

МИНИСТЕРСТВО НАУКИ И ВЫСШЕГО ОБРАЗОВАНИЯ
РОССИЙСКОЙ ФЕДЕРАЦИИ

Федеральное государственное бюджетное
образовательное учреждение высшего образования
«Пензенский государственный университет» (ПГУ)

Ю. Н. Косников, А. П. Писарев

СОЗДАНИЕ ИЗОБРАЖЕНИЙ
СРЕДСТВАМИ ГРАФИЧЕСКИХ РЕДАКТОРОВ
И ГРАФИЧЕСКОЙ БИБЛИОТЕКИ

Лабораторный практикум

Пенза
Издательство ПГУ
2023

УДК 004.92
К71

Р е ц е н з е н т
доктор технических наук, профессор кафедры
«Системы автоматизированного проектирования»
Пензенского государственного университета
А. С. Бождай

Косников, Юрий Николаевич.
К71 Создание изображений средствами графических редакто-
ров и графической библиотеки : лаб. практикум / Ю. Н. Косников,
А. П. Писарев. – Пенза : Изд-во ПГУ, 2023. – 40 с.

Приведены задания и программы выполнения лабораторных работ с применением популярных средств компьютерной графики: векторного редактора Inkscape, растровых редакторов Gimp, Adobe Photoshop, графической библиотеки OpenGL. Показаны примеры результатов выполнения работ. Программы выполнения некоторых работ предусматривают дополнительные задания, выполняемые студентами самостоятельно. По всем работам приведены контрольные вопросы.

Издание подготовлено на кафедре «Информационно-вычислительные системы» ПГУ и предназначено для обучающихся по специальности 09.05.01 «Применение и эксплуатация автоматизированных систем специального назначения» и по направлению подготовки 09.03.03 «Прикладная информатика».

УДК 004.92

© Пензенский государственный
университет, 2023

СОДЕРЖАНИЕ

СОЗДАНИЕ ИЗОБРАЖЕНИЙ СРЕДСТВАМИ ВЕКТОРНОГО ГРАФИЧЕСКОГО РЕДАКТОРА INKSCAPE	4
Лабораторная работа 1. Создание геометрических образов средствами редактора векторной графики INKSCAPE	4
Лабораторная работа 2. Создание многоцветных векторных изображений	8
Лабораторная работа 3. Рисование схем	10
СОЗДАНИЕ ИЗОБРАЖЕНИЙ СРЕДСТВАМИ РАСТРОВЫХ ГРАФИЧЕСКИХ РЕДАКТОРОВ ADOBE PHOTOSHOP, GIMP.....	13
Лабораторная работа 4. Изучение инструментов выделения редактора GIMP, ADOBE PHOTOSHOP и работа со слоями.....	13
ОТОБРАЖЕНИЕ ПРОСТРАНСТВЕННЫХ СЦЕН СРЕДСТВАМИ ГРАФИЧЕСКОЙ БИБЛИОТЕКИ OPENGL	16
Лабораторная работа 5. Геометрические примитивы графической библиотеки	16
Лабораторная работа 6. Проектирование и визуализация пространственной сцены средствами графической библиотеки	23
Лабораторная работа 7. Моделирование освещения объектов средствами графической библиотеки	28
Лабораторная работа 8. Текстурирование объектов средствами графической библиотеки.....	33
Лабораторная работа 9. Визуализация динамической сцены средствами графической библиотеки	37

СОЗДАНИЕ ИЗОБРАЖЕНИЙ СРЕДСТВАМИ ВЕКТОРНОГО ГРАФИЧЕСКОГО РЕДАКТОРА INKSCAPE

Лабораторная работа 1 СОЗДАНИЕ ГЕОМЕТРИЧЕСКИХ ОБРАЗОВ СРЕДСТВАМИ РЕДАКТОРА ВЕКТОРНОЙ ГРАФИКИ INKSCAPE

1.1. Цель работы и лабораторное задание

Целью работы является изучение интерфейса редактора, получение практических навыков создания простых и комбинированных изображений. В задание входит рисование изображения с инициалами студента.

Для выполнения работы необходимо изучить инструменты рисования и редактирования объектов, знать способы создания и трансформации фигур, их перемещения и вращения, выделения и дублирования (пиктограммы инструментов показаны на рис. 1.1). Нужно изучить операции группировки, заливки и штриховки, а также правила использования панели «Выровнять и расставить», переход к которой осуществляется из панели «Объект» (рис. 1.2).

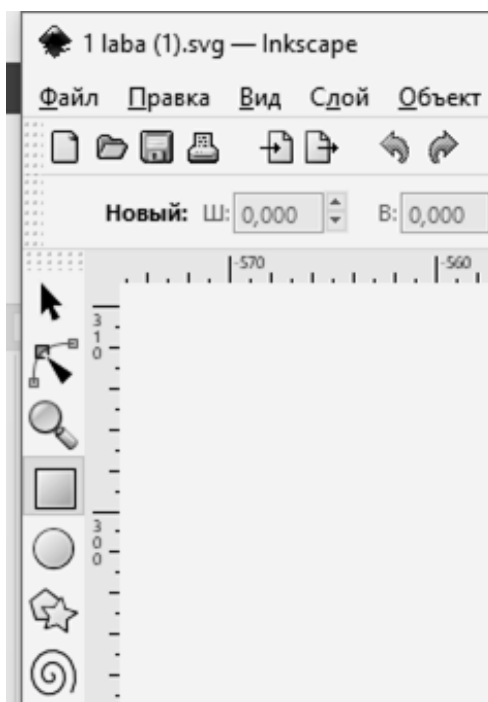


Рис. 1.1. Инструменты создания
геометрических объектов

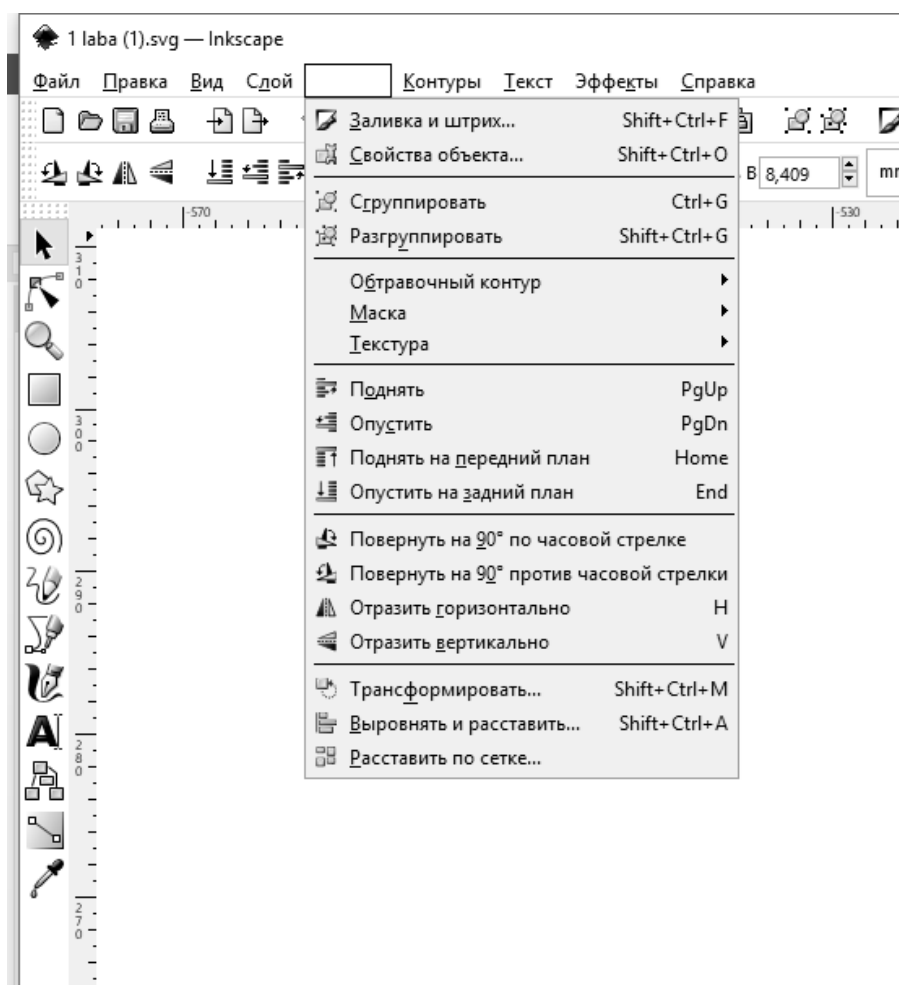


Рис. 1.2. Панель «Объект»

В лабораторное задание входит создание изображения трех букв, образующих инициалы исполнителя, с помощью различных геометрических фигур. Цвета заливки и штриховки, применяемые фильтры и эффекты должны быть различными для каждой фигуры. Ориентировочные размеры иллюстрации – формат А4. Фигуры внутри букв должны быть выровнены по горизонтали и вертикали с помощью панели «Выровнять и расставить».

1.2. Программа работы

Спроектировать изображение, определив его внешний вид, размещение, ориентацию, размеры, цветовые решения, способы заливки и штриховки геометрических фигур. Использовать все инструменты рисования геометрических фигур: прямоугольники, эллипсы, звезды и многоугольники, спирали, кривые, параллелепипеды.

После загрузки графического редактора создать библиотеку геометрических фигур, применив различные цвета, стили заливки и обводки фигур, размытие, прозрачность, а также текстурную заливку и инструменты панели «Фильтры» (фильтры выбрать по своему усмотрению). Из полученной библиотеки собрать буквы, выровнять их и сгруппировать. Под рисунком необходимо написать номер группы и фамилию исполнителя. Изображение сохранить в формате **svg** и представить на экране монитора для проверки преподавателю.

