

МИНИСТЕРСТВО НАУКИ И ВЫСШЕГО ОБРАЗОВАНИЯ РФ  
Федеральное государственное бюджетное образовательное учреждение высшего  
образования  
«ДАГЕСТАНСКИЙ ГОСУДАРСТВЕННЫЙ УНИВЕРСИТЕТ»  
Факультет математики и компьютерных наук

**РАБОЧАЯ ПРОГРАММА ДИСЦИПЛИНЫ**  
Компьютерная графика

Кафедра дискретной математики и информатики  
факультета математики и компьютерных наук

**Образовательная программа**  
**02.03.02 - Фундаментальная информатика и информационные технологии**

Профиль подготовки  
Информатика и компьютерные науки

Уровень высшего образования  
**бакалавриат**

Форма обучения  
**очная**

Статус дисциплины: базовый

Махачкала, 2018

Рабочая программа дисциплины «Компьютерная графика» составлена в 2018 году в соответствии с требованиями ФГОС ВО по направлению подготовки 02.03.02 - Фундаментальная информатика и информационные технологии (уровень бакалавриата) от 12.03.2015. №224

Разработчик: д.ф.-м.н., проф. по специальности 01.01.09 - «дискретная математика и математическая кибернетика» Магомедов Абдулкарим Магомедович




Рабочая программа дисциплины одобрена:

на заседании кафедры дискретной математики и информатики от 27.04.2018, протокол № 8

Зав. кафедрой  Магомедов А.М.

на заседании Методической комиссии факультета математики и компьютерных наук от 27.06.2018, протокол №6;

Председатель  В.Д. Бейбалаев

Рабочая программа дисциплины согласована с учебно-методическим управлением « »  2018 г. 

(подпись)





## **1. Цели изучения дисциплины**

Целями изучения дисциплины являются: 1) овладение основными алгоритмами компьютерной графики в и средствами языков программирования для их воплощения, 2) реализация в алгоритмах и программах методов математического анализа, аналитической и дифференциальной геометрии, алгебры, связанных с компьютерной графикой, 3) изучение основ графической библиотеки, 4) ознакомление с элементами компьютерного зрения.

Актуальность выбранной тематики обусловлена практически повсеместным использованием компьютерной графики в различных отраслях и сферах деятельности выпускника бакалавриата по направлению 02.03.02. Планируется, что ее изучение расширит диапазон специальностей, по которым академический бакалавр может трудоустроиться в данном регионе (Республика Дагестан); создаст умения и навыки относительно-свободного применения компьютерной графики в широком диапазоне: от дизайна интерьера до производства компьютерных игр, а также в вопросах создания привлекательного интерфейса для программного обеспечения видеоуроков по различным учебным дисциплинам средней и высшей школ; будет способствовать осознанию внутренних связей между дисциплинами фундаментальной математики – с одной стороны и компьютерной графикой – с другой.

## **2. Место дисциплины в структуре ОПОП бакалавриата**

Дисциплина входит в базовую часть образовательной программы бакалавриата по направлению 02.03.02 – «Фундаментальная информатика и информационные технологии» и изучается в соответствии с графиком учебного процесса в 6 семестре (28 ч. лек., 56 ч. лаб., экзам.).

Успешному изучению дисциплины способствуют знания, полученные в процессе изучения дисциплин компьютерных наук в течение 1-2 курсов, прежде всего – представления и понятия компьютерной графики, выработанные при изучении дисциплины «Технологии 3D моделирования» в 4-м семестре. Последнее, хотя и не является строго необходимым для изучения компьютерной графики, в ощутимой мере усиливает дидактический успех ее изучения. Из предшествующего изучения фундаментальных и общематематических дисциплин, в наибольшей степени востребованы «Математический анализ», «Алгебра», разделы, относящиеся к дисциплинам «Аналитическая геометрия» и «Дифференциальная геометрия» (тема).

В свою очередь, знания, полученные в процессе изучения компьютерной графики, используются при выполнении выпускных квалификационных работ, разнообразных конкурсных проектов. Составление программ, реализующих алгоритмы компьютерной графики, являются ценной поддержкой закрепления навыков программирования на языках высокого уровня и способствуют освоению программистского инструментария и аппаратного обеспечения, востребованных в области компьютерной графики.

