

## 第一周大作业设计文档

2018011347 肖易佳 计 83

### 实验目的：

通过实现数字微流控生物芯片模拟界面，复习巩固 Qt 中的基本窗口调用、绘图、字符串、文件等

### 设计思路：

数据与图形界面封装为两个类，实现数据与界面分离。

### 数据部分：

液滴的数据与功能通过 waterdrop 里的 Drop 类实现，

网格由 grid 部分的 Grid 类实现

窗口 MainWindow 为程序输入、处理与输出的核心部分，manipulation

类：

`class Point`

`int x, y;` 坐标位置的封装

`class Drop`

`Point pos;` `int type;`

`bool operated, longing, shuZhi;`

`pos` 存储液滴位置，`type` 记录液滴种类（记录污染和上色需要用到）

`operated` 用于记录液滴是否在当前时刻被操作，`longing` 记录液滴是否正在伸长，

`shuZhi` 判别液滴伸长的方向是竖直方向还是水平方向

`class DropVector`

`vector<Drop> dropvector;` 存储某一时刻的所有液滴

### 基本思路

Global variables:

`Current Time, Time Limit`

先处理好数据，之后根据时间需求，再更新界面。按照 `Current Time`，`repaint` 对应时刻的 `map` 情况即可

数据与绘图分离，先读入指令，之后先将指令标准化处理，指令有 6 种可能，“Input” “Output” “Move” 等，用字符串比较即可确定情况，根据情况处理数据。先考虑数据的范围，由于规模不大，因此将所有的指令一次性读入，之后模拟液滴的行为，并且在每一个时刻，将棋棋盘的布局、污染的情况记录并且保存，最终可以得到 T 张图表，这样以来，一方面可以实现操作的指令，另一方面也实现了上一步、下一步的要求。

对于移动混合这类简单的操作，改变液滴所在的位置、更新格子的污染、当前液滴等数据即可；而对于合并、分裂，由于分列、合并的中间状态为椭圆状态，有两个可能的方向，所以引入一个 bool 变量（vertical）来解决这个问题。

关于静态约束，棋盘的范围最多不超过  $20 * 20$ ，所以利用暴力遍历搜索即可解决；而动态约束部分，再加上时间参数 0-100，复杂度也是可以接受的，则只需将上一时刻和这一时刻的所有液滴，进行遍历计算距离，判断是否符合要求即可

关于错误检查部分，端口位置、输入输出合法性，作图可得到相关的条件

全部播放的要求，则利用全局变量 cur\_time，记录当前时间，每走一步，cur\_time 加一，刷新界面即可，循环实现；复位则将时间设置为 0，刷新即可；上一步、下一步则分别将时间+1、-1。

每一个格子的污染次数，则和液滴的种类（drops 类加入了液滴种类的属性）相关联。每当一个液滴到一个新的位置，就会判断此种类液体是否曾经污染过该格子，如果污染过，那么污染次数不做变化；否则污染次数加一。

关于音效功能，则根据四类指令调用相应的音频即可。

### 关键函数

`void MainWindow::manipulation()`

此函数是工程的主体，读取指令文件，并且将对应的 map 数据初始化，根据相应的指令操作对应的格子、液滴的数据，并且产生每一时刻的棋盘状态的二维表格。总而言之，该函数主要处理内部的数据，是项目内核

`void MainWindow::paintEvent(QPaintEvent *)`

重载绘图函数，用于将数据可视化，即显示出对应的格子以及液滴。

`bool MainWindow::isValid()`

判断约束条件是否满足，函数和 manipulation 相关，用于判断操作是否合理。

The End...

感谢阅读！