В отчете по лабораторной работе нужно описать ход выполнения задания: выбор фигур, их внешнего вида и эффектов, последовательность операций по созданию фигур (на примере одной – двух фигур). Пример изображения буквы, выполненного из геометрических фигур, приведен на рис. 1.3.

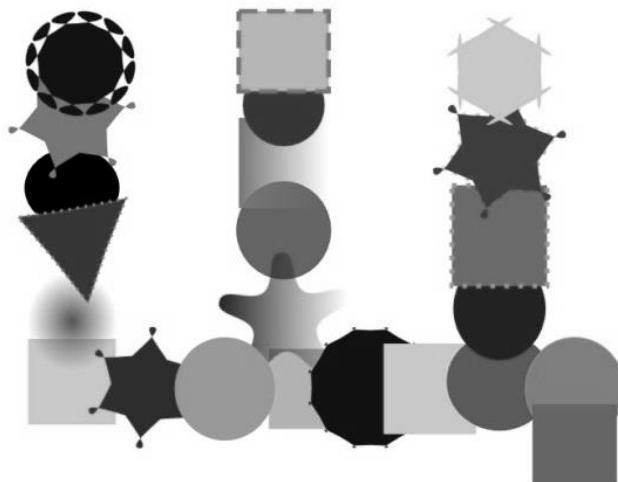


Рис. 1.3. Пример выполнения буквы из геометрических фигур векторного редактора

1.3. Контрольные вопросы и дополнительное задание к лабораторной работе 1

1. Расскажите о настройках инструмента выделения и трансформации объекта редактора Inkscape.
2. Расскажите о возможностях и правилах использования одного из инструментов (по указанию преподавателя).
3. На что влияет параметр «цвет, стиль обводки»?
4. Что такое направляющие, как их использовать?
5. Как «сочинить» новую текстуру заливки и использовать ее?
6. Как пользоваться панелью выравнивания?

После сдачи практической части работы и ответов на вопросы студент получает у преподавателя дополнительное задание. В качестве дополнительного задания выполняется самостоятельная работа по созданию простых графических объектов. Объекты создаются с использованием изученных инструментов геометрических форм и панели «Объект», примеры выполнения приведены на рис. 1.4.

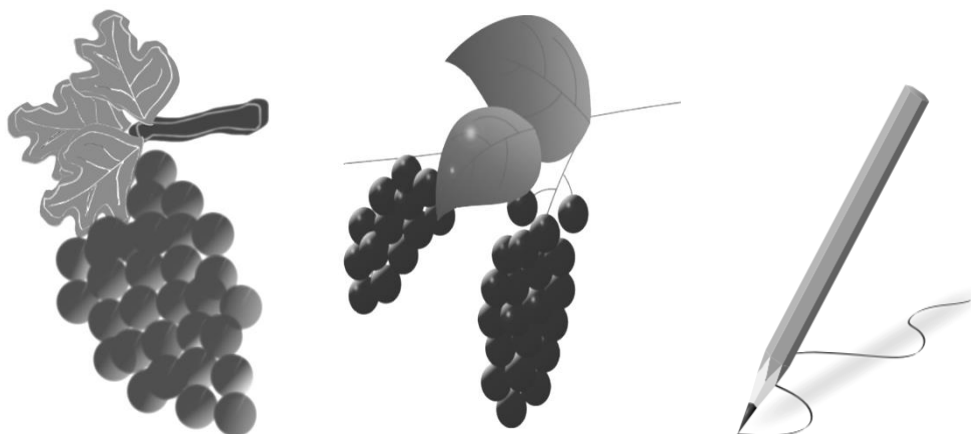


Рис. 1.4. Примеры выполнения дополнительного задания

Лабораторная работа 2

СОЗДАНИЕ МНОГОЦВЕТНЫХ ВЕКТОРНЫХ ИЗОБРАЖЕНИЙ

2.1. Цель работы и лабораторное задание

Целью работы является практическое освоение тексто-графических возможностей редактора Inkscape, их использование при проектировании логотипа и создании визитной карточки с включением в нее логотипа.

Необходимо, используя изученные инструменты, создать собственный логотип, отражающий направленность занятий и интересов исполнителя. Изображение может быть выполнено в палитре серого или быть цветным. Далее логотип используется при создании визитной карточки. В ней используются текстовые фрагменты, различные заливки и фильтры (по выбору исполнителя).

2.2. Программа работы

Создать эскиз логотипа, обсудить с преподавателем. Используя изученные инструменты рисования редактора Inkscape, нарисовать спроектированное изображение. Готовый логотип предъявить преподавателю для контроля и сохранить на диске в формате **svg**.

Следующий этап работы: с помощью панели «Файл», вкладка «Создать», выбрать формат визитной карточки (**bisness_card**). Для стандартной визитной карточки выбирается размер 85–54 мм или 90–50 мм, при необходимости размеры могут быть изменены. При конструировании визитки на рабочий стол помещается только одна карточка, доступная для редактирования. В каталоге шрифтов выбрать наиболее подходящий для оформления шрифт, с помощью инструмента «Вставка текста» создать нужную запись и расположить ее на заготовке визитки. Вставить логотип.

Изображение сохранить в формате **svg** и представить преподавателю на экране для проверки. Под рисунком необходимо написать номер группы и фамилию автора. В отчете по лабораторной работе описать выбор инструментов и эффектов, а также последовательность операций, примененных для создания визитной карточки.

Примеры внешнего вида визитных карточек с логотипами представлены на рис. 2.1.



Рис. 2.1. Примеры оформления визитных карточек

2.3. Контрольные вопросы и дополнительное задание к лабораторной работе 2

1. Перечислите инструменты, которые использованы при проектировании логотипа.
2. Опишите на примере визитки операции над текстом и объектами.
3. Опишите последовательность действий, необходимых для подготовки вывода на печать страницы текстового документа с иллюстрацией.
4. Расскажите об использовании панели «Заливка и обводка».

После сдачи практической части работы и ответов на вопросы студент получает у преподавателя дополнительное задание. В качестве дополнительного задания выполняется самостоятельная работа по созданию логотипа по предложенной преподавателем тематике. Примеры выполнения приведены на рис. 2.2.

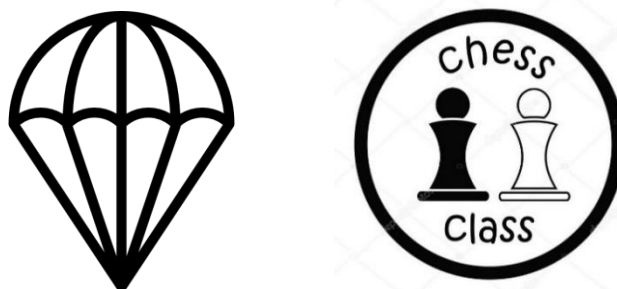


Рис. 2.2. Примеры логотипов парашютного и шахматного клубов

Лабораторная работа 3

РИСОВАНИЕ СХЕМ

3.1. Цель работы и лабораторное задание

Целью работы является изучение ГОСТ 19.701–90 Схемы алгоритмов, программ, данных и систем, его практическое применение при рисовании схем алгоритмов и добавление пояснительных надписей средствами редактора Inkscape.

3.2. Программа работы

Для выполнения работы необходимо изучить ГОСТ 19.701–90. Настоящий стандарт распространяется на условные обозначения (символы) в схемах программ, данных и программных систем и устанавливает правила выполнения схем, которые необходимо соблюдать при выполнении всех видов квалификационных работ.

Для графического представления студент может использовать в качестве основы алгоритм программы, выполненной им по дисциплине «Программирование», или взять за образец другой алгоритм. Количество графических символов в схеме не должно быть менее 16. В обязательном порядке должны быть использованы символы процесса, условия, границы цикла, предопределенного процесса, терминатора, комментария, соединителя. Символы нужно снабдить идентификаторами, комментарии должны содержать несколько строк. Соединительные линии в зависимости от их направления должны или не должны снабжаться стрелками (посмотреть правила в ГОСТ). Схема программы рисуется с помощью изученных ранее инструментов (геометрические фигуры, соединительные линии).

Готовое изображение представляется преподавателю на экране монитора и сохраняется в формате svg. В отчете приводится описание инструментов и настроек, примененных для рисования различных символов схемы.

Рекомендация: рационально сначала создать библиотеку примитивов – геометрических фигур (символов данных, символов процессов, других символов). Фрагмент библиотеки показан на рис. 3.1. Далее, используя библиотеку примитивов, несложно нарисовать требуемое изображение. Пример схемы программы, выполненной по ГОСТ 19.701–90, приведен на рис. 3.2.