### 3. Компетенции обучающегося, формируемые в результате освоения дисциплины

Процесс изучения дисциплины направлен на формирование элементов следующих компетенций в соответствии с ФГОС ВО по данному направлению подготовки.

Компетенции	Наименование компетенции из ФГОС ВО	Планируемые результаты обучения (показатели достижения заданного уровня освоения компетенций)
ОПК-2	способность применять в профессиональной деятельности современные языки программирования и языки баз данных, методологии системной инженерии, системы автоматизации проектирования, электронные библиотеки и коллекции, сетевые технологии, библиотеки и пакеты программ, современные профессиональные стандарты информационных технологий	Знать: основные алгоритмы компьютерной графики, в частности – метод двойной буферизации, метод плавающего горизонта и др. Уметь: применять встроенные средства языков высокого уровня для реализации основных алгоритмов компьютерной графики. Владеть: навыками разработки проектов с использованием GDI и GDI+

### 4. Объем, структура и содержание дисциплины

4.1. Объем дисциплины составляет 4 зачетных единиц, 144 академических часов.

4.2. Структура дисциплины

№ п/п	Раздел дисциплины	Семестр	Неделя семестра	Виды учебной работы, включая самостоятельную работу студентов и трудоемкость (в часах)					Формы текущего контроля успеваемости (по неделям семестра) Форма промежу-	
				Всего	Лек	Лаб.	СРС	Контроль		
<b>Модуль 1. Основы компьютерной графики</b>										
1	Определения компьютерной графики и мультимедиа. Стандартные форматы файлов.			1 2	2		4	5	1	Опрос, проверка лаб. заданий
2	Основные понятия интерфейса графических устройств			1 2	2		4	5	1	Опрос, проверка лаб. заданий
3	Алгоритмы визуализации с удале-			1	2		6	4		Опрос,

	нием невидимых точек			2						проверка лаб. заданий
	<b>Итого по модулю 1</b>			<b>3 6</b>	<b>6</b>		<b>1 4</b>	<b>1 4</b>	<b>2</b>	
Модуль 2. Базовые средства языка высокого уровня для реализации алгоритмов компьютерной графики										
	Класс Graphics, создание холста			8	2		2	4	1	Опрос, проверка лаб. заданий
	Вычерчивание графических примитивов методами класса Graphics			1 0	2		3	4	1	Опрос, проверка лаб. заданий
	Тема 3. Класс Bitmap, карандаш, кисть, текст			8	2		3	2		Опрос, проверка лаб. заданий
	Регионы. Фигурные вырезания и аффинные преобразования			1 0	2		4	3	1	Коллоквиум
	<b>Итого по модулю 2</b>			<b>3 6</b>	<b>8</b>		<b>1 4</b>	<b>1 1</b>	<b>3</b>	
Модуль 3. Трехмерная компьютерная графика										
	Вывод трехмерных фигур			1 0	2		4	4		Отчет по проекту
	3-мерная графика на основе библиотек (OpenGL/DirectX)			1 0	2		4	4		Отчет по проекту
	Библиотека компьютерного зрения			1 6	4		6	5	1	Проверка программы с в/камерой
	<b>Итого по модулю 4</b>			<b>3 6</b>	<b>8</b>		<b>14</b>	<b>1 3</b>	<b>1</b>	
Модуль 4. Подготовка к экзамену										
	<b>Итого по модулю 4</b>			<b>36</b>						<b>Экз</b>
	Итого			144	28		56	51	9	

### 4.3. Содержание дисциплины, структурированное по разделам и темам

#### 4.3.1. Содержание лекционных занятий по дисциплине

##### **Модуль 1. Основы компьютерной графики**

**Тема 1.** Определения компьютерной графики и мультимедиа. [2], 27-30.

Стандартные форматы файлов: BMP, GIF, EXIF, JPG, PNG и TIFF.

**Тема 2.** Основные понятия интерфейса графических устройств. Независимость приложений от аппаратуры. Контекст отображения. Размещение инструментария.

Двойная буферизация. Аффинные преобразования: перенос, поворот, масштабирование. [1], С. 63-94.

