. Решение задачи исследования функции.

Тема 3. Решение дифференциальных уравнений и систем (1 час.)

Решение задачи Клеро. Решение системы дифференциальных уравнений.

II. СТРУКТУРА И СОДЕРЖАНИЕ ПРАКТИЧЕСКОЙ ЧАСТИ КУРСА И САМОСТОЯТЕЛЬНОЙ РАБОТЫ

Практические занятия(18час.)

Занятие 1 (2 час). Вычислительные задачи

1. Геометрические вычисления на плоскости и в пространстве
2. Вычислительные задачи механики

Занятие 2 (2 час). Комплексные числа

1. Калькулятор комплексных чисел
2. Задача о потенциальном течении идеальной жидкости

Занятие 3 (2 час). Построение графиков и поверхностей

1. Построение графиков функций на плоскости (простые, параметрические и пр.)
2. Построение поверхностей.
3. Визуализация потенциального течения идеальной жидкости

Занятие 4 (2 час). Векторы и матрицы

1. Матричные вычисления
2. Решение системы линейных уравнений

Занятие 5 (2 час). Клеточный автомат

1. Решение одномерной задачи теплопроводности

Занятие 6 (2 час). Клеточный автомат

1. Решение двумерной задачи теплопроводности

Занятие 7 (2 час). Прогнозирование с помощью тренда

1. Решение задач кинематики

Занятие 8 (2 час). Символьные вычисления

1. Решение уравнений
2. Символьные преобразования

Занятие 9 (2 час). Дифференциальные уравнения

1. Решение задачи Клеро
2. Решение системы дифференциальных уравнений

Лабораторные работы (18 час.)

Лабораторная работа 1. Система организации обучения BlackBoard, создание и размещение учебных элементов. (1 час.)

Лабораторная работа 2. Геометрические вычисления на плоскости. (1 час.)

Лабораторная работа 3. Задача о потенциальном течении идеальной жидкости (2 час.)

Лабораторная работа 4. Визуализация графиков функций. (2 час.)

Лабораторная работа 5. Построение поверхностей. Визуализация потенциального течения идеальной жидкости (2 час.)

Лабораторная работа 6. Операции с матрицами и векторами (2 час.)

Лабораторная работа 7. Одномерная задача теплопроводности, моделирование клеточным автоматом (2 час.)

Лабораторная работа 8. Двумерная задача теплопроводности, моделирование клеточным автоматом(2 час.)

Лабораторная работа 9. Символьные операции. Решение уравнений.(2 час.)

Лабораторная работа 10. Решение задачи Клеро (2 час)

Самостоятельная работа (54 часа)

План-график выполнения самостоятельной работы по дисциплине

№ п/п	Дата/сроки выполнения	Вид самостоятельной работы	Примерные нормы времени на выполнение	Форма контроля
1	1 -2 неделя семестра	Изучение теоретического материала в системе BlackBoard, основная литература [1-2].	3 часа	УО-1
2	3-4 неделя семестра	Выполнение модуля 1 онлайн курса « Основы программирования на языке Python в примерах и задачах »	3 часа	ПР-1
3	5-6 неделя семестра	Проверочные тесты в BlackBoard, модуль 1 Самостоятельные задания к лабораторным модуля	3 часа	ПР-1 ПР 6
4	7-8 неделя семестра	Выполнение модуля 2 онлайн курса « Основы программирования на языке Python в примерах и задачах »	3 часа	ПР-1
5	9-10 неделя семестра	Проверочные тесты в BlackBoard, модуль 2 Самостоятельные задания к лабораторным модуля	3 часа	ПР-1 ПР-6
6	11-12 неделя семестра	Выполнение модуля 3 онлайн курса « Основы программирования на языке Python в примерах и задачах »	3 часа	ПР-1
	13-14 неделя	Проверочные тесты в BlackBoard, модуль 3 Самостоятельные задания к лабораторным модуля	3 часа	ПР-1 ПР-6
8	15-16 неделя семестра	Выполнение модуля 3 онлайн курса « Основы программирования на языке Python в примерах и задачах »	3 часа	ПР-1
9	17-18 неделя семестра	Проверочные тесты в BlackBoard, модуль 4 Самостоятельные задания к лабораторным модуля	3 часа	ПР-1 ПР-6
10	Экзаменационная сессия	Подготовка к экзамену	27 час	Экзамен
Итого:			54 час.	

III. УЧЕБНО-МЕТОДИЧЕСКОЕ ОБЕСПЕЧЕНИЕ САМОСТОЯТЕЛЬНОЙ РАБОТЫ ОБУЧАЮЩИХСЯ

Устный опрос

Вопросы и задания приведены в разделе «VIII. ФОНДЫ ОЦЕНОЧНЫХ СРЕДСТВ». Для подготовки используется основная и дополнительная литература по дисциплине «Информационные технологии в прикладной механике», а также информация, размещенная в LMS BlackBoard.

Вопросы, возникающие в процессе подготовки, студент может задать преподавателю либо на консультациях, либо через специальное средство LMS BlackBoard.

Проверочные тесты

Тестирование осуществляется на итоговом занятии через систему BlackBoard. Примеры тестов для каждого раздела приведены в в разделе «VIII. ФОНДЫ ОЦЕНОЧНЫХ СРЕДСТВ».

Самостоятельные задания к лабораторным работам

Выполняются в соответствии с методическими указаниями к выполнению лабораторных работ, размещенных в системе BlackBoard. Выполненные работы отправляются преподавателю на проверку через LMSBlackBoard. Примеры лабораторных работ по каждой теме приведены в в разделе «VIII. ФОНДЫ ОЦЕНОЧНЫХ СРЕДСТВ».

