Московский Авиационный Институт (Национальный Исследовательский Университет)

Факультет информационных технологий и прикладной математики

Кафедра вычислительной математики и программирования

Лабораторная работа №7 по курсу “Компьютерная Графика”

Студент: А. Т. Бахарев

Преподаватель: Г. С. Филиппов

Группа: М8О-306Б

Дата: 25.12.2018

Оценка:

Подпись:

# Лабораторная работа №7

**Тема:** Построение плоских полиномиальных кривых.

**Задача:** Написать программу, строящую полиномиальную кривую по заданным точкам. Обеспечить возможность изменения позиции точек и, при необходимости, значений касательных векторов и натяжения

**Вариант:** Cardinal spline (фундаментальная кривая) 3-й степени из двух сегментов (выполнить два варианта – с использованием стандартной функции рисования кривой и без). Предусмотреть изменение натяжений

Исходный код

**main.cpp**

#include <QApplication>

#include <QMainWindow>

#include "MyWidget.hpp"

#include <QSurfaceFormat>

#include <QVBoxLayout>

#include <QHBoxLayout>

#include <QGroupBox>

#include <QtWidgets>

#include <QGLWidget>

**int** main(**int** argc, **char** \*argv[])

{

QApplication app(argc, argv);

QSurfaceFormat format;

format.setDepthBufferSize(24);

QSurfaceFormat::setDefaultFormat(format);

app.setApplicationName("");

app.setApplicationVersion("");

MyWidget widget{};

**auto** qvbox = **new** QVBoxLayout{};

qvbox->addWidget(&widget);

**auto** groupbox = **new** QGroupBox{QObject::tr("")};

groupbox->setLayout(qvbox);

QMainWindow window{};

window.setCentralWidget(groupbox);

window.resize(QSize(800, 600));

window.show();

**return** app.exec();

}

**MyWidget.hpp**

#ifndef MYWIDGET\_HPP

#define MYWIDGET\_HPP

#include <QVector2D>

#include <QChart>

#include <QChartView>

#include <QLineSeries>

#include <utility>

#include <vector>

#include <QWidgets>

#include "SettingsWidget.hpp"

**class** MyWidget : **public** QWidget

{

Q\_OBJECT

**public**:

**explicit** MyWidget(QWidget \*parent = **nullptr**);

**virtual** ~MyWidget() = **default**;

**protected**:

**void** keyPressEvent(QKeyEvent \*event) override;

**private**:

**double** begin = 0;

**double** end = 2;

std::vector<QVector2D> splinePoints;

std::vector<QVector2D> splineTangents;

QtCharts::QChartView \*chartView;

**void** draw();

std::pair<SettingsWidget \*, SettingsWidget \*> makeLines();

};

#endif // MYWIDGET\_HPP

# Выводы

Выполнив 7ую лабораторную работу по курсу Компьютерная Графика, я научился строить полиномиальные кривые по заданным точкам. Изучил на практике построение фундаментальной кривой(Cardinal Spline).