**Тема 3.** Алгоритмы визуализации с удалением невидимых точек: сортировка граней по глубине, метод плавающего горизонта, метод Z-буфера ([1], с. 161-163). Алгоритмы закрашивания поверхностей: модели отражения света (зеркальное и диффузное отражение). [1], С. 163-177.

##### **Модуль 2. Базовые средства языка высокого уровня для реализации алгоритмов компьютерной графики**

**Тема 1. Класс Graphics, создание холста** – объекта класса Graphics. Свойства SmoothingMode и Transform. Цвет, перо, кисть, шрифт, точка, прямоугольник.

**Тема 2. Вычерчивание графических примитивов методами класса Graphics:** Clear, DrawArc, DrawBezier, DrawClosedCurve, DrawCurve, DrawEllipse, DrawImage, DrawLine, DrawPolygon, DrawPath, DrawRectangle, DrawString, FillClosedCurve, FillPath, FillEllipse, FillPie, FillPolygon, FillRectangles, FillRegion.

**Тема 3. Класс Bitmap, карандаш, кисть, текст.** Конструкторы, свойства (Height, Size, Width), методы: Clone, Dispose, GetPixel, MakeTransparent, RotateFlip, Save, SetPixel.

Создание карандаша: цвет, толщина и стиль линии. [2], 62-73. Вывод текста. [2], 80-83.

Создание кисти для закраски внутренних областей геометрических фигур. Четыре типа кистей: стандартные (Brush), штриховые (HatchBrush), градиентные (LinearGradientBrush) И текстурные (TextureBrush).

**Тема 4. Регионы.** Фигурные вырезания и аффинные преобразования. Примеры регионов. Формы произвольных очертаний: в виде букв, в виде двух кругов.

##### **Модуль 3. Трехмерная компьютерная графика.**

**Тема 1. Вывод трехмерных фигур.** Создание реалистичных рельефов. [2], С. 304-317.

Изображение и управление поверхностями. [2], с. 319-332.

**Тема 2. 3-мерная графика на основе библиотек (OpenGL/DirectX).** [2], с. 378-400.



**Тема 3. Библиотека компьютерного зрения.** [5]. Распознавание текста на изображении [4].

4.3.2. Содержание лабораторных занятий по дисциплине

**Модуль 1. Основы компьютерной графики**

**Тема 1.** Определения компьютерной графики и мультимедиа. Структура стандартных форматы файлов (упражнения на языках программирования).

**Тема 2.** Основные понятия интерфейса графических устройств. Контекст отображения.

Двойная буферизация. 2-мерные и 3-мерные аффинные преобразования: перенос, поворот, масштабирование. [1], с. 63-94.

**Тема 3.** Алгоритмы визуализации. Сортировка граней по глубине, метод плавающего горизонта, метод Z-буфера [1], с. 161-163.

**Модуль 2.** Базовые средства языка высокого уровня для реализации алгоритмов компьютерной графики

**Тема 1. Класс Graphics, создание холста** – объекта класса Graphics. Свойства SmoothingMode и Transform. Цвет, перо, кисть, шрифт, точка, прямоугольник.

**Тема 2. Вычерчивание графических примитивов методами класса Graphics:** Clear, примитивы с префиксом Draw и примитивы с префиксом Fill.

**Тема 3. Класс Bitmap, карандаш, кисть, текст.**

Конструкторы, свойства. Создание карандаша: цвет, толщина и стиль линии. [2], 62-73.

Вывод текста. [2], 80-83.

Создание кисти для закраски внутренних областей геометрических фигур. Упражнения на создание четырех типов кистей.

**Тема 4. Регионы.** Фигурные вырезания и аффинные преобразования. Примеры регионов.

Вывод форм произвольных очертаний: в виде букв, в виде двух кругов.

**Модуль 3. Трехмерная компьютерная графика.**

**Тема 1. Вывод трехмерных фигур.** Создание реалистичных рельефов. [2], С. 304-317.

Изображение и управление поверхностями. [2], с. 319-332.