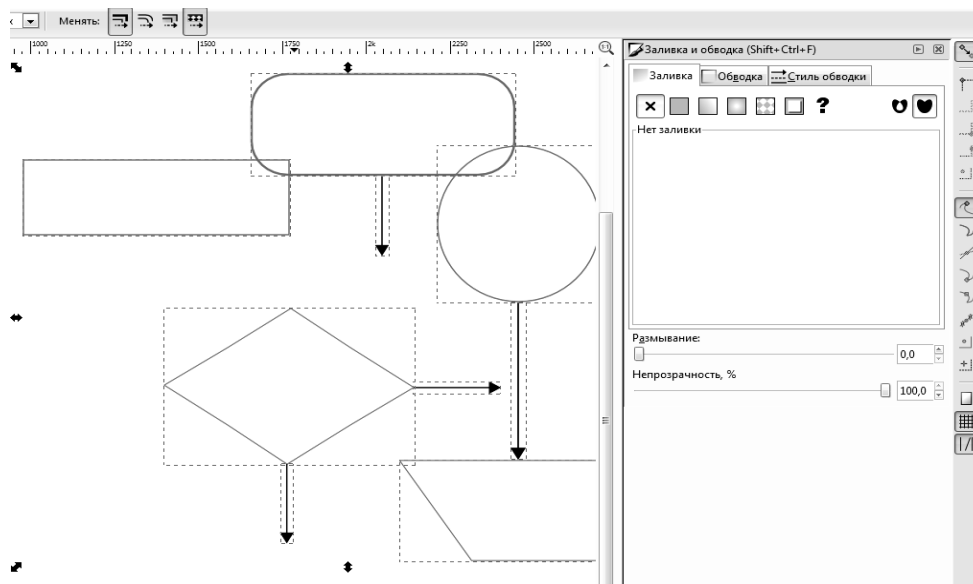


Рис. 3.1. Фрагмент библиотеки символов

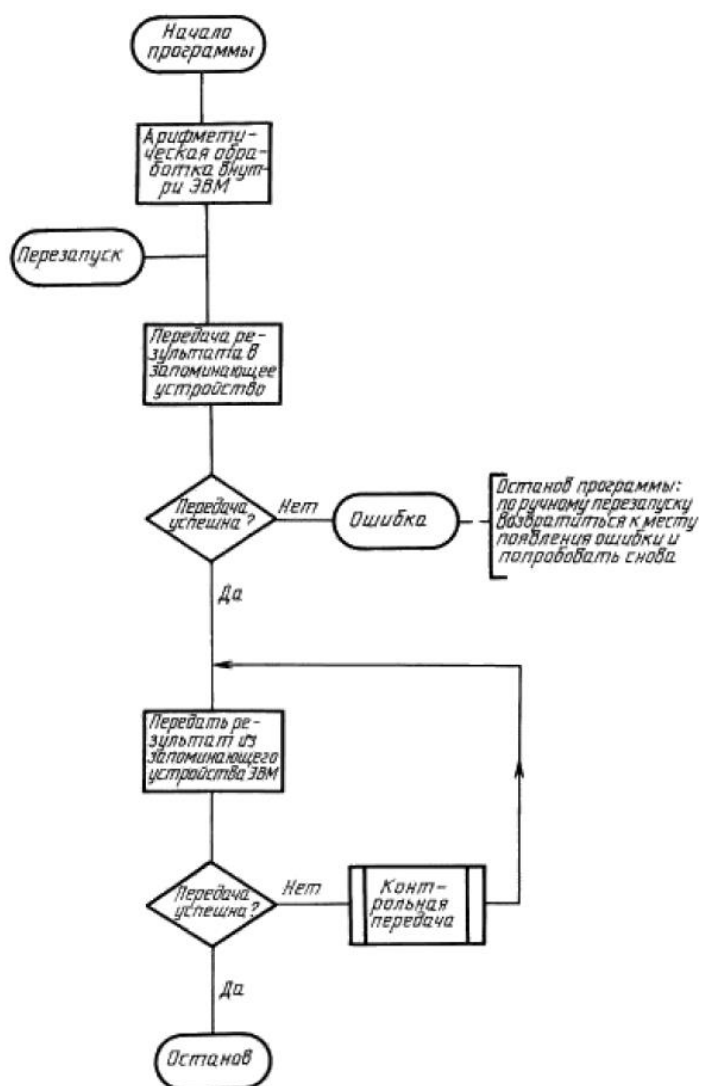


Рис. 3.2. Пример схемы программы

3.3. Контрольные вопросы и дополнительное задание к лабораторной работе 3

1. Расскажите о приемах и методах, которые использованы при рисовании схемы алгоритма.
2. Как определить необходимость нанесения стрелки на соединительную линию?
3. В чем различие между изображениями символов внутристраничного и межстраничного соединителя?
4. Для чего в схеме алгоритма применяются идентификаторы символов, комментарии?
5. Чем различается начертание символа данных и символа процесса?

После сдачи практической части работы и ответов на вопросы студент получает у преподавателя дополнительное задание. В качестве дополнительного задания выполняется самостоятельная работа по созданию оригинального векторного текста с использованием изученных инструментов и приемов. Пример выполнения приведен на рис. 3.3.



Рис. 3.3. Пример оформления дополнительного задания

СОЗДАНИЕ ИЗОБРАЖЕНИЙ СРЕДСТВАМИ РАСТРОВЫХ ГРАФИЧЕСКИХ РЕДАКТОРОВ ADOBE PHOTOSHOP, GIMP

Лабораторная работа 4 ИЗУЧЕНИЕ ИНСТРУМЕНТОВ ВЫДЕЛЕНИЯ РЕДАКТОРОВ GIMP, ADOBE PHOTOSHOP И РАБОТА СО СЛОЯМИ

4.1. Цель работы и лабораторное задание

Целью работы является изучение инструментов графического редактора GIMP, Photoshop и получение практических навыков создания многослойных изображений средствами редактора. В лабораторное задание входит создание многослойных изображений, которые включают фрагменты, выделенные из других графических объектов. Для выполнения работы необходимо изучить инструменты выделения, приемы работы со слоями многослойного изображения, знать способы копирования.

4.2. Программа работы

После загрузки графического редактора (GIMP, Adobe Photoshop) следует изучить набор его инструментов и их настройки. Для освоения работы с инструментами выделения необходимо загрузить изображение, выбранное базовым слоем-фоном, и, поочередно выделяя фрагменты других изображений (изображения выбираются студентом самостоятельно) всеми возможными инструментами используемого редактора, перенести их на базовый слой-фон. Для каждого фрагмента автоматически создается свой слой (нужно уметь менять порядок слоев). Каждому фрагменту нужно присвоить свой номер, который указывает на используемый инструмент. На полученное изображение наложить текстовый слой с номером лабораторной работы и фамилией автора.

Изображение сохранить в формате используемого растрового редактора (psd, XCF) и в одном из стандартных растровых форматов (jpeg, png и др.), а также представить на экране монитора для проверки преподавателем. В отчете нужно привести фоновое изображение, выделяемые фрагменты, итоговое многослойное изображение, а также описать цель применения инструментов, их названия и результат их применения.

В отчет следует вставить скриншот окна редактора, на котором видны названия созданных слоев. После сдачи практической части работы и оформления отчета студенту предстоит ответить на дополнительные вопросы.

Пример многослойного изображения приведен на рис. 4.1. На нем цифрами помечены использованные инструменты (1 – прямоугольное выделение, 2 – округлое выделение, 3 – лассо и т.д.).



Рис. 4.1. Пример многослойного изображения

4.3. Контрольные вопросы и дополнительное задание к лабораторной работе 4

1. Расскажите о настройках инструментов выделения редактора GIMP, Adobe Photoshop.
2. Какие инструменты выделения (перечислить) есть в редакторе GIMP, Adobe Photoshop?
3. Как выделение превратить в слой, в новое изображение?
4. Как поменять порядок видимости слоев?

После сдачи практической части работы и ответов на вопросы студент получает у преподавателя дополнительное задание. В качестве дополнительного задания выполняется самостоятельная работа по созданию коллажа. Тема коллажа предлагается студентом и утверждается преподавателем. Пример выполнения приведен на рис. 4.2.

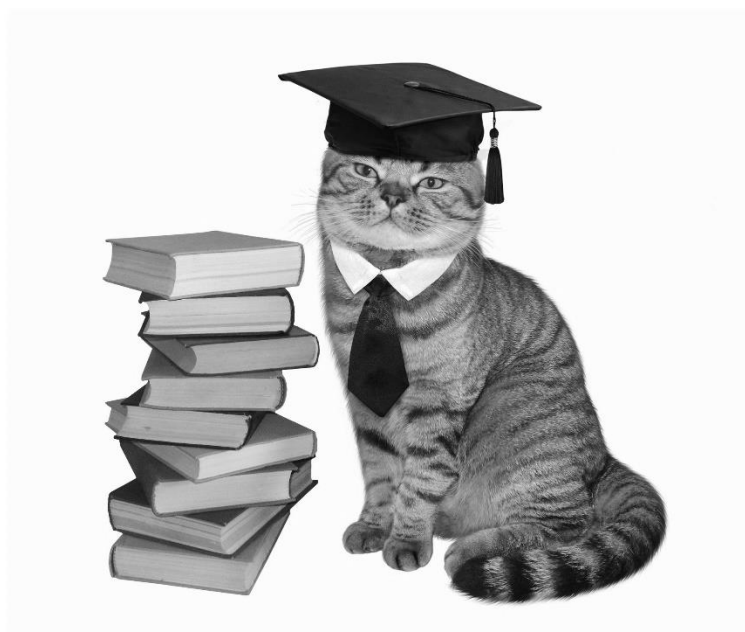


Рис. 4.2. Пример коллажа

ОТОБРАЖЕНИЕ ПРОСТРАНСТВЕННЫХ СЦЕН СРЕДСТВАМИ ГРАФИЧЕСКОЙ БИБЛИОТЕКИ OPENGL

Лабораторная работа 5 ГЕОМЕТРИЧЕСКИЕ ПРИМИТИВЫ ГРАФИЧЕСКОЙ БИБЛИОТЕКИ

5.1. Цель работы и лабораторное задание

Цель работы – изучить на практике отображение простых геометрических фигур в пределах нормализованного объема видимости. Лабораторная работа состоит из выполнения двух заданий. По первому заданию студенты формируют фигуры из плоских геометрических примитивов. По второму заданию рисуются поверхности второго порядка (квадрики) и многогранники (полиэдры).

