ГОСУДАРСТВЕННОЕ БЮДЖЕТНОЕ ОБРАЗОВАТЕЛЬНОЕ УЧРЕЖДЕНИЕ

ВЫСШЕГО ОБРАЗОВАНИЯ МОСКОВСКОЙ ОБЛАСТИ

«Университет «Дубна»

ИНСТИТУТ СИСТЕМНОГО АНАЛИЗА И УПРАВЛЕНИЯ

Кафедра распределенных информационно-вычислительных систем

Кафедра информационных технологий

КУРСОВАЯ РАБОТА

по дисциплине

«Программированиена языке высокого уровня»

Моделирование вращения тороида

ТЕМА: \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

(наименование темы)

1013

Выполнил: студент группы \_\_\_\_\_\_\_\_\_

Ржевский В. С.

*\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_*

(Ф.И.О.)

Руководители:

по дисциплине ОИТ

*Отлично*

27.05.2020 г.

Доцент Мурадян А.В.

*\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_*

(ученая степень, ученое звание, занимаемая должность, ФИО)

Дата:\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

Оценка:\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

по дисциплине ПЯВУ

*\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_*

Ассистент Самойлов Ю.Е.

(ученая степень, ученое звание, занимаемая должность, ФИО)

Дата защиты:\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

Оценка:\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

Дубна, 2020

**Содержание**

[Введение 3](#_Toc41408803)

[Постановка задачи 4](#_Toc41408804)

[Теоретическая часть 5](#_Toc41408805)

[Тороид 5](#_Toc41408806)

[OpenGl 5](#_Toc41408807)

[Tao Framework 5](#_Toc41408808)

[Практическая часть 6](#_Toc41408809)

[Выводы 8](#_Toc41408810)

[Список литературы 9](#_Toc41408811)

[Приложение 10](#_Toc41408812)

# Введение

С изобретением компьютера жизнь человека сильно изменилась, ошибки в финансовых расчетах сократились до минимума, расшифровка закодированных сообщений ускорилась в разы, но для обычных пользователей такое понятие как **ЭВМ** оставалось далеким, непонятным и ненужным.

Отправной точкой развития компьютерной графики можно считать 1930 год, когда Владимир Зворыкин изобрел электронно-лучевую трубку, которая позволяла получать изображения на экране без использования механических движущихся частей. Первым компьютером с графическим интерфейсом пользователя принято считать *Macintosh*, который дал толчок для массового использования домашних компьютеров [1].

В современном мире компьютерная графика играет огромную роль: начиная от видеоигр, заканчивая кинематографом и программами для создания чертежей и сложных инженерных моделей.

# Постановка задачи

Цель: разработать программу, реализующую модель тороида в разных режимах рисования и возможностью вращения.

Исходные данные: информация, собранная в процессе выполнения курсовой работы, а также среда программирования — *Microsoft Visual Studio* 2019.

Правила пользования: интерфейс программы представляет собой окно отображения и инструменты для управления направлением и скоростью вращения тороида, и выбором его режима отображения.

Результат: программа, которая дает возможность пользователю управлять трехмерной фигурой.

# Теоретическая часть

## Тороид

Тором, или тороидом, называется поверхность, образованная вращением окружности вокруг оси, принадлежащей плоскости окружности, но не проходящей через ее центр. При этом ось вращения может пересекать окружность, касаться ее и располагаться вне окружности. В первых двух случаях тор называется закрытым, в последнем — открытым [2, 3]. Уравнение тора с расстоянием от центра образующей окружности до оси вращения *R* и с радиусом образующей окружности *r* может быть задано в параметрическом виде:

при

OpenGl

*OpenGl* — это название спецификации, которая описывает поведение системы отрисовки на основе процесса, в результате которого получается растровое изображение [5]. Она определяет API, программный интерфейс приложения, с помощью которого клиентское приложение может управлять этой системой [4, 6].

Tao Framework

*Tao Framework* — это свободно-распространяемая библиотека с открытым исходным кодом, предназначенная для быстрой и удобной разработки кроссплатформенного мультимедийного программного обеспечения в среде *.NET Framewrok*. Главными составляющими данной библиотеки является сама спецификация *OpenGl* и библиотека *FreeGlut*, позволяющая создавать окна, предоставляющие контекст *OpenGl*, а также взаимодействовать с мышью и клавиатурой [6].

**Практическая часть**

При запуске программы пользователь видит перед собой форму (см. Приложение, рис. 1), в левой части которой расположены инструменты управления тороидом, а справа окно отрисовки. Первоначально уже заданы следующие параметры: тип отображения — сеточный, скорость вращения — 0, оси вращения не выбраны.

