

Министерство науки и высшего образования РФ  
Федеральное государственное автономное образовательное учреждение  
высшего образования  
«СИБИРСКИЙ ФЕДЕРАЛЬНЫЙ УНИВЕРСИТЕТ»  
**РАБОЧАЯ ПРОГРАММА ДИСЦИПЛИНЫ (МОДУЛЯ)**

Б1.О.15 Компьютерная геометрия и геометрическое  
моделирование

наименование дисциплины (модуля) в соответствии с учебным планом

Направление подготовки / специальность

02.03.01 Математика и компьютерные науки

Направленность (профиль)

02.03.01.31 Математическое и компьютерное моделирование

Форма обучения

очная

Год набора

2022

Красноярск 2023

## РАБОЧАЯ ПРОГРАММА ДИСЦИПЛИНЫ (МОДУЛЯ)

Программу составили \_\_\_\_\_

к.ф.-м.н., доцент, Голованов М.И.

\_\_\_\_\_  
должность, инициалы, фамилия

## 1 Цели и задачи изучения дисциплины

### 1.1 Цель преподавания дисциплины

«Компьютерная геометрия и геометрическое моделирование» — это раздел математики, изучающий алгоритмы решения геометрических задач. Такие задачи возникают в вычислительной математике, компьютерной графике, проектировании и др. Цель преподавания дисциплины – приобретение студентами знаний современных методов решения задач вычислительной геометрии и обработки графической информации. Курс знакомит студентов с основами компьютерной графики и с базовыми математическими понятиями и методами, лежащими в основе графических программных средств, использующихся в автоматизированных информационных системах, которые становятся все более важной областью техники. Информация, полученная при изучении этой дисциплины является основополагающей при изучении дисциплин бакалаврской подготовки по направлению 02.03.01 «Математика. Компьютерные науки».

### 1.2 Задачи изучения дисциплины

В соответствии с ФГОС ВПО в результате изучения дисциплины обучающиеся должны:

- знать способы задания кривых, поверхностей, трехмерных геометрических объектов и основные методы их изображения в различных средах, основные виды графических форматов изображения, методы визуализации при решении геометрических и динамических задач;
- уметь создавать изображения кривых, поверхностей, трехмерных геометрических объектов в различных средах, использовать методы компьютерной визуализации;
- владеть математическим аппаратом, информационными и компьютерными технологиями, необходимыми для решения этих задач.

В курсе «Компьютерная геометрия и геометрическое моделирование» ставится задача освоения современных информационных и компьютерных технологий для изображения и моделирования геометрических объектов.

### 1.3 Перечень планируемых результатов обучения по дисциплине (модулю), соотнесенных с планируемыми результатами освоения образовательной программы

Код и наименование индикатора достижения компетенции	Запланированные результаты обучения по дисциплине
<b>ОПК-4: Способен находить, анализировать, реализовывать программно и использовать на практике математические алгоритмы, в том числе с применением современных вычислительных систем</b>	
ОПК-4.1: Находит и анализирует математические алгоритмы для решения практических задач	<ul style="list-style-type: none"><li>- основные понятия, факты, концепции, принципы теорий математики и информатики;</li><li>- возможности базового пакета прикладных программ.</li><li>- анализировать информационные источники (сайты, форумы, периодические издания);</li><li>- ориентироваться в круге основных проблем и</li></ul>

	<p>использовать методы анализа и синтеза для получения новых научных знаний;</p> <ul style="list-style-type: none"> <li>- строить математическую модель с алгоритмом её реализации в соответствии с выбранными методами решения;</li> <li>- навыками систематизации и выбора необходимой информации согласно поставленной задаче;</li> <li>- способностью к освоению новых алгоритмов и программ в рамках тематики дисциплины;</li> </ul>
<p>ОПК-4.2: Реализовывает программно и использует на практике математические алгоритмы, в том числе с применением современных вычислительных систем</p>	<ul style="list-style-type: none"> <li>- классические методы, применяемые в математическом моделировании;</li> <li>- разрабатывать алгоритмы для решения поставленных задач и реализовывать их по средствам базовых языков программирования;</li> <li>- вести отладку и тестирование разработанных программ в рамках используемой ОС и поддерживающей среды разработки.</li> <li>- способностью к построению алгоритмов в рамках тематики дисциплины на базовом языке</li> </ul>

#### **1.4 Особенности реализации дисциплины**

Язык реализации дисциплины: Русский.

