**САНКТ-ПЕТЕРБУРГСКИЙ ГОСУДАРСТВЕННЫЙ**

**ПОЛИТЕХНИЧЕСКИЙ УНИВЕРСИТЕТ ПЕТРА ВЕЛИКОГО**

**Институт компьютерных наук и технологий**

**Кафедра компьютерных систем и программных технологий**

**ОТЧЕТ**

**по курсовой работе на тему**

**«QT. GUI-приложение. Реализация игры**

**“Жизнь” »**

**Дисциплина "Программирование"**

**Студент гр.13501/4 Степин А.С.**

**Преподаватель Глухих М.И.**

**Санкт-Петербург**

**2015**

Оглавление

[Описание задачи 2](#_Toc420972108)

[Текст и описание программы 2](#_Toc420972109)

[**Тестирование 5**](#_Toc420972110)

[Заключение 8](#_Toc420972111)

# Описание задачи

Реализация игры «Жизнь» со стандартными правилами в режиме пользователь-компьютер.

Место действия этой игры — «вселенная» — это размеченная на клетки поверхность или плоскость — безграничная, ограниченная, или замкнутая (в пределе — бесконечная плоскость).

Каждая клетка на этой поверхности может находиться в двух состояниях: быть «живой» или быть «мёртвой» (пустой). Клетка имеет восемь соседей (окружающих клеток).

Распределение живых клеток в начале игры называется первым поколением. Каждое следующее поколение рассчитывается на основе предыдущего по таким правилам:

в пустой (мёртвой) клетке, рядом с которой ровно три живые клетки, зарождается жизнь;

если у живой клетки есть две или три живые соседки, то эта клетка продолжает жить; в противном случае (если соседей меньше двух или больше трёх) клетка умирает («от одиночества» или «от перенаселённости»)

Игра прекращается, если на поле не останется ни одной «живой» клетки, если при очередном шаге ни одна из клеток не меняет своего состояния (складывается стабильная конфигурация) или если конфигурация на очередном шаге в точности (без сдвигов и поворотов) повторит себя же на одном из более ранних шагов (складывается периодическая конфигурация).

Эти простые правила приводят к огромному разнообразию форм, которые могут возникнуть в игре.

Игрок не принимает прямого участия в игре, а лишь расставляет или генерирует начальную конфигурацию «живых» клеток, которые затем взаимодействуют согласно правилам уже без его участия (он является наблюдателем)..

# Текст и описание программы

private slots:

void on\_tableWidget\_cellClicked(int row, int column);//Выделение живых клеток

void on\_ClearButton\_clicked();//Очистка поля

public slots:

void StartButton();//Старт игры

void StopButton();// Остановка игры

void Create\_Future\_Field();//Создание массива следующего поколения

void saveGame();//Сохронение текущего поля

void loadGame();//Загрузка сохронённого поля

private:

Ui::MainWindow \*ui;

QChar\*\* now;//Массив текущего поколения

QChar\*\* future;//Массив следующего поколения

QTimer\* timer;//Таймер

int lab;//Номер поколения

QFile file;

//void Create\_Future\_Field();

void Check\_Cell(int row, int col,int n);//Определение жизни клетки для следующего поколения

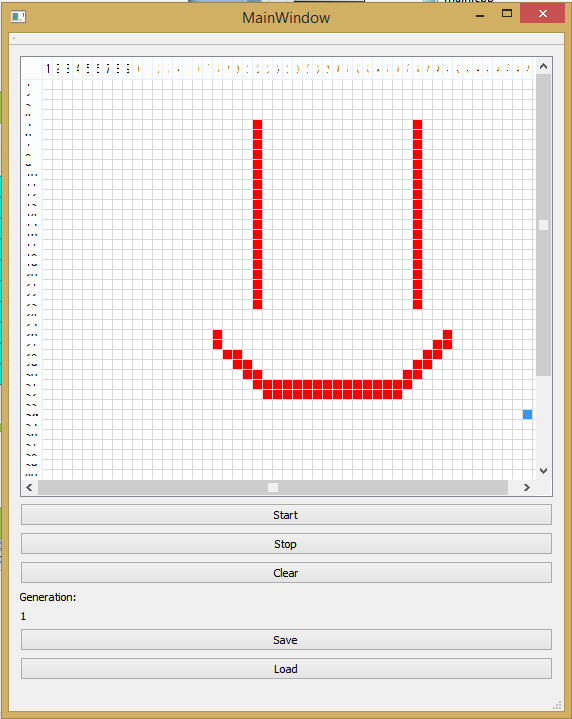
void Counter\_Cell(int row, int col);//Подсчёт живых клеток вокруг