Задание 1

Изучаемые команды:

- команды инициализации: `auxInitWindow Position`, `auxInitWindowSize`, `glutInitWindowPosition`, `glutInitWindowSize`;
- команды настройки модельно-видовых и проективных матриц: `glMatrixMode`, `glLoadIdentity`, `glOrtho`;
- команды настройки цветов фона и изображения: `glClearColor`, `glClear`, `glColor`;
- команды рисования геометрических примитивов: `glVertex`, `glBegin` – `glEnd`;
- команды задания стиля рисования: `glShadeModel`, `glLineWidth`, `glPointSize`, `glEnable(GL_LINE_SMOOTH)`, `glEnable(GL_POINT_SMOOTH)`.

Варианты заданий приведены в табл. 5.1.

Таблица 5.1

Номер варианта	Фигура	Геометрический примитив
1	2	3
1	Два отдельных неправильных треугольника	Отрезок (<code>GL_LINES</code>), треугольник (<code>GL_TRIANGLES</code>)
2	«Песочные часы» – два треугольника с общей вершиной	Связанные отрезки (<code>GL_LINE_STRIP</code>), треугольник (<code>GL_TRIANGLES</code>)

1	2	3
3	Правильный шестиугольник с диагоналями и точкой большого размера в центре	Связанные треугольники – «веер» (GL_TRIANGLE_FAN), точка (GL_POINTS)
4	Смежные неправильный треугольник и прямоугольник	Полигон (GL_POLYGON)
5	Два смежных неправильных треугольника	Треугольник (GL_TRIANGLES)
6	«Ёлочка» – вертикально расположенные три треугольника и четырехугольник	Треугольник (GL_TRIANGLES), четырехугольник (GL_QUADS),
7	Неправильный четырехугольник с точкой в вершине	Четырехугольник (GL_QUADS), точка (GL_POINTS)
8	Неправильный четырехугольник с диагональю	Связанные треугольники (GL_TRIANGLE_STRIP)
9	Неправильный четырехугольник с диагональю	Треугольник (GL_TRIANGLES)
10	Два смежных неправильных четырехугольника	Связанные четырехугольники (GL_QUAD_STRIP)
11	Неправильный четырехугольник с диагональю	Связанные треугольники (GL_TRIANGLE_FAN)
12	Контурный четырехугольник с диагональю, лежащий на квадрате	Связанные отрезки (GL_LINE_STRIP), полигон (GL_POLYGON)
13	Параллелограмм с точками на углах	Точка (GL_POINTS), полигон (GL_POLYGON)
14	«Домик» – смежные треугольник и квадрат	Треугольник (GL_TRIANGLES), четырехугольник (GL_QUADS)
15	Треугольник и квадрат с общей вершиной	Замкнутый контур (GL_LINE_LOOP), четырехугольник (GL_QUADS)
16	Треугольник, вписанный в прямоугольник	Треугольник (GL_TRIANGLES), связанные отрезки (GL_LINE_STRIP)
17	Смежные пятиугольник и треугольник	Полигон (GL_POLYGON)
18	«Крыша дома» – параллелограмм, сопряженный с треугольником	Треугольник (GL_TRIANGLES), четырехугольник (GL_QUADS)
19	«Флажок» – вектор и невыпуклый пятиугольник	Отрезок (GL_LINES), полигон (GL_POLYGON)
20	«Кораблик» – трапеция и вытянутый прямоугольник	Четырехугольник (GL_QUADS), отрезок (GL_LINES)

1	2	3
21	«Молоток» – трапеция и вытянутый прямоугольник	Полигон (GL_POLYGON), четырехугольник (GL_QUADS)
22	«Окно» – прямоугольник и многоугольник	Замкнутый контур (GL_LINE_LOOP), полигон (GL_POLYGON)
23	«Лопата» – выпуклый пятиугольник и вытянутый прямоугольник	Полигон (GL_POLYGON), отрезок (GL_LINES)
24	«Дупель один-один» – два смежных квадрата с точками в центрах	Четырехугольник (GL_QUADS), точка (GL_POINTS),
24	Пятиугольник с двумя диагоналями	Треугольник (GL_TRIANGLES), отрезок (GL_LINES)
26	«Веселый Роджер» – вытянутая по вертикали трапеция и ниже – косой крест из двух прямоугольников	Четырехугольник (GL_QUADS), полигон (GL_POLYGON)
27	«Рюмка» – равнобокая трапеция на 6-угольной ножке	Полигон (GL_POLYGON), замкнутый контур (GL_LINE_LOOP),
28	«Часы» – квадрат, два узких прямоугольника разной длины и крупная точка в центре	Связанные отрезки (GL_LINE_STRIP), четырехугольник (GL_QUADS), точка (GL_POINTS)
29	«Наполовину пустая бутылка» – прямоугольник и замкнутая ломаная линия из 8 сегментов	Четырехугольник (GL_QUADS), связанные отрезки (GL_LINE_STRIP)
30	«Квадратная гайка» – правильный квадрат с отверстием (многоугольником) внутри	Полигон (GL_POLYGON), четырехугольник (GL_QUADS)

Задание 2

Изучаемые команды:

– вспомогательные команды для отображения квадрик: gluNewQuadric, gluDeleteQuadric, glScale.

– команды рисования фигур: gluCylinder, gluDisk, gluPartialDisk, gluSphere, glut(Wire|Solid)Sphere, glut(Wire|Solid)Cone, glut(Wire|Solid)Cube, glut(Wire|Solid)Tetrahedron (в скобках через черту – варианты написания).

Варианты заданий приведены в табл. 5.2.

Таблица 5.2

Номер варианта	Фигура	Геометрический примитив
1	Эллиптический цилиндр	Цилиндр
2	Эллиптический диск с отверстием	Диск (gluDisk)
3	Сектор эллиптического диска с отверстием	Диск (gluPartialDisk)
4	Эллипсоид	Сфера
5	Конус	Конус или цилиндр
6	Усеченный конус	Цилиндр
7	Куб	Куб или цилиндр
8	4-гранная пирамида	Цилиндр или тетраэдр
9	6-гранная усеченная пирамида	Цилиндр
10	Прямоугольный параллелепипед	Цилиндр или куб
11	Вытянутая 4-угольная рамка	Диск
12	Вытянутый угол («клюшка»)	Диск (gluPartialDisk)
13	«Блин» – эллипсоид, одна из осей которого намного меньше двух других	Сфера
14	«Шпиль» – сильно вытянутая 4-гранная пирамида	Цилиндр или тетраэдр
15	«Граненая» сфера с 48 гранями	Сфера
16	«Лист бумаги» – параллелепипед, высота которого намного меньше длины и ширины	Диск или куб
17	«Наконечник» – треугольник малой толщины	Конус или цилиндр
18	«Пентагон» – пятиугольник с отверстием	Диск
19	Два конуса, соединенные основаниями	Сфера
20	«Доска» – плоский длинный параллелепипед	Цилиндр
21	Эллиптический диск	Диск (gluPartialDisk)
22	«Граненый» конус с 7 гранями	Конус или цилиндр
23	«Граненая» сфера с 8 гранями	Сфера
24	«Граненый» эллиптический цилиндр с 8 гранями	Цилиндр
24	Кольцо	Диск
26	Усеченный конус, высота которого гораздо меньше диаметра оснований	Цилиндр
27	Два концентрических кольца	Два диска (gluDisk)
28	Треугольник с отверстием в центре	Диск (gluDisk)
29	Трехсторонняя рамка («ворота»)	Диск (gluPartialDisk)
30	Параллелепипед с сечением в форме вытянутого ромба	Цилиндр

5.2. Программы выполнения заданий

Программа выполнения первого задания:

1. Спроектировать изображение, выбрав координаты вершин, которые ограничивают примитивы. Координаты x и y вершин не должны превышать ± 1 , координата z берется равной нулю. Выбрать размеры окна вывода. Определить цвет фона (не черный) и цвет изображения: разный для каждого примитива.
2. В программу-заготовку добавить команды задания окна и рисования изображения. Использовать формат команды задания вершин `glVertex3f`. Получить изображение заданных фигур на экране. Предъявить результаты преподавателю.
3. Изменяя координаты x и y вершин в диапазоне $(-4, +4)$, определить границы объема видимости по соответствующим координатам, а также направления координатных осей и точку их начала.
4. Задать координату z одной из вершин равной $+0.4$, затем -0.4 . Объяснить наблюдаемый эффект. Задать координату z одной из вершин равной $+4$, затем -4 . Объяснить наблюдаемый эффект.
5. Ответить на контрольные вопросы.

Программа выполнения второго задания:

1. Спроектировать изображение, выбрав для заданных примитивов значения геометрических параметров. Линейные размеры и радиусы фигур выбрать из диапазона $(0, 1)$. Соотношение (ширина/высота/глубина) задавать с помощью команды масштабирования. Выбрать цвета фона и фигуры.
2. Подготовить программу отображения заданной геометрической фигуры в окне. Фигуры будут выглядеть объемными, если прибегнуть к моделированию их освещения. Для этого перед рисованием необходимо разрешить их освещение командой `glEnable(GL_LIGHTING)`. Для улучшения восприятия фигуры следует задать ракурс ее представления, применив команды `glRotatef(30, 1, 0, 0)`; `glRotatef(30, 0, 1, 0)`.
3. В программу-заготовку добавить команды рисования окна вывода и геометрической фигуры. Получить изображение.
4. Изменяя параметры команд рисования, добиться соответствия изображения заданию. Выяснить зависимость между значениями всех параметров и внешним видом фигуры. Представить результаты преподавателю.
5. Ответить на контрольные вопросы.