**Тема 2. 3-мерная графика на основе библиотек.** Введение в OpenGL. Синтаксис команд OpenGL [3], С. 27-29.

Конвейер рисования OpenGL: списки отображения ([3], с. 238-258), операции над вершинами, компоновка примитивов, операции над пикселями, компоновка текстур, растеризация, операции над фрагментами [3], С. 30-33.

Библиотеки, связанные с OpenGL: GLUT, набор инструментов OpenGL [3], С. 33-37.

Описание и отображение точек, линий, многоугольников. [3], С. 51-60. [3], С. 61-71.

Визуализация. [3], С. 96-141. Команды преобразования: модельно-видовые, проецирования. [3], С. 97-124. Управление стеком матриц. [3], С. 129-132.

**Тема 3. Библиотека компьютерного зрения.** [5]. Распознавание цифр и простых букв текста на изображении [4].

## 5. Образовательные технологии

Используемые образовательные технологии:

Проблемные лабораторные занятия;

Лабораторные занятия по обобщению и углублению знаний с элементами дискуссии;

Лекционные занятия в форме мини-конференции с использованием мультимедийных средств;

Коллоквиум;

Выполнение инициативных проектов (альтернативных шаблонным, запланированным проектам). Доля занятий с использованием активных и интерактивных методов составляет 90 %.

Изложение учебного материала сопровождается систематическими (на каждом занятии) компьютерными презентациями и демонстрацией решения задач с использованием мультимедийного оборудования.

Стиль лекционных занятий соответствует стилю проведения лекционных занятий по компьютерной графике на 2-м курсе ВМК МГУ, адаптированному к малочисленной аудитории.

Предусмотрено регулярное общение со студентами по электронной почте, а также публикация материалов лекций на сайте кафедры.

## 6. Учебно-методическое обеспечение самостоятельной работы студентов

*(Приводятся виды самостоятельной работы обучающегося, порядок их выполнения и контроля, дается учебно-методическое обеспечение (возможно в виде ссылок) самостоятельной работы по отдельным разделам дисциплины.*

### 6.1. Виды и порядок выполнения самостоятельной работы

1. Изучение лекционных материалов (электронные варианты) и рекомендованной литературы.
2. Выполнение индивидуальных заданий на составление программ и подготовка к отчету по ним.

3. Решение задач и упражнений, сформулированных в электронных приложениях к лекции.
4. Подготовка к текущему и промежуточному контролю.
5. Подготовка к экзамену

**6.2. Порядок контроля:** 1. Блиц-опрос на лабораторных занятиях, 2. Проверка выполнения пакета заданий и прием отчета по ним, 3. Текущий контроль за выполнением задач, сформулированных в электронных вариантах к лекции, 4. Промежуточный отчет (коллоквиумы), 5. Экзамен.

Раздел (модуль. тема)	Наименование самостоятельной работы	Практическое содержание	Контрольные сроки (в нед.)	Уч.-мет. обеспечение (указаны источники из списка основной литературы)
1.1	Основы Microsoft .NET Framework	Синтаксис языка C#. Разработка класса и реализация консольной программы на основе разрабатываемого класса	1	[6]
1.2	Основы Windows.Forms	Создание первого оконного приложения в .NET ("Убегающее окно"), создание меню, taskbar, дополнительных диалоговых окон, открытие и загрузка изображений	2	[7]
1.3	Многопоточное программирование в C#	Создание проекта с несколькими потоками	3	[8]
2.1	Введение в OpenGL. Начальные сведения о TAO framework	Тестирование визуализации 3D-сферы	4	[9]
2.2	Инициализация OpenGL в C#	Основы визуализации 2D примитивов в OpenGL. Вывод 2D треугольника с разложением цветового спектра. Визуализация графика функции	5	[10]
2.3	Создание растрового редактора на OpenGL	Оболочка программы, система слоев, завершение программы, оптимизация	6-7	[11]
2.4	Создание графических фильтров	Создание графических фильтров для обработки изображений с помощью OpenGL	8	[12]
3.1	Слайны	Реализация алгоритма на OpenGL	9-10	[13]
3.2	Алгоритмы геометрических пре-	Разработка проектов геометрических преобразова-	11	[14]

	образований в компьютерной графике	ний		
<b>3.3</b>	Формирование тел вращения	Реализация средствами OpenGL алгоритма построения тел вращения	12-13	[15]
<b>4.1</b>	Библиотека GLUT	Создание примитивов, трансформация, перенос и масштабирование объектов в OpenGL	14-15	[16]
<b>4.2</b>	Текстурирование	Текстурирование в OpenGL - библиотека DevIL.	16-17	[17]
<b>4.3</b>	3D-модели в OpenGL	Загрузка, текстурирование и визуализация 3D-моделей в OpenGL. Формат ASE.	18	[18]
	Система частиц	Алгоритм реализации системы частиц		[19]

Текущий контроль включает систематический блиц-опрос и проверку компьютерных программ.