IV. КОНТРОЛЬ ДОСТИЖЕНИЯ ЦЕЛЕЙ КУРСА

Контролируемые разделы / темы дисциплины	Коды и этапы формирования компетенций		Оценочные средства	
			текущий контроль	промежуточная аттестация
Модуль 1. Основы вычислений	ПК-10	знает	Устный опрос (УО-1) Проверочный тест (ПР-1)	Вопросы 1-7
		умеет	Лабораторная работа (ПР-6)	Задание 1-2
		владеет		
Модуль 2. Графика и визуализация	ПК-6	знает	Проверочный тест (ПР-1)	Вопросы 8-18
		умеет	Лабораторная работа (ПР-6)	Задание 3-4
		владеет		
Модуль 3. Матрицы и векторы	ПК-6	знает	Проверочный тест (ПР-1)	Вопросы 19-24
		умеет	Лабораторная работа (ПР-6)	Задание 5-6
		владеет		
Модуль 4.	ПК-6	знает	Проверочный тест (ПР-1)	Вопросы 25-26

Контролируемые разделы / темы дисциплины	Коды и этапы формирования компетенций	Оценочные средства	
		текущий контроль	промежуточная аттестация
Символьные вычисления			
	умеет	Лабораторная работа (ПР-6)	Задание 7-8
владеет			

V. СПИСОК УЧЕБНОЙ ЛИТЕРАТУРЫ И ИНФОРМАЦИОННО-МЕТОДИЧЕСКОЕ ОБЕСПЕЧЕНИЕ ДИСЦИПЛИНЫ

Основная литература

1. Эйдлина Г.М., Милорадов К.А. Delphi: программирование в примерах и задачах : практикум : учебное пособие / Г. М. Эйдлина,. – М.: Риор, : Инфра-М, 2013,113 с.
<http://lib.dvfu.ru:8080/lib/item?id=chamo:751710&theme=FEFU>
2. Златопольский Д.М. Основы программирования на языке Python [Электронный ресурс] : учебник / Д.М. Златопольский. — Электрон. дан. — Москва: ДМК Пресс, 2017. — 284 с. — Режим доступа: <https://e.lanbook.com/book/97359> .
3. Зайцев, М. Г. Программирование. Структурное программирование, подпрограммы, строки : учебное пособие / М. Г. Зайцев. — Новосибирск : Новосибирский государственный технический университет, 2016. — 103 с. : <https://www.iprbookshop.ru/91680.html>
4. Гуриков, С. Р. Основы алгоритмизации и программирования на Python : учебное пособие / С.Р. Гуриков. — Москва : ФОРУМ : ИНФРА-М, 2020. — 343 с. - ISBN 978-5-00091-487-8. - Текст : электронный. - URL: <https://znanium.com/catalog/product/1206074> (дата обращения: 15.05.2020)
5. Борзунов, С. В. Алгебра и геометрия с примерами на Python : учебное пособие для вузов / С. В. Борзунов, С. Д. Кургалин. — Санкт-Петербург : Лань, 2020. — 444 с. — ISBN 978-5-8114-5489-1. — Текст : электронный // Лань : электронно-библиотечная система. — URL: <https://e.lanbook.com/book/149336> (дата обращения: 15.05.2020)

Дополнительная литература

1. Жуков, Р. А. Язык программирования Python: практикум : учебное пособие / Р.А. Жуков. — Москва : ИНФРА-М, 2020. — 216 с. + Доп. материалы [Электронный ресурс]. — www.dx.doi.org/10.12737/textbook_5cb5ca35aaa7f5.89424805. - ISBN 978-5-16-014701-7 . - Текст : электронный. - URL: <https://znanium.com/catalog/product/1045700> (дата обращения: 15.12.2020).
2. Swaroop Chitlur. A Byte of Python — Режим доступа: <http://wombat.org.ua/AByteOfPython/AByteofPythonRussian-2.01.pdf>

Перечень ресурсов информационно-телекоммуникационной сети «Интернет»

Справочник по языку Python. – Режим доступа:

[самоучитель по языку Python](#)
[библиотека numpy, часть 1](#)
[библиотека numpy, часть 2](#)
[библиотека matplotlib](#)
[основные графические команды](#)
[модуль sympy](#)

Перечень информационных технологий и программного обеспечения

При осуществлении образовательного процесса студентами и профессорско-преподавательским составом используется следующее программное обеспечение:

1. Python, Anaconda.
2. Notepad++.
3. Программное обеспечение электронного ресурса сайта ДВФУ, включая ЭБС ДВФУ.

4. Материалы курса, размещенные в LMS BlackBoard, идентификатор:

FU50219-15.04.03-ИТvPM-01: Информационные технологии в прикладной механике.

При осуществлении образовательного процесса студентами и профессорско-преподавательским составом используются следующие информационно-справочные системы:

1. Научная электронная библиотека eLIBRARY.
2. Электронно-библиотечная система издательства «Лань».
3. Электронно-библиотечная система «IPRbooks».
4. Электронно-библиотечная система «Znanium»
5. Информационная система "ЕДИНОЕ ОКНО доступа к образовательным ресурсам".

VI. МЕТОДИЧЕСКИЕ УКАЗАНИЯ ПО ОСВОЕНИЮ ДИСЦИПЛИНЫ

1. Описание последовательности действий студента («сценарий изучения дисциплины»).

Для успешного изучения дисциплины «Информационные технологии в прикладной механике» студенту необходимо:

1. При подготовке к практическому и лекционному занятию повторить основные теоретические положения, изученные на предыдущем занятии, а также при необходимости прочитать теоретический материал, представленный в системе BlackBoard. Далее ознакомиться с новым теоретическим материалом (также через систему BlackBoard), вникнуть в суть изучаемой проблемы, подготовить вопросы.