Далее будут приведены части кода, описывающие работу программы. При запуске программы выполняется функция *Form1\_Load* (см. Приложение, рис. 2), в которой происходит инициализация класса *Glut*, необходимого для работы с функциями библиотеки *Glut*, устанавливается режим отображения *GLUT\_RGB* и *GLUT\_DOUBLE*, двойная буферизация, необходимая для плавного отрисовывания вращения тороида. В следующей строке происходит очистка окна вывода. Далее определяется область вывода, в данном случае по всей области элемента *space*, определяется матрица, над которой будут происходить дальнейшие операции, затем происходит определение проекции, данная функция, *gluPerspective*, строит пирамиду охвата видимости (см. Приложение, рис. 3), используя такие входные параметры как: угол визуального охвата, отношение сторон порта вывода, ближняя и дальняя плоскость просмотра. В конце снова вызывается функция определения матричного режима и устанавливается тест глубины для корректного отображения сцены.

Запуск самой сцены осуществляется по нажатию кнопки *Visual*, которая запускает таймер (см. Приложение, рис. 5). В самой функции тика таймера происходит изменение переменной *Angle* на заданную скорость, которая регулируется инструментом *trackBar1* (см. Приложение, рис. 4), и, в случае достижения переменной угла поворота значения 360, совершается ее обнуление, это необходимо для корректного отображения поворота. Основной вызываемой таймером функцией является функция *Draw* (см. Приложение, рис. 6). В данной функции сначала происходит очистка буферов цвета и глубины, и матрицы, так как до этого в окне вывода может быть уже что-то нарисовано, затем матрица помещается в стек, "камера" отдаляется на 6 единиц по оси Z, для комфортного обзора отрисовываемой сцены. Условные конструкции предотвращают, в случае если не выбрана ни одна ось вращения, переворачивание самой фигуры. В конце матрица снова помещается в стек, вызывается функция тора, матрица возвращается из стека, кадр визуализируется и происходит сама отрисовка сцены в поле вывода.

Функция моделирования тора (см. Приложение, рис. 7) представляет собой проверку на режим отображения, по умолчанию задан сеточный, и само заполнение точек. Входными параметрами данной функции являются радиусы образующей окружности и оси вращения, по которым проходятся первый и вложенный в него второй циклы. Третий вложенный цикл нужен для соединения соседних точек. Сами точки, полученные параметрическим уравнением, передаются в функции *glVertex3d* и в *glColor3d*. Первая функция определяет вершину в трехмерном пространстве, а вторая задает цвет каждой вершине с добавлением 0.5 для более насыщенного цвета.

Пример запущенной программы с измененными параметрами углами вращения и режимом отображения (см. Приложение, рис. 8).

# Выводы

В ходе выполнения данной курсовой работы была создана программа, которая дает представление о работе с программным интерфейсом — *OpenGl* и библиотекой *freeGlut*. Программу можно улучшить следующими способами: добавить другие поверхности и фигуры, дать возможность выводить несколько объектов одновременно, реализовать перемещение камеры с помощью клавиатуры.

# Список литературы

1. История компьютерной графики [Электронный ресурс] / Автор [Голышева Светлана](https://www.timetoast.com/users/2766858). Текстовые дан.; Режим доступа: https://www.timetoast.com/timelines/6248910a-e70f-4a32-b27c-e13ef9c355dc, свободный, — Загл. с экрана.
2. Математическое описание тороида [Электронный ресурс] / Сайт компании *Wolfram Research*. Электрон. Текстовые дан.; Режим доступа: <https://mathworld.wolfram.com/Torus.html>, свободный, — Загл. с экрана.
3. Поверхности вращения [Электронный ресурс] / Файловый архив. Электрон. Текстовые дан.; Режим доступа: https://studfile.net/preview/557582/, свободный, — Загл. с экрана.
4. *OpenGl wiki* [Электронный ресурс] / Сайт компании *Kronos*. Электрон. Текстовые дан.; Режим доступа: <https://www.khronos.org/opengl/wiki/Main_Page>, свободный, — Загл. с экрана.
5. Computer graphics from scratch. Part II: Rasterization. [Электронный ресурс] / Автор [Gabriel Gambetta](http://www.gabrielgambetta.com/computer-graphics-from-scratch/rasterization.html). Электрон. Текстовые дан.; Режим доступа: <https://www.gabrielgambetta.com/computer-graphics-from-scratch/rasterization.html>, свободный. — Загл. с экрана.
6. Уроки *OpenGl* в *C#*. [Электронный ресурс] / Сайт esate.ru. Электрон. Текстовые дан.; Режим доступа: <http://esate.ru/uroki/OpenGL/uroki-OpenGL-c-sharp/>, свободный. — Загл. с экрана.