Промежуточный контроль проводится в виде отчета по заданиям.

Итоговый контроль проводится в виде компьютерной экзаменационной работы с обязательным устным собеседованием по результатам компьютерной части.

Критерии выставления оценок:

«отлично» - владение теоретическим материалом (включая вопросы по открытой графической библиотеке и библиотеке компьютерного зрения), возможно, за исключением деталей справочного плана, и наличие навыков грамотного программирования, и умение составлять программы средней сложности с использованием графических библиотек;

«хорошо» - владение теоретическим материалом (без вопросов по библиотеке компьютерного зрения) и умение составлять простые программы;

«удовлетворительно» - владение основными теоретическими знаниями без вопросов по OpenGL и OpenCV.

### 6.3. Примеры заданий для самостоятельного решения.

Вывод сферы средствами GDI.

Вывод тора средствами GDI.

Вывод графика функции  $y=f(x)$  средствами GDI.

Вывод графика функции  $z=F(x,y)$  средствами GDI.

Опишите операцию поворота с помощью матричной операции.

Опишите операцию переноса с помощью матричной операции.

Опишите операцию масштабирования с помощью матричной операции.

Напишите программу вывода точки средствами OpenGL.

Напишите программу вывода окружности средствами OpenGL.

Напишите программу вывода цилиндра средствами OpenGL.

Напишите программу вывода сферы средствами OpenGL.

Как составить программу с использованием OpenGL, которая рисует одну сферу с поворотом на 60 градусов вокруг оси OZ, другую – с поворотом на 30 градусов?

Как вывести средствами OpenGL сферу с использованием модуля glut и без него?

## 7. Фонд оценочных средств для проведения текущего контроля успеваемости, промежуточной аттестации по итогам освоения дисциплины

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

Код компетенции из ФГОС ВО	Наименование компетенции из ФГОС ВО	Планируемые результаты	Процедура освоения
ОПК-2	способность применять в профессиональной деятельности современные языки программирования и языки баз данных, методологии системной инженерии, системы автоматизации проектирования, электронные библиотеки и коллекции, сетевые технологии, библиотеки и пакеты программ, современные профессиональные стандарты информационных технологий	Знать: основы алгоритмов компьютерной графики	Изучение тем 1-2-3 модуля 1
		Уметь: применять метод двойной буферизации и методы удаления невидимых граней.	Составление программ реализации данных методов без привлечения средств графических библиотек
		Владеть: навыками создания программ с применением средств GDI	Отладка и тестирование программ по темам 1-2-3 модуля 1

7.2. Типовые контрольные задания

*(Указываются темы эссе, рефератов, курсовых работ и др. Приводятся примерные тестовые задания, контрольные вопросы и задания для проведения текущего контроля и промежуточной аттестации по итогам освоения дисциплины.)*

### 7.2.1. Темы рефератов и курсовых работ

Инициализация библиотеки OpenGL в C#-проекте.

Вывод примитивов, значения параметра функции GLBegin. Команды OpenGL внутри GLBegin и GLEnd.

Прямое обращение к пикселям экрана. Команды `glRasterPos`, `glDrawPixels`, `glPixelStorei`, `glCopyPixels`, `glPixelTransfer`, `glReadPixels`.

Команды масштабирования, поворота и переноса (`glScalef`, `glRotate`, `glTranslatef`). Матричные аналоги. Применение команд `glPushmatrix`, `glPopMatrix`.

3-мерные системы координат. Схема преобразования координат объекта в оконные. Последовательность преобразования координат объекта в видовые (команды `glMatrixMode`, `glLoadMatrix`, `glLoadIdentity`, `glMultMatrix`, `gluLookAt`).