2. На практическом занятии тщательно конспектировать теоретический материал, участвовать в обсуждении, задавать вопросы.

3. При подготовке к лабораторным работам на основе материалов, представленных в системе BlackBoard, сначала понять задание лабораторной работы, найти теоретический материал, необходимый для работы, изучить алгоритм реализации задания, сформулировать вопросы преподавателю.

4. При выполнении лабораторной работы сначала сформулировать и задать вопросы преподавателю по методике выполнения работы, затем выполнить задание «по образцу». Отправить преподавателю через систему BlackBoard выполненное задание на проверку, ответить на вопросы преподавателя. Прежде, чем приступать к выполнению самостоятельных заданий, обдумать алгоритм их реализации, сформулировать и задать вопросы преподавателю по сути заданий, спланировать их выполнение.

5. Самостоятельные задания лабораторной работы можно выполнять как на аудиторном занятии, так и самостоятельно во внеаудиторное время. При этом результат их реализации необходимо отправить преподавателю на проверку.

6. После каждого практического занятия для закрепления материала необходимо пройти проверочные тесты в системе BlackBoard.

7. В течение недели выбрать время для работы со специальной литературой в библиотеке и для занятий на компьютере.

8. Самостоятельную работу организовывать в соответствии с графиком выполнения самостоятельной работы, приведенном в приложении 1.

5. 2. Рекомендации по использованию материалов учебно-методического комплекса. Рекомендуется использовать методические указания и материалы по курсу [FU50219-15.04.03-ITvPM-01: Информационные технологии в прикладной механике](#). А также теоретические материалы онлайн курса «[Основы программирования на языке Python в примерах и задачах](#)» на платформе Stepik.

3. Рекомендации по работе с литературой. Теоретический материал курса становится более понятным, когда дополнительно прослушиванию лекций, к работе на практических занятиях, использованию учебно-методического комплекса, представленного в системе BlackBoard и на Stepik, изучаются и книги из списка основной и дополнительной литературы. Литературу по курсу можно изучать в библиотеке, брать книгу на дом или читать ее на компьютере (если это электронный ресурс). Полезно

использовать несколько учебников, однако желательно придерживаться рекомендации преподавателя по выбору книг по каждому разделу. Не рекомендуется «заучивать» материал, желательно добиться понимания изучаемой темы дисциплины, а затем использовать изученный материал для реализации заданий. Кроме того, очень полезно выявить тенденции развития технологий программирования, выделить для себя направления дальнейшего изучения материала, для достижения более продвинутого уровня изучения дисциплины.

4. Рекомендации по подготовке к экзамену. Успешная подготовка к экзамену включает, с одной стороны, добросовестную работу в течение семестра, выполнение всех заданий преподавателя, а с другой – правильная организация процесса непосредственной подготовки. При подготовке к экзамену необходимо освоить теорию: разобрать определения всех понятий, повторить приемы решения задач с использованием языков программирования. Затем рассмотреть примеры и самостоятельно реализовать задания из каждой темы. При этом, если задания формулируются студентом самостоятельно, – достигается более продвинутый уровень изучения дисциплины.

VII. МАТЕРИАЛЬНО-ТЕХНИЧЕСКОЕ ОБЕСПЕЧЕНИЕ ДИСЦИПЛИНЫ

Аудиторные занятия по дисциплине включают практические занятия, для проведения которых необходим компьютерный класс со следующим оборудованием:

- 15 моноблоков Lenovo C360G-i34164G500UDK с установленным стандартным программным обеспечением, доступом к сети Интернет.
- мультимедийный проектор OptimaEX542I – 1 шт;
- настенный экран;
- аудио усилитель QVC RMX 850 – 1 шт;
- колонки – 1 шт;
- ИБП – 1 шт;

– документ-камера.

В целях обеспечения специальных условий обучения инвалидов и лиц с ограниченными возможностями здоровья в ДВФУ все здания оборудованы пандусами, лифтами, подъемниками, специализированными местами, оснащенными туалетными комнатами, табличками информационно-навигационной поддержки.

VIII. ФОНДЫ ОЦЕНОЧНЫХ СРЕДСТВ

Шкала оценивания уровня сформированности компетенций

Примечание [A1]: Необходимо заполнить таблицу согласно указанным компетенциям.

Код и формулировка компетенции	Этапы формирования компетенции		критерии	показатели
ПК-6 Способность самостоятельно овладевать языками программирования и разрабатывать оригинальные пакеты прикладных программ и проводить с их помощью расчеты машин и приборов на динамику и прочность, устойчивость, надежность, трение и износ для специализированных задач прикладной механики	Знать	<ul style="list-style-type: none"> - этапы разработки пакетов прикладных программ - общие принципы разработки прикладных программ - синтаксис, особенности программирования и стандартные библиотеки выбранного языка программирования 	<ul style="list-style-type: none"> - знание основных этапов разработки пакетов прикладных программ; - знание общих принципов разработки прикладных программ; - знание синтаксиса, особенностей программирования и стандартные библиотеки выбранного языка программирования. 	<ul style="list-style-type: none"> - способность перечислить и раскрыть суть основных этапов разработки пакетов прикладных программ; - способность перечислить и раскрыть суть общих принципов разработки прикладных программ; - способность воспроизвести и объяснить с требуемой степенью научной точности и полноты основные особенности программирования на выбранном языке, перечислить библиотеки, описать область их применения.
		<ul style="list-style-type: none"> - применять вычислительную технику и программные средства для решения практических задач - разрабатывать алгоритмы и выполнять их программную реализацию для решения типовых задач прикладной механики 	<ul style="list-style-type: none"> - умение использовать вычислительную технику и программные средства для решения практических задач; - умение разрабатывать алгоритмы и выполнять их программную реализацию для решения типовых задач прикладной механики. 	<ul style="list-style-type: none"> - способность применять вычислительную технику и программные средства для решения практических задач; - способность формализовать задачу, разработать алгоритм и выполнить его программную реализацию для решения типовых задач прикладной механики.
		<ul style="list-style-type: none"> - современными интегрированными средами разработки прикладных программ 	<ul style="list-style-type: none"> - современными интегрированными средами разработки прикладных программ; 	<ul style="list-style-type: none"> - способность свободно использовать интегрированные среды разработки прикладных программ;

