# Лабораторная работа 2: Масштабирование интерфейса для больших массивов данных.

Задача: модифицировать программу, полученную в лабораторной 1 следующим образом:

1. Количество квадратов (ранее было 9) сделать динамическим и зависящим от параметров, считываемых из JSON-файла.

2. В json — файле обновить структуру. Список с данными *data*, находящийся в корне файла из предыдущей лабораторной работы, помещен в массив следующей структуры:

{

"width": 3,

"height": 3,

"frames": 2,

"animation": *data*

}

где параметры width, height отвечают за число строк и столбцов в результирующей экранной форме, параметр frames отвечает за число кадров анимации.

3. Модифицировать визуализацию данных с использованием паттерна mvc, где данные в анимацию поступают через model в gridView.

Информация по моделям в qml (оригинальная документация): <http://doc-snapshots.qt.io/qt5-5.11/qtquick-modelviewsdata-modelview.html>

Информация по моделям в qml (уроки на Русском языке): <https://habr.com/post/181712/>

Модели на c++: http://doc-snapshots.qt.io/qt5-5.11/qtquick-modelviewsdata-cppmodels.html

После выполнения шага 2 (перед внедрением mvc), сохранить старую версию программы.

4. Оценить быстродействие работы старой и новой программ на примере больших файлов с анимациями. Файлы доступны по ссылке [https://yadi.sk/d/TE8kV4T0qYrUkg](https://yadi.sk/d/gsNSymZ6Aea5lw). Оценивать следует по двум параметрам: скорости запуска (мсек) и времени задержки одного цикла анимации, вычисленного по формуле: (*x*/*f*-100), где *x* — фактическое время, понадобившееся программе для одного цикла анимации, мсек, *f* — число кадров.

5. Данные быстродействия занести в таблицу ниже:

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Имя файла** | **Число пикселей** | **Число кадров** | **Время запуска, мсек** | **Задержка анимации, мсек** |
| *Filename1.json* | *100* | *100* | *100* | *100* |
| *...* | *...* | *...* | *...* | *...* |