Проекции (`glOrtho`) и область вывода (`glViewport`).

Примитивы библиотеки GLU. Тип `gluQuadricObj`, функции `gluNewQuadric` и `gluQuadricDeawStyle`, построение сферы, диска, сектора.

Команды модуля DLUT для воспроизведения сферы куба, тора, конуса, чайника. Сплошное (каркасное) воспроизведение.

Применение дисплейных списков: создание (`glNewList-glEndList`), вызов (`CallList`, `CallLists`). Дополнительные команды (`glGenList`, `glDeleteLists`, `glPushAttrib`, `glPopAttrib`, `glIsList`, `glListBase`).

Смешение цветов и прозрачность. Команды `glEnable (gl_Blend)`, `glBlendFunc`. Использование буфера накопления (команда `glAccum`) и буфера трафарета.

Создание и наклеивание текстуры. Команды `glTexParameter`, `glTexImage2D`, `glTexEnvf`.

### **7.2.2. Примерные упражнения и задания к лабораторным занятиям и для самопроверки**

1 OpenGL базируется на двух основных библиотеках. Где они размещаются (в какой папке)?

2 Что Вам известно о понятиях «контекст устройства» и «формат пикселя»?

3 В чем смысл двойной буферизации?

4 Как в Delphi выводятся каркасные изображения? А закрасенные поверхности?

5 Как средствами OpenGL вывести 10 сфер разных радиусов?

6 Кратко опишите средства, предусмотренные в OpenGL для подключения одного источника света.

7 Объясните, как написать средствами OpenGL программу равномерного вращения сферы.

8 Объясните, как написать средствами OpenGL программу, которая выводит один неподвижный объект, а другой - подвижный (например, сфера).

9 Напишите процедуру установки формата пикселя с пояснениями к каждой команде и указанием, что здесь относится к OpenGL, а что – нет.

10 Синтаксис и семантика команды `glBegin glEnd`.

11 Когда команда OpenGL завершается символом `v`? Поясните на примере вывода 10 треугольников.

### 7.2.3. Примерный перечень заданий для текущего контроля (одно лабораторное занятие)

Разработка проекта для анимированного вывода сферы средствами GDI:

- 1 Выведите параметрические уравнения для координат текущей точки поверхности сферы как функций радиуса, долготы и широты.
- 2 Создайте и подготовьте графическую компоненту для вывода.
- 3 Создайте равновеликий второй буфер.
- 4 Как организовать встроены циклы для вычисления координат узловых точек сетки сферы?
- 5 Как проецировать узловые точки на плоскость?
- 6 Как выполнить каркасное соединение точек с удалением невидимых граней?
- 7 Как выполнить масштабирование для вывода проекции во второй буфер?
- 8 Когда и как переключить буферы?
- 9 Как выполнить анимацию сферы?
- 10 Как управлять приостановкой и запуском анимации?
- 11 Примерный перечень вопросов текущего контроля (по материалам одной лекции).

### 7.2.4. Примерные вопросы к промежуточному контролю (коллоквиуму)

Форматы BMP и GIF.

Основные понятия интерфейса графических устройств.

Независимость приложений от аппаратуры.

Контекст отображения.

Размещение инструментария.

Двойная буферизация.

Аффинные преобразования: перенос, поворот, масштабирование.

Алгоритмы визуализации с удалением невидимых точек (обзор).

Сортировка граней по глубине.

Метод плавающего горизонта.

Метод Z-буфера.

Класс Graphics.

Создание холста – объекта класса Graphics.

Свойства SmoothingMode и Transform.

Понятия: цвет, перо, кисть, шрифт, точка, прямоугольник.

Вычерчивание графических примитивов методами класса Graphics.

Конструкторы и методы класса Bitmap.

### 7.2.5. Примерные вопросы к итоговому контролю (экзамену)

Определения компьютерной графики и мультимедиа.

Стандартные форматы файлов: BMP, GIF, EXIF, JPG, PNG и TIFF.

Основные понятия интерфейса графических устройств. Независимость приложений от аппаратуры. Контекст отображения. Размещение инструментария.