5.3. Контрольные вопросы к заданиям лабораторной работы 5

Контрольные вопросы к первому заданию:

1. Для чего применяются команды `glMatrixMode`, `glLoadIdentity`, `glOrtho`?
2. В каком диапазоне могут быть заданы координаты вершин примитивов командой `glVertex`?
3. Какими задаются по умолчанию цвет пера и цвет фона? Как их можно изменить?
4. Какие размеры имеет область видимости? Как ее можно изменить?
5. Сколько раз нужно применить команду `glVertex` при рисовании замкнутого шестиугольника с помощью примитива `GL_LINES`, `GL_LINE_STRIP`, `GL_LINE_LOOP`?
6. Если по очереди задать координату z разных вершин объекта равной $+2$ или -2 , произойдет ограничение (отсечение) примитива, при этом линия отсечения будет иметь разный вид. Нужно объяснить вид линии отсечения.
7. Примитивы `GL_QUADS`, `GL_POLYGON`, `GL_TRIANGLE_STRIP`, `GL_TRIANGLE_FAN` можно окрасить со смешиванием цветов и без смешивания, т.е. чистыми цветами. Как это сделать? Как сделать, чтобы каждый треугольник этих примитивов имел свой цвет?

Контрольные вопросы ко второму заданию:

1. Поясните назначение параметров команды `gluCylinder`, `gluDisk`, `gluPartialDisk`, `gluSphere`. В каком диапазоне могут меняться значения параметров? На что это влияет?
2. Чем различаются примитивы `WireSphere` и `SolidSphere` расширения `glut`?
3. Среди геометрических примитивов библиотеки OpenGL нет пятигранной пирамиды. Можно ли ее получить, используя другие примитивы?
4. Что происходит в программе при выполнении команды `gluNewQuadric`?
5. Как будет нарисован цилиндр (`gluCylinder`), если перед его рисованием применить команду `glPolygonMode`?

По лабораторным работам, относящимся к отображению пространственных сцен средствами библиотеки OpenGL, оформляется один отчет.

В его первом разделе описывается ход выполнения двух заданий работы по созданию геометрических фигур из заданных примитивов. Нужно описать закон размещения фигур в системе координат окна вывода и выбор координат вершин геометрических примитивов. В отчет поместить скриншот полученного изображения и результаты выполнения п. 4 программы выполнения первого задания (можно в палитре серого).

Лабораторная работа 6

ПРОЕКТИРОВАНИЕ И ВИЗУАЛИЗАЦИЯ ПРОСТРАНСТВЕННОЙ СЦЕНЫ СРЕДСТВАМИ ГРАФИЧЕСКОЙ БИБЛИОТЕКИ

6.1. Цель работы и лабораторное задание

Цели работы: практическое изучение средств графической библиотеки, необходимых для выполнения модельно-видовых преобразований, проектирование и отображение пространственной сцены в параллельной и перспективной проекции.

Изучаемые команды:

- команды задания геометрических преобразований объектов: `glTranslate`, `glRotate`, `glScale`;
- команды проецирования: `glOrtho`, `glFrustum`, `gluPerspective`;
- команды работы со стеком: `glPushMatrix`, `glPopMatrix`;
- команды создания и вызова дисплейного списка: `glNewList`, `glEndList`, `glCallList`;
- команды работы с буфером глубины: `glEnable(GL_DEPTH_TEST)`, `glClear(GL_DEPTH_BUFFER_BIT)`.

Варианты задания приведены в табл. 6.1.

Таблица 6.1

Номер варианта	Расположение объектов в сцене	Поворот объектов
1	2	3
1	По диагонали объема видимости начиная с ближнего левого нижнего угла	$AX = 30^\circ$, $AY = -20^\circ$
2	По диагонали объема видимости начиная с ближнего левого верхнего угла	$AX = 30^\circ$, $AY = 20^\circ$
3	По диагонали объема видимости начиная с ближнего правого нижнего угла	$AX = -30^\circ$, $AY = -20^\circ$
4	По диагонали объема видимости начиная с ближнего правого верхнего угла	$AX = -30^\circ$, $AY = 20^\circ$
5	По диагонали объема видимости начиная с дальнего левого нижнего угла	$AX = 40^\circ$, $AY = -14^\circ$
6	По диагонали объема видимости начиная с дальнего левого верхнего угла	$AX = 30^\circ$, $AY = -20^\circ$
7	По диагонали объема видимости начиная с дальнего правого нижнего угла	$AX = 30^\circ$, $AY = -40^\circ$

1	2	3
8	По диагонали объема видимости начиная с дальнего правого верхнего угла	$AX = -14^\circ$, $AU = -30^\circ$
9	От середины нижнего ребра ближней грани объема видимости до середины верхнего ребра дальней грани	$AX = 30^\circ$, $AU = 40^\circ$
10	От середины верхнего ребра ближней грани объема видимости до середины нижнего ребра дальней грани	$AX = -30^\circ$, $AU = 30^\circ$
11	От середины левого ребра ближней грани объема видимости до середины правого ребра дальней грани	$AX = 30^\circ$, $AU = -34^\circ$
12	От середины правого ребра ближней грани объема видимости до середины левого ребра дальней грани	$AX = 40^\circ$, $AU = -10^\circ$
13	От ближнего левого нижнего угла объема видимости до середины верхнего ребра дальней грани	$AX = -24^\circ$, $AU = -40^\circ$
14	От ближнего правого нижнего угла объема видимости до середины верхнего ребра дальней грани	$AX = -20^\circ$, $AU = -40^\circ$
15	От ближнего левого верхнего угла объема видимости до середины нижнего ребра дальней грани	$AX = 24^\circ$, $AU = 40^\circ$
16	От ближнего правого верхнего угла объема видимости до середины нижнего ребра дальней грани	$AX = 20^\circ$, $AU = 40^\circ$
17	В середине правого ребра ближней грани объема видимости и в двух его дальних левых углах	$AX = 90^\circ$, $AU = 120^\circ$
18	От ближнего левого верхнего угла объема видимости до середины правого ребра дальней грани	$AX = -40^\circ$, $AU = 34^\circ$
19	От ближнего правого нижнего угла объема видимости до середины левого ребра дальней грани	$AX = 40^\circ$, $AU = 34^\circ$
20	От ближнего правого верхнего угла объема видимости до середины левого ребра дальней грани	$AX = -40^\circ$, $AU = -34^\circ$
21	От середины верхнего ребра ближней грани объема видимости до левого нижнего дальнего угла	$AX = 120^\circ$, $AU = -134^\circ$
22	От середины верхнего ребра ближней грани объема видимости до правого нижнего дальнего угла	$AX = -120^\circ$, $AU = 134^\circ$
23	От середины нижнего ребра ближней грани объема видимости до левого верхнего дальнего угла	$AX = 140^\circ$, $AU = 140^\circ$
24	От середины нижнего ребра ближней грани объема видимости до правого верхнего дальнего угла	$AX = -140^\circ$, $AU = -140^\circ$
24	От середины левого ребра ближней грани объема видимости до правого верхнего дальнего угла	$AX = 100^\circ$, $AU = -14^\circ$
26	От середины правого ребра ближней грани объема видимости до левого верхнего дальнего угла	$AX = 140^\circ$, $AU = -14^\circ$

1	2	3
27	От середины правого ребра ближней грани объема видимости до левого нижнего дальнего угла	$AX = 44^\circ$, $AU = -30^\circ$
28	В центре объема видимости и в двух дальних верхних его углах	$AX = 60^\circ$, $AU = -44^\circ$
29	В середине нижнего ребра ближней грани объема видимости и в серединах двух вертикальных ребер его дальней грани	$AX = -240^\circ$, $AU = 60^\circ$
30	От ближнего левого нижнего угла объема видимости до середины правого ребра дальней грани	$AX = 40^\circ$, $AU = -34^\circ$

6.2. Программа работы

1. Число объектов в сцене – не менее трех. В качестве объектов сцены использовать геометрические фигуры из заданий 1, 2 лабораторной работы 5. Оформить рисование объектов в виде дисплейных списков. Для получения требуемого числа объектов в сцене объекты рисуются несколько раз. Спроектировать сцену, выбрав необходимые команды и их параметры. Для повышения наглядности сцены размеры объектов задать с помощью операции масштабирования.

2. Разработать последовательность геометрических преобразований, необходимых:

а) для поворота вокруг оси глубины системы координат наблюдателя на заданный угол всей сцены,

б) для поворота каждого объекта сцены в отдельности вокруг оси глубины системы координат объекта.

Пояснить различие. Выбрать цвета фона и фигур. Ввести освещение командой `glEnable(GL_LIGHTING)`.

3. В программу-заготовку ввести команды сдвига и поворота системы координат. Чтобы упростить размещение объектов в сцене, использовать стек модельно-видовых матриц. Получить изображение сцены. Надлежащим выбором параметров команд сдвига добиться размещения объектов сцены в соответствии с заданием. Предъявить результат преподавателю.

4. Ввести перспективную проекцию сцены. Учесть, что по умолчанию наблюдатель находится в начале мировой системы координат и в случае перспективного проецирования не может видеть объекты, находящиеся «за спиной». Результат предъявить преподавателю.

5. Спланировать эксперимент, цель которого – определить направление координатной оси z нормализованного объема видимости. Определить направление оси. Представить результат эксперимента преподавателю.

6. Задать такие параметры команд `glTranslate`, `glRotate`, чтобы два графических объекта сцены перекрывались. Проанализировать вид сцены с включенным и выключенным буфером глубины.

7. Изменяя угол при вершине пирамиды видимости, установить связь между его величиной и размерами объектов на экране. Пояснить наблюдаемый эффект.

8. Ответить на контрольные вопросы.

6.3. Контрольные вопросы по лабораторной работе 6

1. Как изменить размеры объема видимости при параллельной проекции, при перспективной проекции?