Код и формулировка компетенции	Этапы формирования компетенции		критерии	показатели
		<ul style="list-style-type: none"> - навыками разработки пакетов прикладных программ на алгоритмических языках - навыками тестирования, отладки и верификации программ 	<ul style="list-style-type: none"> - владение навыками разработки пакетов прикладных программ на алгоритмических языках; - владение навыками тестирования, отладки и верификации программ. 	<ul style="list-style-type: none"> - способность разработать программу на выбранном алгоритмическом языке для решения задачи прикладной механики; - способность выполнить отладку, тестирование и верификацию разработанного программного продукта.
ПК-10 Способность принимать непосредственное участие в учебной и учебно-методической работе кафедр и других учебных подразделений по профилю направления, участвовать в разработке программ учебных дисциплин и курсов	Знать	<ul style="list-style-type: none"> - современные подходы преподавания лекционных курсов, проведения практических учебных занятий, лабораторных работ; - платформы для организации online обучения и основные компоненты online курсов. 	<ul style="list-style-type: none"> - знание современных подходов преподавания лекционных курсов, проведения практических учебных занятий, лабораторных работ; - знание об основных платформах для организации online обучения; - знание способов представления компонент курсов. 	<ul style="list-style-type: none"> - способность описать современных подходов преподавания лекционных курсов, проведения практических учебных занятий, лабораторных работ; - способность перечислить и раскрыть особенности основных платформ для организации online обучения; - способность перечислить и раскрыть назначение основных компонентов online курсов.
	Уметь	<ul style="list-style-type: none"> - применять современные подходы для преподавания лекционных курсов, проведения практических занятий и лабораторных работ с использованием online технологий ; - создавать элементы учебных курсов на online платформах 	<ul style="list-style-type: none"> - умение использовать современные подходы для преподавания лекционных курсов, проведения практических занятий и лабораторных работ с применением online технологий; - умение создавать элементы учебных курсов на online платформах. 	<ul style="list-style-type: none"> - способность использовать современные подходы для преподавания лекционных курсов, проведения практических занятий и лабораторных работ с применением online технологий; - способность создавать элементы учебных курсов на online платформах.
	Владеть	<ul style="list-style-type: none"> методами создания элементов учебных курсов на online платформах с использованием технологии программирования 	<ul style="list-style-type: none"> - владение методами создания элементов учебных курсов на online платформах с использованием технологии программирования. 	<ul style="list-style-type: none"> - способность создавать элементы учебных курсов на online платформах с использованием технологии программирования.

Методические рекомендации, определяющие процедуры оценивания результатов освоения дисциплины

Оценочные средства для промежуточной аттестации

Перечень типовых вопросов к экзамену

1. Понятие алгоритма и программы.
2. Структура программы.
3. Целые и вещественные типы данных. Операции над ними.
4. Операторы присваивания, ввода и вывода.
5. Условный оператор.
6. Модули для математических расчетов и комплексных чисел
7. Описание и операции с комплексными числами
8. Форматный вывод информации
9. Списки.
10. Операции со списками.
11. Модуль работы со списками
12. Операторы повторения
13. Типовые операции со списками
14. Алгоритмы статистической обработки данных.
15. Построение диаграмм и графиков
16. Модуль работы с графическими объектами
17. Построение поверхностей
18. Визуализация динамических процессов
19. Матрицы и вектора
20. Модуль для работы с матрицами и векторами
21. Матричные вычисления
22. Решение систем линейных уравнений
23. Реализация одномерного клеточного автомата
24. Реализация двумерного клеточного автомата
25. Символьные преобразования.
26. Решение уравнений приближенными методами
27. Решение уравнений в символьном виде

28. Решение дифференциальных уравнений в символьном виде

29. Решение систем дифференциальных уравнений

Типовые задания к экзамену

1. Даны координаты вершины куба. Найти все остальные вершины, если считать, что диагональ куба параллельна оси ОХ.

2. Шарик бросают с высоты h с нулевой начальной скоростью. Посчитать, через сколько секунд шарик упадет на землю. Построить график зависимости высоты шарика над землей от времени.

3. Построить график следующей функции:

$$F(x) = \sin(0,3x) \cdot \cos(0,2x) \cdot \sin^2(x), \quad x \in [0^\circ; 360^\circ].$$

4. Построить график следующей функции:

$$F(x, y) = x^2 \cdot y - x \cdot y^2, \quad x \in [-2; 2], \quad y \in [-2; 2].$$

5. Вычислить следующее матричное выражение (размер матриц A и C – n на n , размер вектора B – n , матрицы и вектора заполнить случайными числами от 1 до 6):

$$(4 \cdot A^2 \cdot C - C^{-1} \cdot A^{-1}) \cdot B.$$

6. Решить следующую систему линейных уравнений (с помощью матричного выражения):

$$\begin{cases} 2x_1 - x_2 - x_3 + 3x_4 = 1, \\ 4x_1 + 5x_2 - x_3 + x_4 = 5, \\ x_1 - 3x_2 - x_3 - x_4 = 9, \\ 2x_1 - x_2 + 2x_3 - 12x_4 = 2. \end{cases}$$

7. Найти корни уравнения $f(x) = 0$ на заданном интервале приближенными методами, предварительно построить график:

$$F(x) = 0,1x^3 - 3 \cdot x + 3, \quad x \in [-2; 2].$$

8. В символьном виде решить следующее дифференциальное уравнение:

$$y = x \cdot y' + (y')^2.$$

Образец экзаменационного билета

1. Реализация двумерного клеточного автомата.