Двойная буферизация. Аффинные преобразования: перенос, поворот, масштабирование.

Алгоритмы визуализации с удалением невидимых точек: сортировка граней по глубине, метод плавающего горизонта, метод Z-буфера.

Алгоритмы закрашивания поверхностей: модели отражения света (зеркальное и диффузное отражение).

Класс Graphics. Создание холста – объекта класса Graphics.

Свойства SmoothingMode и Transform.

Цвет, перо, кисть, шрифт, точка, прямоугольник.

Вычерчивание графических примитивов методами класса Graphics (5 методов по выбору преподавателя).

Класс Bitmap: конструкторы, свойства (Height, Size, Width), методы: Clone, Dispose, GetPixel, MakeTransparent, RotateFlip, Save, SetPixel.

Создание карандаша: цвет, толщина и стиль линии.

Вывод текста.

Создание кисти для закраски внутренних областей геометрических фигур.

Четыре типа кистей: стандартные, штриховые, градиентные и текстурные.

Регионы.

Фигурные вырезания и аффинные преобразования.

Примеры регионов.

Формы произвольных очертаний: в виде букв, в виде двух кругов.

Класс GraphicsPath.

Создание класса, свойства, методы: .AddEllipse (), .AddRectangle (), .Transform (объект Matrix).

График функции одной переменной. Выбор управляющего элемента, на холст которого выводится график (например, форма, панель, PictureBox и т.д.).

Вывод графиков нескольких функций с различными цветами и выбором ширины линий; пояснительные надписи.

Схема отображения графика с воображаемой клетчатой страницы ученической тетрадки на выбранный в программе холст; масштабирование при изменении размеров формы.

Анимация графика.

Вывод трехмерных фигур.

Создание объекта Bitmap в качестве второго буфера, параметрические формулы для x, y, z; масштабирование; рисование сетки.

Графики функций двух переменных.

Библиотека компьютерного зрения.

### **7.3. Методические материалы, определяющие процедуру оценивания знаний, умений,**



### **навыков и (или) опыта деятельности, характеризующих этапы формирования компетенций**

Общий результат складывается из текущего контроля - 50% и промежуточного контроля - 50%.

Текущий контроль по дисциплине включает:

- посещение занятий – 5 баллов,
- выполнение текущих лабораторных заданий – 20 баллов,
- выполнение домашних (аудиторных) контрольных работ - 25 баллов.

Промежуточный контроль по дисциплине включает:

- устный опрос - 25 баллов,
- компьютерная контрольная работа - 25 баллов.

## **8. Перечень основной и дополнительной учебной литературы, рекомендуемых для освоения дисциплины**

### **Основная литература**

0. Компьютерная графика [Электронный ресурс]: учебно-методический комплекс/ — Электрон. текстовые данные.— Алматы: Нур-Принт, 2012.— 150 с.— Режим доступа: <http://www.iprbookshop.ru/67066.html>.— ЭБС «IPRbooks»

1. Порев В.Н. Компьютерная графика. – СПб.: БХВ-Петербург, 2004. – 432с.
2. Жарков В.А. Компьютерная графика, мультимедиа и игры на Visual C# 2005. – М.: Жарков Пресс, 2005. – 812 с.
3. М. Ву, Т. Девис, Дж. Нейдер, Д. Шрайнер. OpenGL. Руководство по программированию. Библиотека программиста. – СПб.: Питер, 2006. – 624 с.
4. Уроки OpenGL + C# [Электронный ресурс] – Режим доступа: <http://esate.ru/uroki/OpenGL/uroki-OpenGL-c-sharp/> Время доступа: 25.05.2015.