2. Поясните суть параметров команд `glTranslate`, `glRotate`. Каковы диапазоны значений этих параметров? На какой угол повернется объект на экране, если к нему применить команду `glRotatef(7200.0,0.0,0.0,4.0)`?

3. Как используются в программе параметры команды `glScale`? Какое преобразование будет выполнено командой `glScalef(-1.0,-1.0,1.0)`, `glScalef(2.0,0.0,1.0)`?

4. Чем различаются команды `glFrustum` и `gluPerspective`?

5. Для чего применяется дисплейный список? Как он описывается, как вызывается?

6. Для чего применяется стек модельно-видовых матриц? Как им пользоваться?

7. Какова последовательность вывода на экран изображений трехмерных объектов при включенном и выключенном тесте глубины?

По лабораторным работам, относящимся к отображению пространственных сцен средствами библиотеки OpenGL, оформляется один отчет. В его втором разделе описывается ход выполнения работы по созданию пространственной сцены. Нужно описать выбор параметров для команд расстановки объектов в сцене; пояснить последовательность геометрических преобразований, необходимых для поворота на заданный угол всей сцены и каждого объекта сцены в отдельности; изложить особенности

представления сцены в параллельной и перспективной проекции; описать результаты экспериментов по п. 4, 6 программы работы. В отчет поместить скриншоты полученной сцены в параллельной и перспективной проекции, а также скриншот изображения, полученного при работе с буфером глубины (п. 6 программы работы), можно в палитре серого.

Лабораторная работа 7

МОДЕЛИРОВАНИЕ ОСВЕЩЕНИЯ ОБЪЕКТОВ СРЕДСТВАМИ ГРАФИЧЕСКОЙ БИБЛИОТЕКИ

7.1. Цель работы и лабораторное задание

Целью работы является повышение реалистичности отображения объектов за счет моделирования их отражающих свойств и затенения, применения источников света и материалов объектов с заданными характеристиками.

Изучаемые команды:

- команды задания свойств материала: `glMaterial`;
- команды, задающие свойства источника света: `glEnable(GL_LIGHTING)`, `glDisable(GL_LIGHTING)`, `glEnable(GL_LIGHTi)`, `glLight`;
- команды, задающие особенности освещения: `glNormal`, `glEnable(GL_NORMALIZE)`, `glLightModel`;
- команды запоминания настроек: `glPushAttrib`, `glPopAttrib`.

Варианты задания приведены в табл. 7.1.

Таблица 7.1

Номер варианта	Номер источника	Компоненты освещения: 1) ambient 2) diffuse 3) specular	Свойства материала: 1) ambient 2) diffuse 3) specular 4) emission
1	2	3	4
1	1	1) (0.1, 0.1, 0.1, 1.0) 2) (0.6, 0.8, 0.2, 1.0) 3) (1, 1, 1, 1)	1) (0.8, 0.8, 0.8, 1.0) 2) (0.7, 0.9, 0.8, 1.0) 3) (0.6, 0.6, 0.6, 1.0) 4) (0.2, 0.0, 0.0, 1.0)
2	2	1) (0.2, 0.2, 0.2, 1.0) 2) (0.7, 0.2, 0.8, 1.0) 3) (1, 0, 0, 1)	1) (0.7, 0.7, 0.0, 1.0) 2) (0.7, 0.8, 0.9, 1.0) 3) (0.8, 0.1, 0.2, 1.0) 4) (0.0, 0.2, 0.0, 1.0)
3	3	1) (0.3, 0.3, 0.3, 1.0) 2) (0.2, 0.8, 0.8, 1.0) 3) (1, 0.6, 1, 1)	1) (0.7, 0.6, 0.4, 1.0) 2) (0.9, 0.6, 0.4, 1.0) 3) (0.4, 0.9, 0.4, 1.0) 4) (0.2, 0.0, 0.2, 1.0)
4	4	1) (0.3, 0.1, 0.3, 1.0) 2) (0.8, 0.7, 0.6, 1.0) 3) (1, 1, 0.8, 1)	1) (0.7, 0.9, 0.7, 1.0) 2) (0.8, 0.9, 0.9, 1.0) 3) (0.6, 0.8, 0.9, 1.0) 4) (0.0, 0.2, 0.0, 1.0)

1	2	3	4
4	4	1) (0.3, 0.2, 0.1, 1.0) 2) (0.6, 0.8, 0.4, 1.0) 3) (0.8, 1, 1, 1)	1) (0.8, 0.8, 0.8, 1.0) 2) (0.7, 0.9, 0.8, 1.0) 3) (0.8, 0.6, 0.8, 1.0) 4) (0.2, 0.0, 0.2, 1.0)
6	6	1) (0.1, 0.2, 0.3, 1.0) 2) (0.6, 0.8, 0.8, 1.0) 3) (1, 0, 1, 1)	1) (0.8, 0.8, 0.8, 1.0) 2) (0.7, 0.9, 0.8, 1.0) 3) (0.9, 0.9, 0.9, 1.0) 4) (0.1, 0.1, 0.3, 1.0)
7	7	1) (0.2, 0.1, 0.4, 1.0) 2) (0.8, 0.8, 0.2, 1.0) 3) (1, 1, 0, 1)	1) (0.8, 0.8, 0.8, 1.0) 2) (0.7, 0.9, 0.8, 1.0) 3) (0.6, 0.9, 0.9, 1.0) 4) (0.0, 0.4, 0.0, 1.0)
8	4	1) (0.1, 0.2, 0.3, 1.0) 2) (0.6, 0.8, 0.6, 1.0) 3) (0, 1, 1, 1)	1) (0.8, 0.8, 0.8, 1.0) 2) (0.7, 0.9, 0.8, 1.0) 3) (0.9, 0.9, 0.6, 1.0) 4) (0.0, 0.2, 0.0, 1.0)
9	3	1) (0.1, 0.1, 0.1, 1.0) 2) (0.6, 0.8, 0.2, 1.0) 3) (0.6, 1, 0.7, 1)	1) (0.8, 0.8, 0.8, 1.0) 2) (0.7, 0.9, 0.8, 1.0) 3) (0.8, 0.8, 1.0, 1.0) 4) (0.0, 0.2, 0.2, 1.0)
10	2	1) (0.1, 0.1, 0.1, 1.0) 2) (0.6, 0.8, 0.2, 1.0) 3) (1, 0.8, 1, 1)	1) (0.8, 0.8, 0.8, 1.0) 2) (0.7, 0.9, 0.8, 1.0) 3) (1.0, 0.7, 0.2, 1.0) 4) (0.2, 0.2, 0.0, 1.0)
11	1	1) (0.1, 0.1, 0.1, 1.0) 2) (0.6, 0.8, 0.2, 1.0) 3) (1, 0.9, 0.4, 1)	1) (0.8, 0.8, 0.8, 1.0) 2) (0.7, 0.9, 0.8, 1.0) 3) (0.6, 0.6, 0.6, 1.0) 4) (0.1, 0.3, 0.0, 1.0)
12	0	1) (0.1, 0.2, 0.1, 1.0) 2) (0.6, 0.8, 0.2, 1.0) 3) (0.4, 0.4, 0.4, 1)	1) (0.8, 0.9, 0.8, 1.0) 2) (0.4, 0.9, 0.4, 1.0) 3) (0.2, 1.0, 0.2, 1.0) 4) (0.2, 0.3, 0.0, 1.0)
13	7	1) (0.1, 0.2, 0.1, 1.0) 2) (0.7, 0.2, 0.8, 1.0) 3) (1, 0.8, 1, 1)	1) (0.8, 0.9, 0.8, 1.0) 2) (0.4, 0.9, 0.4, 1.0) 3) (0.2, 1.0, 0.2, 1.0) 4) (0.2, 0.3, 0.0, 1.0)
14	6	1) (0.0, 0.2, 0.1, 1.0) 2) (0.6, 0.8, 0.2, 1.0) 3) (1, 0.9, 0.4, 1)	1) (0.8, 0.9, 0.8, 1.0) 2) (0.4, 0.9, 0.4, 1.0) 3) (0.2, 1.0, 0.2, 1.0) 4) (0.1, 0.3, 0.2, 1.0)
15	4	1) (0.1, 0.3, 0.1, 1.0) 2) (0.6, 0.3, 0.9, 1.0) 3) (1, 0.0, 0.8, 1)	1) (0.8, 0.9, 0.8, 1.0) 2) (0.4, 0.9, 0.4, 1.0) 3) (0.2, 1.0, 0.2, 1.0) 4) (0.0, 0.0, 0.0, 1.0)