2. Решить следующую систему линейных уравнений:

$$\begin{cases} 2x_1 - x_2 - x_3 + 3x_4 = 1, \\ 4x_1 + 5x_2 - x_3 + x_4 = 5, \\ x_1 - 3x_2 - x_3 - x_4 = 9, \\ 2x_1 - x_2 + 2x_3 - 12x_4 = 2. \end{cases}$$

3. В символьном виде решить следующее дифференциальное уравнение:

$$y = x \cdot y' + (y')^2.$$

Оценочные средства для текущей аттестации

Примечание [ТЕН2]: Комплекты оценочных средств для текущей аттестации, примерный перечень и краткая характеристика приведена в Приложении 1, форма представления в Приложениях 6-14 вышеприведенного документа по ФОС.

Вопросы для устного опроса

1. Типы данных.
2. Переменные.
3. Числовые типы данных.
4. Операции над числовыми типами данных.
5. Строки.
6. Вывод данных.
7. Ввод данных.
8. Форматированный ввод/вывод.
9. Списки.
10. Выражения в списках.
11. Операции сравнения для списков.
12. Диапазоны.
13. Оператор if. Особенности операторов сравнения.
14. Операторы цикла. Оператор for. Оператор while. Завершение цикла.
15. Определение функции.
16. Передача параметров.
17. Основные онлайн платформы для организации обучения
18. Структура курса на онлайн платформе
19. Типы учебных компонент
20. Типы контрольно-измерительных заданий
21. Способы занесения заданий на онлайн платформы

Задания для проверочных тестов

Каждому студенту формируется индивидуальный тест, в который входит 2 - 4 задания, выбранные случайным образом из общего списка. Каждое задание включает список вопросов, на которые должен ответить студент. Для реализации теста используется задание с многократным вводом текста. Пример тестового задания (рис. 1):

Даны комплексные числа:
 $z1 = -8.0 - i \cdot 7.7$
 $z2 = 7.48 - i \cdot 5.44$
 $z3 = 1.041 + i \cdot 0.9 \cdot \pi$

Выполните следующие задания (вещественные числа округлите до 2 знаков после запятой, в качестве разделителя дробной и целой части используйте точку, числа вводите так, как они отображаются в консоли, пробелы НЕ ИСПОЛЬЗУЙТЕ):

1. Найдите аргумент числа $z2$:
2. Вычислите выражение:
 $(2 \cdot z1 + 4 \cdot z3) \cdot z2^2$
Значения действительной и мнимой части результата занесите в соответствующие поля:
result = + i
3. Преобразуйте результат вычисления выражения к потенциальной форме. Заполните поля:
 $\cdot e^{i \cdot$
4. Вычислите значение комплексной функции от аргумента $z3$:
 $f(z) = e^{0.2z} \cos(2z)$
в поле занесите действительную часть результата:

Рис. 1. Пример проверочного теста

Лабораторные работы по дисциплине «Информационные технологии»

Структура каждой лабораторной работы следующая:

1. Описание реализации нескольких заданий в качестве образца рассматриваемой технологии.

2. Задания для самостоятельной работы.

Выполненные лабораторные работы необходимо защитить, но предварительно они должны быть отправлены преподавателю на проверку через систему BlackBoard.

Типовая лабораторная работа «Вычисления с комплексными числами и функциями»

Задание 1. Реализовать калькулятор комплексных чисел, который для двух чисел выводит их сумму, разность, произведение и частное.

Порядок выполнения лабораторной работы

1. Ввести действительную и мнимую часть первого числа:

```
print('Введите первое комплексное число:')
z1_real = float(input('действительная часть: '))
z1_imag = float(input('мнимая часть: '))
```

2. Ввести действительную и мнимую часть второго числа:

```
print('Введите второе комплексное число:')
z2_real = float(input('действительная часть: '))
z2_imag = float(input('мнимая часть: '))
```

3. Создать комплексные числа **z1** и **z2**, действительная и мнимая часть которых уже введена.

Для создания комплексного числа используется функция:

```
complex(a, b)
```

где a , b – вещественные числа.

Например,

```
complex(4, 3)
```

создает комплексное число $4+3j$ (в Python мнимая единица обозначается как j).

Создание чисел **z1** и **z2**:

```
z1 = complex(z1_real, z1_imag)
```

```
z2 = complex(z2_real, z2_imag)
```

```
print(z1)
```

```
print(z2)
```

Результат:

Введите первое комплексное число:

действительная часть: 3.1

мнимая часть: -4.7

Введите второе комплексное число:

действительная часть: 8.12

мнимая часть: 2.7

```
z1 = (3.1-4.7j)
```

```
z2 = (8.12+2.7j)
```

4. Вычислить сумму, разность, произведение и частное комплексных чисел.

Для комплексных чисел допустимы операции сложения (+), вычитания (-), умножения (*), деления (/) и возведение в степень (**). Результат операции – комплексное число.

Фрагмент программы, выполняющей арифметические операции с числами **z1** и **z2**:

```
result = z1 + z2
```

```
print('z1 + z2 =', result)
```

```
result = z1 - z2
```

```
print('z1 - z2 =', result)
```

```

result = z1 * z2
print('z1 * z2 =', result)

result = z1 / z2
print('z1 / z2 =', result)

```

Результат (с входными данными из предыдущего пункта):

```

z1 + z2 = (11.219999999999999-2j)
z1 - z2 = (-5.02-7.4j)
z1 * z2 = (37.861999999999995-29.793999999999993j)
z1 / z2 = (0.1704623049147552-0.6354985496637734j)

```

6. Вывести мнимую и действительную часть с 2 знаками после запятой.