# Приложение

****

Рис. 1. Стартовое окно

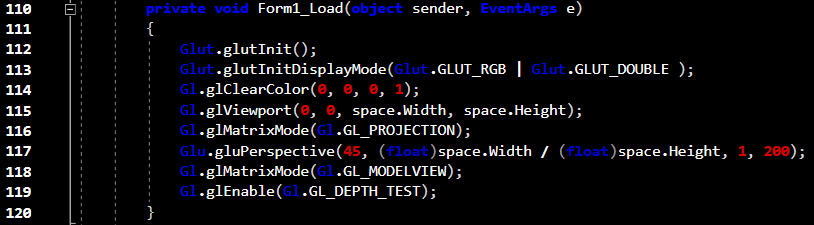


Рис. 2. Функция загрузки формы

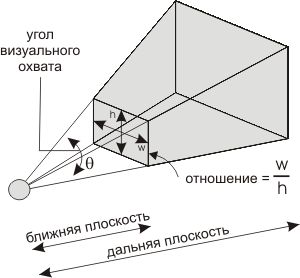


Рис. 3. Пирамида видимости

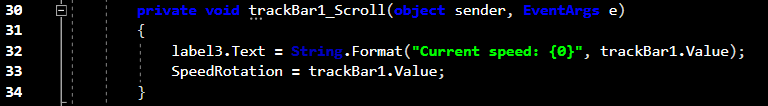


Рис. 4. Регулировка скорости



Рис. 5. Функция таймера

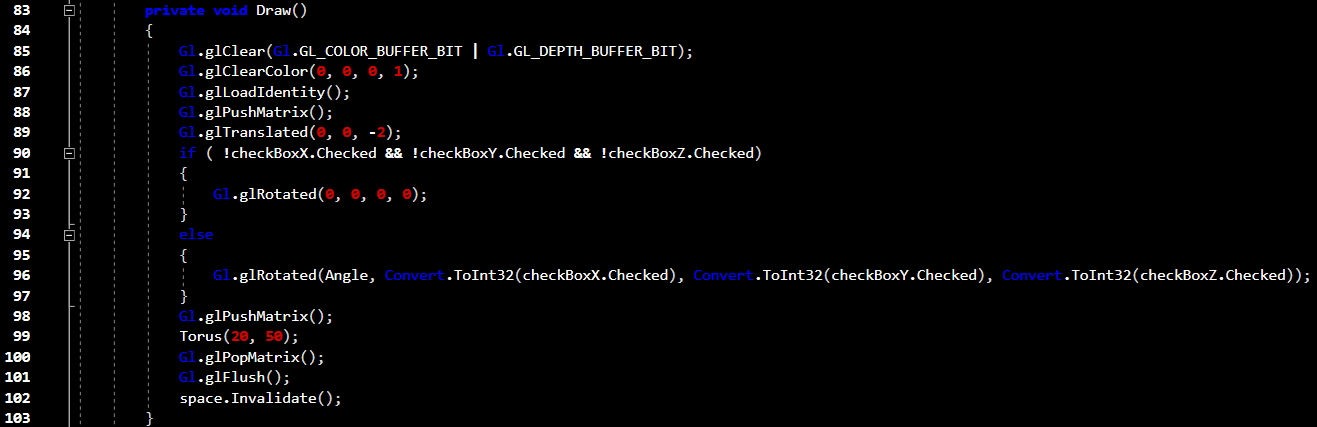


Рис. 6. Функция отрисовки



Рис. 7. Функция рисования тороида

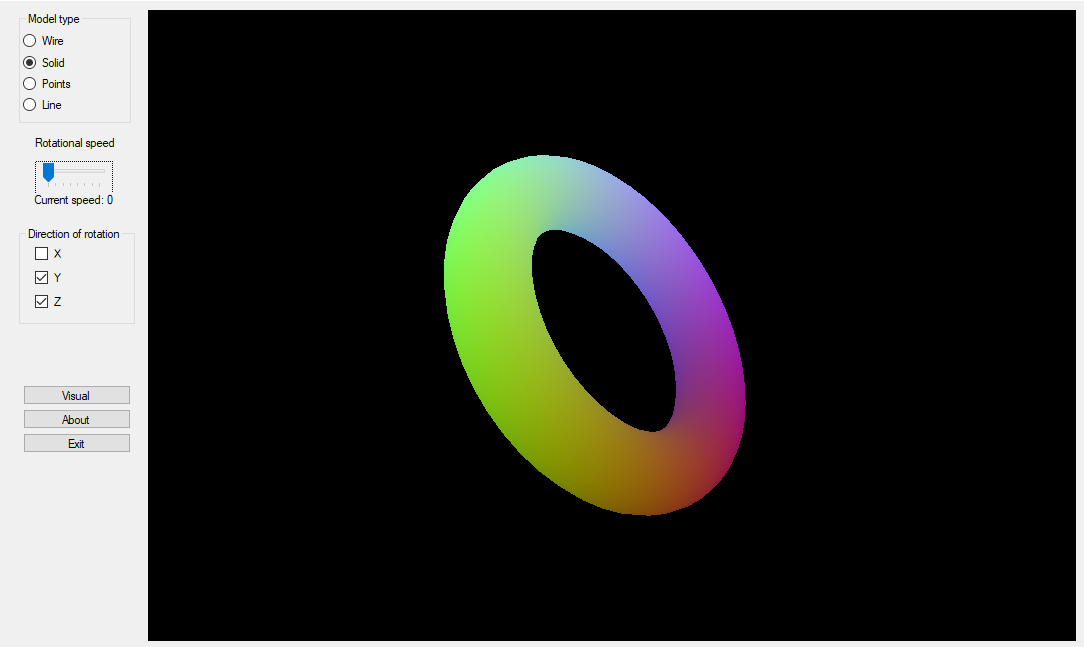


Рис. 8. Пример работающей программы