1	2	3	4
16	4	1) (0.1, 0.2, 0.0, 1.0) 2) (0.2, 0.8, 0.8, 1.0) 3) (1, 0.0, 1, 1)	1) (0.8, 0.9, 0.8, 1.0) 2) (0.4, 0.9, 0.4, 1.0) 3) (0.2, 1.0, 0.2, 1.0) 4) (0.2, 0.0, 0.0, 1.0)
17	3	1) (0.1, 0.2, 0.2, 1.0) 2) (0.6, 0.8, 0.2, 1.0) 3) (1, 0.9, 0.4, 1)	1) (0.8, 0.9, 0.8, 1.0) 2) (0.4, 0.9, 0.4, 1.0) 3) (0.2, 1.0, 0.2, 1.0) 4) (0.0, 0.1, 0.0, 1.0)
18	2	1) (0.0, 0.2, 0.2, 1.0) 2) (0.6, 1.0, 0.0, 1.0) 3) (1, 0.9, 0.4, 1)	1) (0.8, 0.9, 0.8, 1.0) 2) (0.6, 0.9, 0.3, 1.0) 3) (0.2, 1.0, 0.2, 1.0) 4) (0.2, 0.1, 0.0, 1.0)
19	1	1) (0.1, 0.3, 0.2, 1.0) 2) (0.6, 0.2, 0.9, 1.0) 3) (1, 1, 1, 1)	1) (0.8, 0.9, 0.8, 1.0) 2) (0.8, 0.8, 0.9, 1.0) 3) (0.2, 1.0, 0.2, 1.0) 4) (0.2, 0.3, 0.1, 1.0)
20	0	1) (0.0, 0.3, 0.0, 1.0) 2) (0.0, 0.8, 0.1, 1.0) 3) (0.6, 0.9, 0.6, 1)	1) (0.8, 0.9, 0.8, 1.0) 2) (0.8, 0.8, 0.4, 1.0) 3) (1.0, 0.0, 0.2, 1.0) 4) (0.2, 0.2, 0.2, 1.0)
21	3	1) (0.0, 0.3, 0.2, 1.0) 2) (0.9, 0.2, 0.2, 1.0) 3) (1, 0.4, 0.2, 1)	1) (0.8, 0.9, 0.8, 1.0) 2) (0.4, 0.4, 0.9, 1.0) 3) (0.0, 0.6, 0.6, 1.0) 4) (0.3, 0.3, 0.3, 1.0)
22	4	1) (0.2, 0.2, 0.0, 1.0) 2) (0.4, 0.8, 0.4, 1.0) 3) (1, 0.0, 0.4, 1)	1) (0.8, 0.9, 0.8, 1.0) 2) (0.9, 0.9, 0.0, 1.0) 3) (0.9, 1.0, 0.0, 1.0) 4) (0.1, 0.3, 0.1, 1.0)
23	4	1) (0.3, 0.2, 0.0, 1.0) 2) (0.2, 0.8, 0.8, 1.0) 3) (1, 0.2, 1, 1)	1) (0.8, 0.9, 0.6, 1.0) 2) (0.4, 0.9, 0.4, 1.0) 3) (0.2, 1.0, 0.2, 1.0) 4) (0.2, 0.0, 0.0, 1.0)
24	6	1) (0.0, 0.0, 0.1, 1.0) 2) (0.2, 0.8, 0.2, 1.0) 3) (1, 0.9, 0.4, 1)	1) (0.8, 0.9, 0.8, 1.0) 2) (0.4, 0.9, 0.4, 1.0) 3) (0.2, 1.0, 0.2, 1.0) 4) (0.0, 0.3, 0.0, 1.0)
24	7	1) (0.1, 0.1, 0.0, 1.0) 2) (0.7, 0.2, 0.8, 1.0) 3) (0.8, 0.8, 1, 1)	1) (0.8, 0.7, 0.4, 1.0) 2) (0.4, 0.9, 0.4, 1.0) 3) (1.0, 1.0, 0.4, 1.0) 4) (0.2, 0.1, 0.0, 1.0)

1	2	3	4
26	3	1) (0.2, 0.3, 0.0, 1.0) 2) (0.9, 0.2, 0.2, 1.0) 3) (1, 0.4, 0.4, 1)	1) (0.8, 0.9, 0.8, 1.0) 2) (0.4, 0.9, 0.4, 1.0) 3) (0.2, 1.0, 0.2, 1.0) 4) (0.2, 0.2, 0.0, 1.0)
27	2	1) (0.0, 0.4, 0.2, 1.0) 2) (0.9, 0.6, 0.2, 1.0) 3) (1, 0.4, 0.4, 1)	1) (0.8, 0.6, 0.9, 1.0) 2) (0.9, 0.9, 0.4, 1.0) 3) (0.2, 1.0, 0.2, 1.0) 4) (0.2, 0.2, 0.2, 1.0)
28	1	1) (0.1, 0.2, 0.1, 1.0) 2) (0.7, 0.2, 0.8, 1.0) 3) (1, 0.8, 1, 1)	1) (0.8, 0.9, 0.8, 1.0) 2) (0.9, 0.9, 0.0, 1.0) 3) (0.9, 1.0, 0.0, 1.0) 4) (0.1, 0.1, 0.1, 1.0)
29	4	1) (0.0, 0.3, 0.2, 1.0) 2) (0.9, 0.2, 0.1, 1.0) 3) (1, 0.4, 0.2, 1)	1) (0.8, 0.9, 0.8, 1.0) 2) (0.6, 0.9, 0.3, 1.0) 3) (0.2, 1.0, 0.2, 1.0) 4) (0.2, 0.1, 0.0, 1.0)
30	4	1) (0.3, 0.1, 0.3, 1.0) 2) (0.8, 0.7, 0.6, 1.0) 3) (1, 1, 0.8, 1)	1) (0.8, 0.9, 0.8, 1.0) 2) (0.8, 0.8, 0.9, 1.0) 3) (0.2, 1.0, 0.2, 1.0) 4) (0.2, 0.0, 0.1, 1.0)

7.2. Программа работы

1. Спроектировать сцену, используя объекты заданий лабораторной работы 6. Для плоскостей полигональных объектов, не параллельных фронтальной плоскости xu , задать компоненты нормалей командой `glNormal`. Для приведения нормалей к единичной длине использовать команду `glEnable(GL_NORMALIZE)`.

2. В разделе описаний программы ввести векторы, задающие свойства источника освещения и материала объектов. Источник в сцене расположить «за спиной» наблюдателя.

3. В программу-заготовку ввести команды, включающие и настраивающие источник света с заданным номером, команды, задающие текущие свойства материала, и команды рисования. Для упрощения отладки установить параллельное проецирование объектов.

4. Получить изображение сцены. Манипулируя положением источника света и объектов, убедиться в изменении освещенности сцены. Перейти к перспективному проецированию объектов на экран. Предъявить изображение преподавателю.

5. Изменяя компоненты настроек источника света и материала, выяснить их влияние на внешний вид изображения. Ввести управление размером зеркального блика.

6. Ввести второй источник освещения, настройки и положение которого выбрать самостоятельно. Получить на поверхности объекта-квадрики два зеркальных блика разного цвета.

7. Ответить на контрольные вопросы.

7.3. Контрольные вопросы по лабораторной работе 7

1. Какие свойства материала задает команда `glMaterial`?
2. Какие свойства источника света задает команда `glLight`?
3. Для чего применяется команда `glLightModel`? Когда имеет смысл использовать в качестве второго параметра этой команды строку `GL_LIGHT_MODEL_LOCAL_VIEWER`?
4. Чем различаются параметры `ambient` и `diffuse` при настройке источника света, материала объекта?
5. При разрешении освещения командой `glEnable(GL_LIGHTING)` по умолчанию включается один источник света. Какой номер он имеет, каким светом светит, можно ли его перенастроить?
6. Можно ли управлять диаметром зеркального блика? Можно ли «выключить» блик?

По лабораторным работам, относящимся к отображению пространственных сцен средствами библиотеки OpenGL, оформляется один отчет. В его третьем разделе описывается ход выполнения работы по моделированию освещения пространственной сцены. Нужно описать порядок и средства настройки материала и источников света. В отчет поместить скриншоты изображений, полученных при выполнении пунктов 4 и 6 программы работы, можно в палитре серого.

Лабораторная работа 8

ТЕКСТУРИРОВАНИЕ ОБЪЕКТОВ СРЕДСТВАМИ ГРАФИЧЕСКОЙ БИБЛИОТЕКИ

8.1. Цель работы и лабораторное задание

Целью работы является повышение реалистичности отображения объектов за счет наложения на их поверхность характерного узора – текстуры; изучение средств текстурирования.

Изучаемые команды:

- команды подготовки текстур: `glEnable(GL_TEXTURE_2D)`, `glBindTexture`, `glTexImage2D`, `gluQuadricTexture`, `auxDIBImageLoad`;
- команды наложения текстуры: `glTexParameter`, `glTexCoord`, `glTexEnv`, `glHint`.

Задания на текстурирование предусматривают применение в сцене двух текстур: для фоновой заставки и для геометрических объектов. Текстура заставки накладывается на отдельный четырехугольный примитив и имеет тему – лесную, морскую и т.д. Фон, элементы и детальность текстуры определяются студентами самостоятельно, но исходя из темы. Например, «полевая» текстура может включать изображения цветов и насекомых на голубом фоне. Фоновая текстура дополняется фамилиями исполнителей. Текстура для объектов сцены имеет геометрический рисунок. Цветовые решения текстур определяются студентами. Обе текстуры выполняются инструментами графических редакторов «от руки».

Варианты задания на текстурирование приведены в табл. 8.1

Таблица 8.1

Номер варианта	Тема текстуры для фона	Текстура для объектов
1	2	3
1	Полевая	Шахматное поле
2	Горная	Линейчатое поле
3	Морская	Диагональное поле
4	Лесная	В разноцветный горошек
5	Марсианская	Концентрические круги
6	Городская	Поле с разноцветными звездами
7	Пустынная	Поле с ромбами
8	Осенняя	Поле с треугольниками

1	2	3
9	Космическая	Поле с ломаной линией
10	Зимняя	Поле с разноцветными эллипсами
11	Техническая	Концентрические пятиугольники
12	Праздничная	Пляшущие человечки
13	Медицинская	Поле с разноцветными трапециями
14	Вулканическая	Знаки зодиака
14	Транспортная	Разноцветные синусоиды
16	Компьютерная	Параллельные полосы разной ширины и цвета
17	Музыкальная	Разноцветные треугольные флажки
18	Библиотечная	Треугольник, вписанный в окружность
19	Военная	Векторы, выходящие из одной точки (пауки)
20	Производственная	Разноцветные параллелограммы
21	Сельскохозяйственная	Разноцветные эллиптические кольца
22	Научная	Разноцветные круговые секторы
23	Студенческая	Разноцветные круговые спирали
24	Географическая	Правильные шестиугольники
24	Цирковая	Крестики и нолики
26	Химическая	Перечеркнутые окружности
27	Поэтическая	Разноцветные неправильные четырехугольники из линий большой толщины
28	Офисная	Спираль
29	Спортивная	Правильный треугольник, вписанный в окружность
30	Аптечная	Четыре кружка разного цвета

8.2. Программа работы

1. Спроектировать сцену, используя объекты лабораторной работы 6 и добавив общую заставку. Изготовить текстуры средствами растрового графического редактора. Размеры текстур выбрать кратными степени числа 2, при выборе размеров текстур учесть соотношение размеров объектов и фона. Текстуры сохранить в формате bmp.