### **Дополнительная литература**

1. Васильев С.А. OpenGL. Компьютерная графика [Электронный ресурс]: учебное пособие/ Васильев С.А.— Электрон. текстовые данные.— Тамбов: Тамбовский государственный технический университет, ЭБС АСВ, 2012.— 81 с.— Режим доступа: <http://www.iprbookshop.ru/63931.html>.— ЭБС «IPRbooks»
2. Уроки OpenGL + C#. О Microsoft .NET Framework [Электронный ресурс] – Режим доступа: <http://esate.ru/uroki/OpenGL/uroki-OpenGL-c-sharp/o-microsoft-net-framework/> Время доступа: 25.05.2015.
3. Уроки OpenGL + C#. Основы Windows.Forms [Электронный ресурс] – Режим доступа <http://esate.ru/uroki/OpenGL/uroki-OpenGL-c-sharp/osnovi-windows-forms/> Время доступа: 25.05.2015.
4. Уроки OpenGL + C#. Подробный анализ кода инициализации OpenGL в C# [Электронный ресурс] – Режим доступа <http://esate.ru/uroki/OpenGL/uroki-OpenGL-c-sharp/initializatsiya-opengl-v-c-sharp-step-2/> Время доступа: 25.05.2015.

## **9. Перечень рекомендуемых ресурсов информационно-телекоммуникационной сети «Интернет»**

См. пункты [4]-[20] предыдущего раздела.

## **10. Методические указания для обучающихся по освоению дисциплины**

- 1) Вначале рекомендуется освоить основные методы компьютерной графики без реализации средствами языков высокого уровня: двойная буферизация, удаление невидимых граней, аффинные преобразования (перенос, поворот, масштабирование) и др.
- 2) До применения развитых средств компьютерной графики рекомендуется освоить средства GDI.
- 3) При использовании языков типа C# необходимо уделять внимание средствам GDI+.
- 4) В зависимости от используемой операционной системы и языка программирования выберите одну из графических библиотек – OpenGL или DirectX. При использовании Windows 7, 8.1, 10.0 и языка C# можно, например, выбрать DirectX (продукты одной фирмы обычно не конфликтуют между собой).
- 5) Для лучшего освоения материала рекомендуется ознакомиться с разработками кафедры, получившие свидетельства Роспатента.
- 6) Материалы занятий периодически публикуются на сайте кафедры. Необходимо отслеживать электронные материалы.
- 7) Упражнения по первому модулю рекомендуется решить как на языке C#, так и на языке Delphi.
- 8) Настоятельно рекомендуется ознакомиться со следующими задачами повышенной трудности (и попытаться решить).
  - 8.1) Анимация надписей населенных пунктов на карте РД
  - 8.2) Анимация погоды на карте РД – дождь, снег, солнце.
  - 8.3) Нарисовать линию – оцифрованную шоссеиную дорогу, по которой едет рисунок автомобиля, поворачиваясь по направлению дороги.
  - 8.4) Быстрый показ известного художественного произведения 10000\*10000 в окне OPENGL
  - 8.5) По заданной контурной карте (с изолиниями) создать 3-мерный ландшафт местности.
  - 8.6) «Полет» прозрачного изображения персонажа по другому AVI-файлу.
  - 8.7) Выбрать 10 прилегающих районов карты РД. Их изображения должны перемещаться в окне OpenGL, отталкиваясь при соприкосновении. При нажатии на пробел складываться в карту.
  - 8.8) Собрать в папку изображения 1-секундных фонемов А, О, И, произнесенных однообразно одним и тем же человеком. Построить их огибающие и равномерно масштабировать в прямоугольниках одного и того же размера. Для вновь озвученного фонема выбирается наиболее близкий рисунок. Его огибающая заносится в базу.
  - 8.9) Графическое тестирование с автоматическим запуском сканера, поворотом рисунка, распознаванием.

## **11. Перечень информационных технологий, используемых при осуществлении образовательного процесса по дисциплине, включая перечень программного обеспечения и информационных справочных систем**

Операционная система Windows 7, 8.1 и 10, среда программирования MS Visual Studio. Используются также графические библиотеки OpenGL, DirectX и OpenCV. На части занятий используются веб-камеры.

## **12. Описание материально-технической базы, необходимой для осуществления образовательного процесса по дисциплине**

На каждой лекции используется стационарное мультимедийное презентационное оборудование (ауд. 3-73). Часть лекций предоставляется студенту в электронном формате. Лабораторные занятия проводятся в компьютерных классах с современным аппаратным и программным обеспечением. При выполнении лабораторных заданий студенту предоставляется право выбора одного из двух языков программирования из поддерживаемых MS Visual Studio. На сайте кафедры размещаются учебные пособия и презентации к лекции.