Дисциплина (модуль) реализуется без применения ЭО и ДОТ.

## 2. Объем дисциплины (модуля)

Вид учебной работы	Всего, зачетных единиц (акад.час)	Сем естр	
		1	2
<b>Контактная работа с преподавателем:</b>	<b>2,44 (88)</b>		
занятия лекционного типа	1 (36)		
практические занятия	1,44 (52)		
<b>Самостоятельная работа обучающихся:</b>	<b>1,56 (56)</b>		
курсовое проектирование (КП)	Нет		
курсовая работа (КР)	Нет		
<b>Промежуточная аттестация (Зачёт) (Экзамен)</b>	<b>1 (36)</b>		

### 3 Содержание дисциплины (модуля)

#### 3.1 Разделы дисциплины и виды занятий (тематический план занятий)

		Контактная работа, ак. час.							
№ п/п	Модули, темы (разделы) дисциплины	Занятия лекционного типа		Занятия семинарского типа				Самостоятельная работа, ак. час.	
				Семинары и/или Практические занятия		Лабораторные работы и/или Практикумы			
		Всего	В том числе в ЭИОС	Всего	В том числе в ЭИОС	Всего	В том числе в ЭИОС	Всего	В том числе в ЭИОС
<b>1. Основы векторной и растровой графики.</b>									
	1. Лекция 1. Предмет компьютерной геометрии Предмет и содержание курса. Растровая и векторная графика. Особенности. Различия. Достоинства и недостатки. Область применения векторной и растровой графики. Форматы векторных и растровых графических файлов. Цвет и цветовые модели. Аддитивная цветовая модель RGB.	2							
	2. Лекция 2. Координатный метод в компьютерной графике Однородные и барицентрические координаты. Двумерные и трёхмерные преобразования координат. Матричный подход. Преобразование точек (поворот, масштабирование, отражения, сдвиги). Преобразование прямых и плоскостей. Пересечение прямых и плоскостей. Комбинированные преобразования. Инварианты.	2							

<p>3. Лекция 3. Аффинные преобразования на плоскости Перенос, поворот вокруг произвольной точки. Отражение относительно произвольной прямой. Обозначение для двухмерного случая в компьютерной графике. Растяжение вдоль координатных осей. Отражение. Матричный подход.</p>	2							
<p>4. Лекция 4. Трехмерное аффинное преобразование Повороты вокруг координатных осей. Отражения относительно координатных плоскостей. Перенос. Композиция преобразований. Поворот вокруг произвольной оси. Отражение относительно произвольной плоскости</p>	2							
<p>5. Лекция 5. Проекция Иерархическая схема основных типов проекций. Параллельные проекции. Косоугольные проекции. Перспективная проекция. Изометрическая проекция. Методы создания перспективных видов. Точки схода и след точек. Стереографическая проекция. Восстановление трехмерных объектов по проекциям.</p>	3							
<p>6. Практическая работа №1. Структура файла .bmp; создание монохромного bmp-файла графика кусочно-линейной функции. Знакомство с растровым форматом изображения и со структурой bmp-файла. Работа над программой создания монохромного графика функции и сохранением его в bmp-файл.</p>			6					

7. Практическая работа №2. Структура файла .eps; создание монохромного eps-файла графика кусочно-линейной функции Знакомство с векторным форматом изображения и со структурой eps-файла. Работа над программой создания монохромного графика функции и сохранением его в eps-файл.			4					
8. Практическая работа №3. Построение плоских проекций куба при различных его аффинных преобразованиях. Знакомство с группой вращений трёхмерного пространства, построение проекции куба на заданную плоскость; разработка программы вращения куба и запись проекции в графический файл (из пр. раб. 1 или 2).			6					
9. Практическая работа №4. Базовые алгоритмы вычислительной геометрии: пересечения прямых и плоскостей, построение плоских кривых, кривая Безье. Программная реализация основных методов построения сплайнов и решения традиционных задач вычислительной геометрии о пересечениях геометрических объектов.			10					
10. Практическая работа №5. Заливка плоской фигуры. Знакомство с основными алгоритмами заливки плоской фигуры и их программная реализация с последующей записью результатов заливки в графический файл (из пр. раб. 1 или 2).			8					
11. Изучение теоретического материала и реализация алгоритмов							20	
<b>2. Компьютерные модели простейших геометрических объектов</b>								