2. В программу-заготовку ввести команды подготовки и наложения текстур на объекты сцены. В программах на C-подобных языках использовать команды расширения GLAUX, в программах на Delphi (Object Pascal) – свободно распространяемый модуль BMP.pas.

Для использования модуля BMP его необходимо присоединить к программе в разделе `uses`. Загрузка текстуры осуществляется с помощью вызова процедуры

`LoadTexture ('<имя>.bmp', TexName); ,`

где `<имя>` – имя bmp-файла текстуры; `TexName` (типа `GLuint`) – имя текстурного объекта.

После загрузки наложение текстуры на графический объект выполняется обычным образом (`glTexCoord`).

3. Для лучшего восприятия текстурированных объектов их цвет перед наложением текстур нужно сделать белым (задать единичные значения RGB-компонентам в команде `glMaterial`, `glColor`). Учсть, что параметры источника освещения могут перестраиваться перед отображением каждого графического объекта. Это позволяет подбором улучшить вид объектов.

4. Получить изображение сцены с текстурированными объектами. Предъявить изображение преподавателю. Установить зависимость вида текстурированных объектов от их удаления (при перспективном проецировании) и от расположения источника света.

5. Ответить на контрольные вопросы.

8.3. Контрольные вопросы по лабораторной работе 8

1. Зачем перед наложением текстуры на объект рекомендуется применить команду задания отражающих свойств поверхности объекта `glMaterial`?

2. Назовите требования, предъявляемые к рисунку, который предполагается использовать в качестве текстуры.

3. Для чего служат команды `glEnable(GL_TEXTURE_2D)`, `glBindTexture`?

4. Можно ли в одной программе чередовать отображение объектов с однотонной заливкой и с наложенной текстурой?

5. Как выбрать текстурные координаты для вершин объекта неправильной формы, чтобы при наложении текстуры ее внешний вид не искажился?

6. Куда вставляется в программе команда `glTexCoord`? Что задается в эту команду в качестве параметров? В каком диапазоне могут лежать значения этих параметров?

7. Как задать закон наложения текстуры на объект-квадрику?

По лабораторным работам, относящимся к отображению пространственных сцен средствами библиотеки OpenGL, оформляется один отчет. В его четвертом разделе описывается ход выполнения работы по текстурированию пространственной сцены. Нужно описать созданные текстуры, указать вид графического редактора, в котором они выполнены. Выбрать и описать закон наложения текстуры на объекты сцены, обосновать выбор значений текстурных координат, примененных для объектов. В отчет поместить скриншот изображения, полученного в результате текстурирования, можно в палитре серого.

Лабораторная работа 9

ВИЗУАЛИЗАЦИЯ ДИНАМИЧЕСКОЙ СЦЕНЫ СРЕДСТВАМИ ГРАФИЧЕСКОЙ БИБЛИОТЕКИ

9.1. Цель работы и лабораторное задание

Целью работы является совершенствование навыков работы с инструментами графической библиотеки, самостоятельное освоение и применение средств придания объектам пространственной сцены динамики поворота, перемещения и масштабирования в соответствии с заданием преподавателя. В качестве исходной используется сцена, спроектированная в лабораторной работе 8.

9.2. Программа работы

В процессе выполнения лабораторной работы необходимо в соответствии с заданием разработать законы движения или трансформации объектов, составить математическое или логическое описание их динамики. Нужно выбрать команды графической библиотеки, обеспечивающие реализацию заданных видов динамики, разработать последовательность их применения. Составить программу на выбранном языке программирования, предусмотрев программные средства реализации динамики. Динамическую сцену предъявить преподавателю для контроля.

9.3. Контрольные вопросы к лабораторной работе 9

1. С помощью каких средств программирования и графической библиотеки можно реализовать вращение объекта вокруг центра, положение которого не совпадает ни с одной вершиной объекта?

2. Чем различается реализация кругового движения объекта, при котором ось объекта остается параллельной самой себе, и кругового движения, при котором ось объекта все время проходит через центр кругового движения?

3. Как реализовать попеременное увеличение и уменьшение размеров объекта вдоль одной координаты объектной системы координат?

4. Как можно задать криволинейную траекторию движения объекта в окне вывода?

5. Почему при повороте 2D-объекта на 90° вокруг вертикальной оси он пропадает из вида? Проверьте, будет ли в этом случае пропадать из вида контурный примитив `GL_LINE_LOOP`.

По лабораторной работе 9 оформляется отдельный отчет. В нем следует описать ход создания динамической сцены. Особое внимание нужно уделить алгоритмам задания динамических преобразований в режиме реального времени. В отчете поместить и описать скриншоты нескольких фаз динамики изображения, можно в палитре серого.

Учебное издание

**Косников Юрий Николаевич,
Писарев Аркадий Петрович**

**СОЗДАНИЕ ИЗОБРАЖЕНИЙ
СРЕДСТВАМИ ГРАФИЧЕСКИХ РЕДАКТОРОВ
И ГРАФИЧЕСКОЙ БИБЛИОТЕКИ**

Редактор *А. Г. Темникова*
Технический редактор *Р. Б. Бердникова*
Компьютерная верстка *Р. Б. Бердниковой*

Подписано в печать 20.07.2023. Формат 60×84¹/₁₆.

Усл. печ. л. 2,32.

Заказ № 322. Тираж 20.

Издательство ПГУ.
440026, г. Пенза, ул. Красная, 40.
Тел.: (8412) 66-60-49, 66-67-77; e-mail: iic@pnzgu.ru

Вниманию авторов!

Издательство ПГУ выпускает учебную, научную и художественную литературу, презентационную и акцидентную продукцию, а также полноцветные юбилейные и мемориальные издания в соответствии с ГОСТ 7.60–2003.

Издательство ПГУ принимает к изданию рукописи, подготовленные с использованием текстового редактора Microsoft Word for Windows версий **2003 и выше**. Формат – А4, основной шрифт – Times New Roman, 14–16 pt через одинарный интервал (минимальный размер шрифта в таблицах и сносках – 12,5 pt). Тип файла в электронном виде – doc, docx.

Работа должна содержать индекс УДК, аннотацию.

Аннотация (ГОСТ 7.86–2003, ГОСТ 7.9–1995) включает характеристику основной темы, проблемы объекта, цели работы и ее результаты. В аннотации указывают, что нового несет в себе данный документ в сравнении с другими, родственными по тематике и целевому назначению. Аннотация может включать сведения о достоинствах произведения. Текст аннотации начинают фразой, в которой сформулирована главная тема документа. Заканчивается аннотация читательским адресом.

Рисунки и таблицы должны быть размещены в тексте после ссылки на них (растровые рисунки предоставляются в виде отдельных файлов в формате jpg, BMP с разрешением 300 dpi, векторные рисунки в формате Corel Draw с минимальной толщиной линии 0,75 pt. Рисунки должны быть доступны для правки!). Рисунки должны сопровождаться подрисовочными подписями, на все рисунки и таблицы в тексте должны быть ссылки.

Формулы в тексте выполняются только в редакторе формул **MathType версия 5.0** и выше. Символы греческого и русского алфавита должны быть набраны прямо, нежирно; латинского – курсивом, нежирно; обозначения векторов и матриц – прямо, жирно; цифры – прямо, нежирно. Наименования химических элементов набираются прямо, нежирно. Эти же требования необходимо соблюдать и в рисунках.

В списке литературы **нумерация источников** должна соответствовать очередности ссылок на них в тексте ([1], [2], ...). Номер источника указывается в квадратных скобках. Требования к оформлению списка литературы на русские и иностранные источники (ГОСТ Р 7.0.5–2008): для книг – фамилия и инициалы автора, название, город, издательство, год издания, том, количество страниц; для журнальных статей, сборников трудов – фамилия и инициалы автора, название статьи, полное название журнала или сборника, серия, год, том, номер, страницы; для материалов конференций – фамилия и инициалы автора, название статьи, название конференции, город, издательство, год, страницы.

К материалам **должна** прилагаться следующая информация: фамилия, имя, отчество, контактные телефоны.

Контакты Издательства ПГУ: т.: (8412) 66-60-49, 66-67-77, e-mail: iic@pnzgu.ru



**МИНИСТЕРСТВО НАУКИ И ВЫСШЕГО ОБРАЗОВАНИЯ
РОССИЙСКОЙ ФЕДЕРАЦИИ**

**Федеральное государственное бюджетное
образовательное учреждение высшего образования
«ПЕНЗЕНСКИЙ ГОСУДАРСТВЕННЫЙ УНИВЕРСИТЕТ»**

Ю. Н. Косников, А. П. Писарев

**СОЗДАНИЕ ИЗОБРАЖЕНИЙ
СРЕДСТВАМИ ГРАФИЧЕСКИХ РЕДАКТОРОВ
И ГРАФИЧЕСКОЙ БИБЛИОТЕКИ**

ПЕНЗА 2023