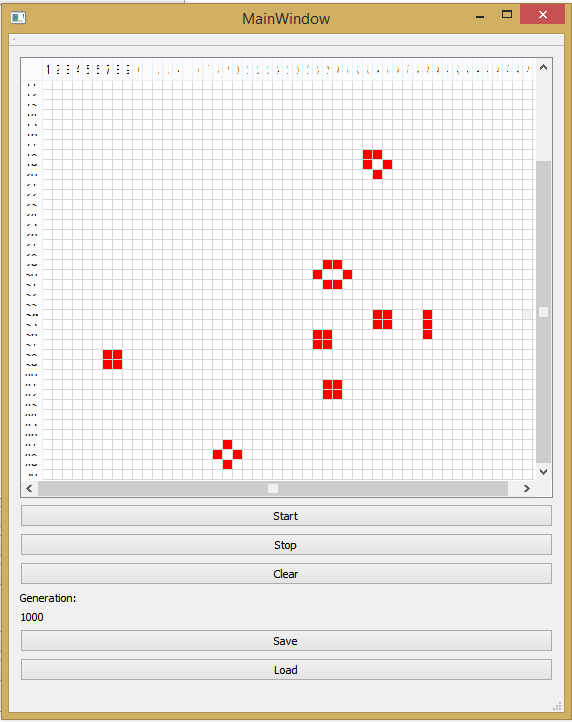
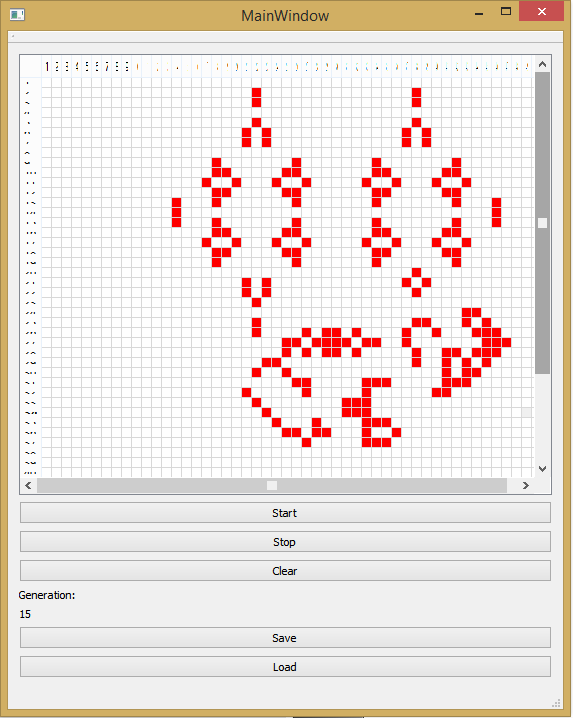
void Copy\_Future\_Field\_in\_Now\_Field();//Передвижение времени

void Watch( );//Показ на экране

Класс реализует все действия игры от внутреигровых до графического интерфейса. Поле в игре реализовано в виде массива и в каждой клетке жизнь протекает в зависимости от соседних клеток.   
Пользователь имеет возможность выбирать расположение живых клеток в начале и менять их в течении модуляции.

# Тестирование





# Заключение

Была достигнута основная цель: реализовать игру «Жизнь». При этом имеются перспективы на дальнейшее развитие: изменение таймера, реализация бесконечного поля(или любые её модификации), реализация полов, старение клеток, выбор цвета клеток.