<p>1. Лекция 6. Базовые вычислительные и растровые алгоритмы</p> <p>Вычислительная сложность геометрических задач. Отсечение отрезка. Построение выпуклой оболочки. Алгоритмы растеризации линий. Прямое вычисление координат и инкрементные алгоритмы. Построение плоских кривых. Классические методы. Полиномиальные сплайновые кривые. Алгоритм Брезенхэма для эллипса. Параметрические полиномиальные кривые. Кривая Безье.</p>	2							
<p>2. Лекция 7. Заливка плоской фигуры</p> <p>Простейший рекурсивный алгоритм закрашивания. Рекурсивный алгоритм закрашивания линиями. Алгоритмы заполнения, которые используют математическое описание контура.</p>	2							
<p>3. Лекция 8. Линии и поверхности в пространстве</p> <p>Конструирование простейших кривых и поверхностей. Кубические кривые в форме Безье. Частные случаи кубической кривой. Модели описания поверхностей. Аналитическая модель, параметрические полиномиальные поверхности. Полигональное представление трехмерных объектов (векторная полигональная модель). Линейно-узловая модель. Изолинии.</p>	2							
<p>4. Лекция 9. Геометрия фракталов</p> <p>Определение фрактала, салфетка Серпинского. Размерность. Фракталы в природе.</p>	1							
<b>3. Методы построения сеток</b>								

<p>1. Лекция 10. Выпуклые оболочки Отсечение (двумерное). Определение выпуклости – невыпуклости плоского многоугольника. Разбиение невыпуклого многоугольника на выпуклые. Алгоритм построения выпуклой оболочки конечного набора точек на плоскости. Близость и диаграммы Вороного. Пересечения.</p>	4							
<p>2. Лекция 11. Триангуляция Делоне Триангуляция. Области Вороного и триангуляция Делоне. Метод гранично-согласованных координатных систем. Эллиптические параболические и гиперболические генераторы сеток.</p>	4							
<p>3. Лекция 12. Методы построения двумерных сеток Алгебраические методы генерации сеток, двумерные ортогональные координатные системы. Конформные отображения и построение сеток. Адаптивные сетки. Классификация методов. Оценка качества сетки. Особенности построения сеток в сложных областях.</p>	4							
<p>4. Лекция 13. Итерационные методы построения трёхмерных сеток. Методы граничной коррекции. Построение первичной сетки. Коррекция первичной сетки. Методы на основе критерия Делоне. Построение триангуляции Делоне на заданном наборе точек. Триангуляция Делоне с ограничениями. Особенности технической реализации алгоритмов на основе критерия Делоне. Методы исчерпывания.</p>	4							

<p>5. Лекция 14. Прямые методы построения трёхмерных сеток</p> <p>Методы на основе шаблонов. Триангуляция параллелепипеда. Триангуляция цилиндра. Триангуляция шара. Методы отображения.</p>	2							
<p>6. Лабораторная работа №1. Триангуляция Делоне плоской фигуры.</p> <p>Программная реализация решения задач о близости, построение диаграмм Вороного и ассоциированной с ней триангуляции Делоне плоской фигуры с последующей записью результатов в графический файл (из лаб. раб. 1 или 2).</p>			4					
<p>7. Лабораторная работа №2. Триангуляция плоской фигуры методом граничной коррекции.</p> <p>Программная реализация алгоритма граничной коррекции для триангуляции плоской фигуры с последующей записью результатов в графический файл (из лаб. раб. 1 или 2).</p>			5					
<p>8. Лабораторная работа №3. Триангуляция плоской фигуры методом шаблонов.</p> <p>Знакомство с основными алгоритмами метода шаблонов и построение на их базе последовательности вложенных сеток (метод бисекции, фрактальный алгоритм).</p> <p>Программная реализация одного из алгоритмов триангуляции плоской фигуры методом шаблонов с последующей записью результатов в графический файл (из лаб. раб. 1 или 2).</p>			5					

9. Лабораторная работа №4. Триангуляция параллелепипеда, цилиндра, шара. Знакомство с основными трёхмерными шаблонами и построение на их базе сеток для пространственных областей.			4					
10. Изучение теоретического материала, реализация алгоритмов							36	
Всего	36		52				56	