Результат некоторых арифметических операции – комплексное число, действительная и мнимая часть которых – вещественное число с большим количеством знаков после запятой. Чтобы вывести такие числа в более удобном для восприятия виде необходимо выделить действительную и мнимую часть, а затем округлить их.

Для выделения действительной части используется метод **real**, а для выделения мнимой – метод **imag**. Например:

```

z = 3-4j
z_real = z.real # результат 3
z_imag = z.imag # результат -4

```

Вывод результата математической операции сложения с округленными до двух знаков значениями мнимой и действительной части:

```

print('z1 + z2 =', round(result.real, 2), '+', round(result.imag, 2), 'j')

```

Самостоятельно исправьте вывод остальных вычисленных значений.

Результат:

```

z1 + z2 = 11.22 + -2.0 j
z1 - z2 = -5.02 + -7.4 j
z1 * z2 = 37.86 + -29.79 j
z1 / z2 = 0.17 + -0.64 j

```

7. При выполнении операции деления может возникнуть ошибка деления на 0 в том случае, если и действительная, и мнимая части числа **z2** равны 0. Самостоятельно выполните эту проверку. Результат:

Верные входные данные:

Введите первое комплексное число:
действительная часть: -5.325
мнимая часть: 0.18
Введите второе комплексное число:
действительная часть: 4.209
мнимая часть: -6.089
z1 = (-5.325+0.18j)
z2 = (4.209-6.089j)
z1 + z2 = -1.12 + -5.91 j
z1 - z2 = -9.53 + 6.27 j
z1 * z2 = -21.32 + 33.18 j
z1 / z2 = -0.43 + -0.58 j

Неверные входные данные:

Введите первое комплексное число:
действительная часть: 3
мнимая часть: 5
Введите второе комплексное число:
действительная часть: 0
мнимая часть: 0
z1 = (3+5j)
z2 = 0j
z1 + z2 = 3.0 + 5.0 j
z1 - z2 = 3.0 + 5.0 j
z1 * z2 = 0.0 + 0.0 j
Ошибка! Деление на ноль.

Задание 2. Вычислить значения комплексной функции в заданных точках:

$$f(z) = \frac{2 \cdot \pi}{i} - \operatorname{arccot}(z) + z^3,$$

если $z = 2.045, -i \cdot 4.67, 3 - i \cdot \frac{\pi}{2}, \sqrt{\pi} + i \cdot 1.2$

Порядок выполнения лабораторной работы

1. Подключить модуль `cmath`:

```
import cmath
```

В таблице 1 приведены некоторые функции, реализованные в модуле `cmath`, z – комплексное число:

Таблица 1

Функция	Описание
<code>cmath.exp(z)</code>	e^z .
<code>cmath.log(z, [base])</code>	логарифм z по основанию <code>base</code> . Если <code>base</code> не указан, вычисляется натуральный логарифм.
<code>cmath.log10(z)</code>	логарифм z по основанию 10.
<code>cmath.log2(z)</code>	логарифм z по основанию 2.
<code>cmath.sqrt(z)</code>	квадратный корень из z .
<code>cmath.acos(z)</code>	арккосинус z .
<code>cmath.asin(z)</code>	арксинус z .
<code>cmath.atan(z)</code>	арктангенс z .
<code>cmath.cos(z)</code>	косинус z .
<code>cmath.sin(z)</code>	синус z .
<code>cmath.tan(z)</code>	тангенс z .
<code>cmath.pi</code>	$\pi = 3,1415926\dots$
<code>cmath.e</code>	$e = 2,718281\dots$

Для вычисления функций $\cot(z)$ и $\operatorname{arccot}(z)$ можно использовать формулы:

$$\cot(z) = \frac{1}{\tan(z)}, \quad \operatorname{arccot}(z) = \frac{\pi}{2} - \arctan(z).$$

2. Создать функцию $f(z)$:

```
def f(z):
    result = 2 * cmath.pi / 1j - (cmath.pi / 2 - cmath.atan(z)) + z ** 3
    return result
```

Пояснение. Комплексные числа можно задавать в виде $a + bj$, если действительная a и мнимая часть b – числовые константы.

3. Вычислить значение функции от аргумента $z = 2.045$:

```
z = 2.045
f_z = f(z)

print('f(', z, ') = ', f_z)
```

Результат:

```
f( 2.045 ) = (8.097434150113479-6.283185307179586j)
```

4. **Самостоятельно** вычислить значения функции от остальных аргументов, вывести значения действительной и мнимой части аргументов и результатов с 3 знаками после запятой. Результат:

```
f( 2.045 + 0.0 j ) = 8.097 + -6.283 j
f( -0.0 + -4.67 j ) = -3.142 + 95.347 j
f( 3.0 + -1.571 j ) = 4.533 + -44.948 j
f( 1.772 + 1.2 j ) = -2.479 + 3.528 j
```

Задание 3. Опишите функцию комплексного потенциала для потенциального потока вида «Источник», если расход жидкости равен $8 \cdot \pi$, а источник находится в точке на

плоскости с координатами (-3, 2). Функцию задайте в алгебраической и потенциальной форме. Получите функцию тока и потенциальную функцию, а также вычислите их значения в точках (-5, 3) и (-3, 2).