## **4 Учебно-методическое обеспечение дисциплины**

### **4.1 Печатные и электронные издания:**

1. Никулин Е. А. Компьютерная геометрия и алгоритмы машинной графики: учебное пособие для вузов(Санкт-Петербург: БХВ-Петербург).
2. Никулин Е. А. Компьютерная графика. Модели и алгоритмы(Москва: Лань).
3. Голованов Н. Н., Ильютко Д. П., Носовский Г. В., Фоменко А. Т. Компьютерная геометрия: учебное пособие для вузов по направлениям подготовки "Информатика и вычислительная техника", "Информационные системы"(Москва).
4. Дегтярев В.М. Компьютерная геометрия и графика: учеб. для студентов вузов, обучающихся по специальности "Информационные системы и технологии" направления подгот. "Информационные системы" : рекомендовано УМО по университетскому политехническому образованию(М.: Академия).
5. Перфильев Д.А., Кушнарченко А.В., Сиротин Э.Е., Сарафанов А.В. Компьютерная геометрия и графика: электрон. учеб.-метод. комплекс дисциплины(Красноярск: СФУ).

### **4.2 Лицензионное и свободно распространяемое программное обеспечение, в том числе отечественного производства (программное обеспечение, на которое университет имеет лицензию, а также свободно распространяемое программное обеспечение):**

1. Microsoft Visual Studio,
2. MS Office

### **4.3 Интернет-ресурсы, включая профессиональные базы данных и информационные справочные системы:**

1. Информационные справочные системы не используются

## **5 Фонд оценочных средств**

Оценочные средства находятся в приложении к рабочим программам дисциплин.

## **6 Материально-техническая база, необходимая для осуществления образовательного процесса по дисциплине (модулю)**

Лекционные занятия должны проводиться в аудитории, оснащенной проекционным оборудованием, обеспечивающим показ компьютерных презентаций. Используемый при этом компьютер должен быть оснащен программным обеспечением, позволяющим выполнять необходимые примеры программного кода.

Компьютер может быть оснащен любой из операционных систем, обеспечивающих поддержку инструментальных средств, необходимых для проведения занятий.

В настоящее время основной упор сделан на использование операционной системы Windows. Однако возможно и использование ОС Linux, которая содержит программные средства, позволяющие использовать как уже разработанное методическое обеспечение, так и создаваемые программы. Преподаватель должен постоянно подчеркивать, что методы разработки программ, изучаемые в данной дисциплине, никак не связаны с типом операционной системы, хотя разрабатываемые продукты могут и не обладать свойством переносимости.

Для успешного проведения лекционных занятий необходимо обеспечить показ презентаций в формате MS Power Point.

Также необходимо иметь установленные средства для отображения исходных текстов программ на используемых в примерах языках программирования (C++, C).

В простейшем случае это могут быть обычные текстовые редакторы. Однако целесообразнее использовать текстовые редакторы с подсветкой синтаксиса или среды разработки, обеспечивающие отображение исходных текстов с подсветкой.

Желательна установка ПО со средой разработки, позволяющей собрать проект, откомпилировать и запустить его как в обычном режиме, так и в режиме отладки. Для ОС Windows – это может быть как продукт Visual C, входящий в пакет Microsoft Visual Studio версии не ниже 6.0. Возможно использование более простой среды, например, свободно распространяемого компилятора MinGW C++, помещенного в среду разработки Code::Blocks.

Подводя итог, можно рекомендовать следующие конфигурации программного обеспечения в зависимости от использования той или иной операционной системы:

При использовании операционной системы MS Windows:

Для показа презентаций необходимо использовать MS Power Point из пакета MS Office.

Для демонстрации исходных текстов программ, их компиляции запуска и отладки необходимо наличие среды разработки MS Visual Studio версии не ниже 6.0. Помимо этого для компиляции программ на языке C++ можно использовать свободно распространяемый компилятор MinGW C++ в среде Code::Blocks.

При использовании операционной системы Linux:

Для показа презентаций необходимо использовать программу Impress из свободно распространяемого пакета OpenOffice.org.

Для демонстрации исходных текстов программ, их компиляции запуска и отладки необходимо наличие сред разработки KDevelop, MonoDeveloper. Возможно также использование более простых текстовых редакторов, поддерживающих подсветку синтаксиса, например Kate.

Для компиляции программ на C++ необходимо использовать соответствующий компилятор из GCC.

Рассматриваемые программные продукты поставляются практически с любым дистрибутивом Linux.

Освоение дисциплины инвалидами и лицами с ограниченными возможностями здоровья, в зависимости от нозологий, осуществляется с использованием средств обучения общего и специального назначения.