Математическая постановка задачи:

Комплексный потенциал потока идеальной жидкости с источником в точке (x_0, y_0) , имеет вид:

$$W(z) = \frac{Q}{2 \cdot \pi} \ln(z), \quad z = (x - x_0) + i \cdot (y - y_0)$$

Переход от алгебраической формы записи комплексного числа к потенциальной осуществляется по формуле:

$$z = |z| \cdot e^{i \cdot \text{Arg}(z)}$$

где

$|z|$ – модуль комплексного числа,

$\text{Arg}(z)$ – аргумент комплексного числа.

Таким образом, комплексный потенциал записывается в виде:

$$W(z) = \frac{Q}{2 \cdot \pi} \ln(|z| \cdot e^{i \cdot \text{Arg}(z)})$$

Потенциальная функция:

$$w_x(z) = \text{Real}(W(z)),$$

Функция тока:

$$w_y(z) = \text{Imag}(W(z)).$$

Порядок выполнения лабораторной работы

1. Подключить модуль **cmath**:

```
import cmath
```

В таблице 2 приведены некоторые функции, реализованные в модуле **cmath**, **z** – комплексное число ($z = x + i \cdot y$):

Таблица 2

Функция	Описание
<code>abs(z)</code>	модуль комплексного числа (пишется без префикса, так как не принадлежит модулю cmath), эквивалентно <code>math.sqrt(x ** 2 + y ** 2)</code>
<code>cmath.phase(z)</code>	аргумент комплексного числа (результат лежит в промежутке $[-\pi, \pi]$), эквивалентно <code>math.atan2(y, x)</code> .
<code>cmath.isfinite(z)</code>	True , если действительная и мнимая части конечны
<code>cmath.isinf(z)</code>	True , если либо действительная, либо мнимая часть бесконечна.
<code>cmath.isnan(z)</code>	True , если либо действительная, либо мнимая часть NaN .

2. Создать функцию **W_a(x, y)** для описания комплексного потенциала в алгебраической форме:

```
def W_a(x, y):
    # координаты источника
    x0 = -3
    y0 = 2
    # расход жидкости
    Q = 8 * cmath.pi
    z = complex(x - x0, y - y0)
    result = Q / (2 * cmath.pi) * cmath.log(z)
    return result
```

3. Вычислить значение этой функции в точке (5, 9) и вывести результат:

```
result = W_a(-5, 3)
print('W_a(-5, 3) = ', round(result.real, 3), '+', round(result.imag, 3), 'j')
```

Результат:


```
W_a(-5, 3) = 3.219 + 10.712 j
```

4. Создать функцию $W_p(x, y)$ для описания комплексного потенциала в потенциальной форме:

```
def W_p(x, y):  
    # координаты источника  
    x0 = -3  
    y0 = 2  
    # расход жидкости  
    Q = 8 * cmath.pi  
  
    z = complex(x - x0, y - y0)  
    result = Q / (2 * cmath.pi) *  
            cmath.log(abs(z) * cmath.exp(complex(0, cmath.phase(z))))  
    return result
```

5. Вычислить значение этой функции в точке (-5, 3) и вывести результат:

```
result = W_p(-5, 3)  
print('W_p(-5, 3) =', round(result.real, 3), '+', round(result.imag,  
3), 'j')
```

Результат:

```
W_p(-5, 3) = 3.219 + 10.712 j
```

Значение комплексного потенциала, вычисленное с помощью функций $W_a(x, y)$ и $W_p(x, y)$ – одинаково.

6. Создать потенциальную функцию $w_x(x, y)$ и функцию тока $w_y(x, y)$:

```
def wx(x, y):  
    return W_a(x, y).real  
  
def wy(x, y):  
    return W_p(x, y).imag
```

Важно. При вычислении можно использовать функцию комплексного потенциала в любой форме (алгебраической или потенциальной).

7. Вычислить значение функций в точке (-5, 3) и вывести результат:

```
result = wx(-5, 3)  
print('wx(-5, 3) =', round(result, 3))  
  
result = wy(-5, 3)  
print('wy(-5, 3) =', round(result, 3))
```

Результат:

```
wx(-5, 3) = 3.219  
wy(-5, 3) = 10.712
```

8. **Самостоятельно** добавить возможность вводить точку, в которой нужно вычислить значения функций так, чтобы результат выглядел следующим образом (пользователь вводит точку с координатами (2.7, -5.45)):

```
Введите координаты точки:  
x = 2.7  
y = -5.45  
W_a( 2.7 , -5.45 ) = 8.955 + -3.671 j  
wx( 2.7 , -5.45 ) = 8.955  
wy( 2.7 , -5.45 ) = -3.671
```

9. Вычислить значение комплексного потенциала в точке (-3, 2).

Результат:

```
ValueError: math domain error
```

Выдается сообщение об ошибке, так как в этой точке находится источник, в котором значение потенциальной функции – бесконечность. Эту ошибку можно исключить, используя блок обработки исключений

```
try
...
except
...
```

В разделе **try** размещаются операторы, в которых может возникнуть ошибка, в разделе **except** – действие, которое нужно выполнить, если ошибка возникла.

В нашем случае вернем из функции значение False, при возникновении ошибки:

```
def W_alg(x, y):
    # координаты источника
    x0 = -3
    y0 = 2
    # расход жидкости
    Q = 8 * cmath.pi
    z = complex(x - x0, y - y0)

    try:
        result = Q / (2 * cmath.pi) * cmath.log(z)
    except:
        return False

    return result
```

Вычислим значение функции, а затем проверим, можно ли вычислять остальные функции (координаты точки вводятся в переменные point_x и point_y):

```
result = W_a(point_x, point_y)
if not result:
    print('Ошибка при вычислении комплексного потенциала!')

else:
    print('W_a...')
    result = wx(point_x, point_y)
    print('wx(...)')
    result = wy(point_x, point_y)
    print('wy(...)')
```

10. Протестировать полученную программу на заданных в условии точках. Результат:

Тест 1:

```
Введите координаты точки:
x = -5
y = 3
W_a( -5.0 , 3.0 ) = 3.219 + 10.712 j
wx( -5.0 , 3.0 ) = 3.219
wy( -5.0 , 3.0 ) = 10.712
```

Тест 2:

```
Введите координаты точки:
x = -3
y = 2
Ошибка при вычислении комплексного
потенциала!
```

Критерии оценки текущей аттестации студентов

Критерии оценки устного опроса

✓ 10-8,5 баллов выставляется студенту, обнаружившему всестороннее, систематическое и глубокое знание программного материала, умение применять его и владение изученным материалом; излагающему ответы полно, последовательно и логически стройно; усвоившему взаимосвязь основных и производных понятий; проявившему творческие способности в знании, умении и владении изученным материалом; знающему, умеющему и владеющему навыками приемами выполнения практических заданий и профессиональных задач; показывающему знакомство с основной и дополнительной учебной литературой; способному самостоятельно пополнять и развивать знания, умения и навыки в профессиональной деятельности

✓ 8,5-7,6 баллов выставляется студенту, обнаружившему системное знание, хорошее умение и владение учебным материалом; излагающему ответы грамотно и по существу заданных вопросов; не допускающему грубых неточностей; умеющему применять основные методики решения стандартных задач; способному самостоятельно пополнять умения и навыки в учебной деятельности

✓ 7,5-6,1 балл выставляется студенту, обнаружившему знание программного материала в объеме, необходимом для дальнейшей учебы и профессиональной деятельности; усвоившему взаимосвязь основных понятий; допускающему в ответах неточности, испытывающему затруднения при решении практических задач, способному ликвидировать пробелы в знаниях и умениях под руководством преподавателя

✓ 6-5 баллов выставляется студенту, обнаружившему большие пробелы в знании основного программного материала; допускающему принципиальные ошибки в изложении материала или в ответах на вопросы; не умеющему применять имеющиеся знания в решении практических и

профессиональных задач; не владеющему основными методиками решения задач или испытывающему значительные затруднения в этом.

Критерии оценки проверочных тестов

Оценка	Процент выполнения
неудовлетворительно	<50%
удовлетворительно	50%-64%
хорошо	65%-79%
отлично	80%-100%

Критерии оценки лабораторной работы

✓ 10-8 баллов выставляется студенту, если он выполнил все задания лабораторной работы, в том числе и самостоятельные. Фактических ошибок, связанных с пониманием проблемы, нет; семантических и синтаксических ошибок нет; все инструменты изучаемых средств программирования используются верно. При защите студент отвечает на все вопросы преподавателя.

✓ 7-6 баллов – работа выполнена полностью; студент выполнил все предложенные в лабораторной работе задания, одно самостоятельное задание реализовано не для всех исходных данных или есть погрешности в решении; все инструменты изучаемых средств программирования используются верно. При защите студент отвечает на все вопросы преподавателя.

✓ 5-4 балла – работа выполнена полностью. Два самостоятельных задания реализованы не для всех входных данных или есть значительные погрешности в решении; часть инструментов изучаемых средств программирования используются неверно. При защите студент не отвечает на 1-2 вопроса преподавателя.

✓ 1-3 балла – работа выполнена не полностью. Самостоятельные задания не выполнены или студент демонстрирует слабое владение инструментами изучаемых средств программирования. При защите студент не отвечает более, чем на 2 вопроса преподавателя.

Критерии выставления оценки студенту на экзамене по дисциплине

«Информационные технологии в прикладной механике»

Баллы (рейтинговой оценки)	Оценка экзамена/зачета (стандартная)	Требования к сформированным компетенциям
86-100	«отлично»	Оценка «отлично» выставляется студенту, если он глубоко и прочно усвоил программный материал по информационным технологиям, исчерпывающе, последовательно, четко и логически стройно его излагает, умеет составлять алгоритм решения задачи, реализовывать ее средствами языка программирования, а затем выполнить ее документирование, свободно справляется с задачами, вопросами и другими видами применения знаний, причем не затрудняется с ответом при видоизменении заданий, свободно использует компьютер для сбора и анализа данных, выбирает эффективный алгоритм обработки информации, правильно обосновывает принятое решение, владеет разносторонними навыками и приемами выполнения практических задач, связанных с проектированием и реализацией пакетов прикладных программ в области прикладной механики.
76-85	«хорошо»	Оценка «хорошо» выставляется студенту, если он твердо знает материал по информационным технологиям, грамотно и по существу излагает его, не допуская существенных неточностей в ответе на вопрос, правильно применяет теоретические положения при решении практических вопросов и задач, связанных с проектированием и реализацией пакетов прикладных программ в области прикладной механики, владеет необходимыми навыками и приемами их выполнения с использованием выбранного языка программирования.
61-75	«удовлетворительно»	Оценка «удовлетворительно» выставляется студенту, если он имеет знания только основного материала по информационным технологиям, но не усвоил его деталей, допускает неточности, недостаточно правильные формулировки, нарушения логической последовательности в изложении программного материала, испытывает затруднения при выполнении практических работ, связанным с решением задач и применением средств программирования в области прикладной механики
0-60	«неудовлетворительно»	Оценка «неудовлетворительно» выставляется студенту, который не знает значительной части программного материала по информационным технологиям, допускает существенные ошибки, неуверенно, с большими затруднениями выполняет практические работы, связанные с решением задач на выбранном языке программирования. Оценка «неудовлетворительно» ставится студентам, которые не могут продолжить обучение без дополнительных занятий по соответствующей дисциплине.