

Qt 第一周——数独大作业

一、 工程简介

数独是一款简单但令人着迷的益智游戏，基本要求如下：在每个小方格内填上数字 1~9 使其满足：每行，每列和每个小的九宫格内都包含数字 1~9，但不能重复。为了让游戏者更加方便的进行游戏，我使用 Qt 来创建并绘制了图形界面，完成了基本的游戏逻辑，并且加入了许多实用的功能，使游戏者能更直观的完成数独游戏。

二、 功能介绍

基本功能：

1. 实现了标准数独 9×9 的布局，以鼠标左键点击进行响应，可以用鼠标点选格子。对选中方格进行填数，从下方按钮中选中数字填入选中方格中。
2. 可以在一个格子内填入多个数字，作为备选项。
3. 点击数字会高亮当前界面所有相同数字。
4. 实现了难度选择功能，共有十个难度可供选择。难度的评价标准是数独中空格数目的多少，空格越多难度越大。
5. 实现了删除功能，删除当前格内所有数字。实现了撤销与恢复功能，可以撤销、恢复十步以内的操作，包括删除与恢复。
6. 实现了计时功能，可以实时看到游戏者的用时，在胜利时提醒游戏者。
7. 实现了暂停功能，可以暂停游戏。实现了重新开始的功能。

额外功能：

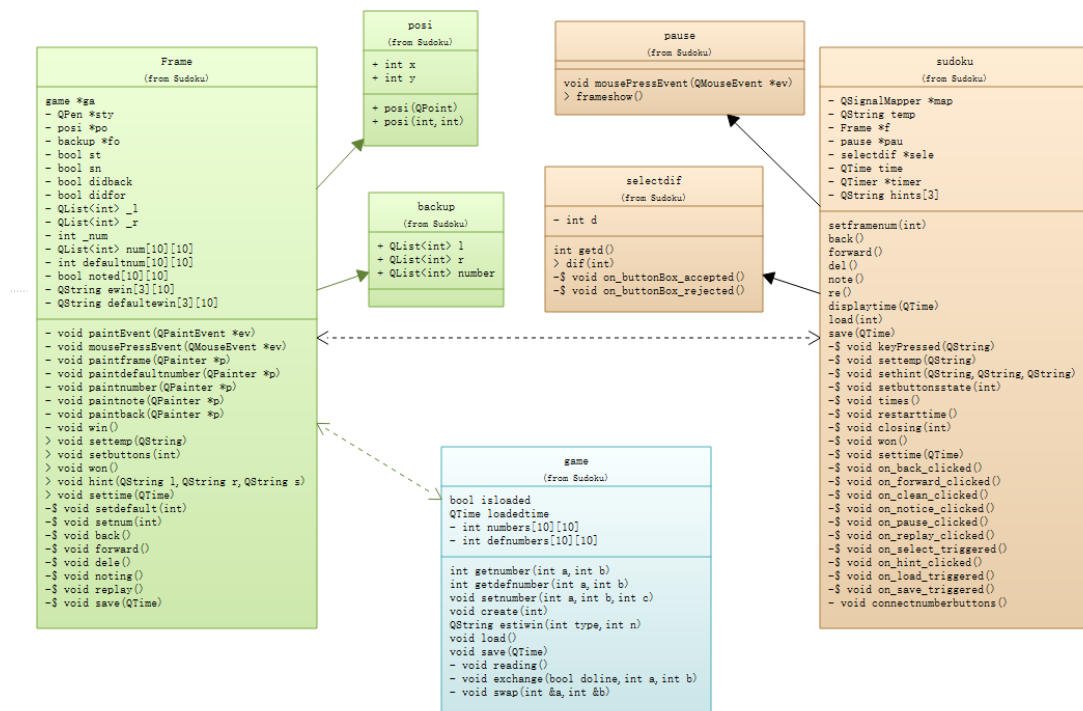
1. 加入了标记功能，可以标记格子或清除格子中的标记。
2. 加入了定位功能，点击后可以定位点击位置。
3. 加入了完成判断功能，若完成了某行、列、九宫格，就会在背景标记出来。
4. 加入了初级随机数独生成器，根据设置的难度随机生成不同难度的初始数独。
5. 加入了初级的数独求解器，用以给出提示。
6. 加入了储存读取功能，可以自定义储存、读取数独。

三、代码实现

1、程序结构

我将这个数独程序主要分为三个部分，一个是主界面 sudoku

类，负责与游戏者交互，其中包含了两个弹出窗口 pause 类和



selectdif 类；一个是主画布 frame 类，负责处理绘画数独界面，其中包含了确定位置的 posi 类和负责储存恢复信息的 backup 类；还有一个是背后的逻辑类 game，负责生成新数独，同时进行胜利判断。

2、 数独的生成

数独的生成有两种方法，一种是根据难度生成。game 类从一个 sample 文件中读取一个基础数独阵，并在保证正确的前提下随机交换行列，然后根据用户决定的难度等级，挖去一定数量的数字来提升难度，难度越高挖的数字越多。还有一种是读取现成的数独文件，包括默认数字，总数字与所用时间。生成的数独以数组的形式储存在 game 类的 defnumbers 与 numbers 中。



3、 行列宫的判断

在与主程序的交互中，game 类实时更新 numbers 的信息，依次通过 estiwin 函数判断当前行、列、及九宫格的完成度，将其储存在一个 QString 中返回。行以 l 开头，列 r，宫 s，若该行可以填入 1 则在 l 后加入 1，有错则加入 m 并返回。

4、 数独画布实现

在创建数独的主画布时，我并没有使用 81 个按钮或者 label，而是用一整块 QFrame 作为画布，用 QPainter 在上面画出格子、标记与数字。我创建了一个 posi 类用来将具体的像素坐标映射到格子的行列上，通过寻找固定的像素位置来绘制，通过循环语句来判断是否绘制。我首先依据 estiwin 判断的结果绘制背景，如果该行、列、宫完成了，就画上背景。然后我根据是否有标记来在不同格子中绘制标记。然后我绘制格子，并根据当前点击的位置将当前行列的格子颜色改为红色。然后根据 defaultnumber 中的数据绘制默认数字，再根据 num 中的信息绘制数字。根据当前点击的数字，改变与当前点击数字相同的数字的颜色。如一个格子内有多个数字，则改变绘制方式，在格子内绘制四个小字。

<u>4</u>	<u>5</u>	<u>8</u>	<u>2</u>	<u>3</u>	<u>6</u>	9	<u>7</u>	<u>1</u>
<u>3</u>		<u>7</u>		<u>9</u>	<u>8</u>	<u>5</u>	<u>6</u>	<u>4</u>
<u>9</u>		<u>6</u>	<u>7</u>	<u>4</u>			<u>3</u>	
<u>8</u>	<u>7</u>	<u>3</u>		1	<u>2</u>		<u>5</u>	
	<u>9</u>	^{3 6}	<u>6</u>	<u>7</u>		<u>1</u>	<u>8</u>	<u>3</u>
	<u>4</u>	<u>1</u>	<u>5</u>	<u>8</u>	2	<u>7</u>		<u>9</u>
	<u>8</u>		<u>4</u>		<u>7</u>	<u>6</u>	<u>4</u>	<u>2</u>
			<u>8</u>	<u>2</u>		<u>3</u>	<u>1</u>	<u>5</u>
<u>5</u>	<u>3</u>	<u>2</u>			<u>1</u>	<u>8</u>		

5、 撤销与恢复的实现

我运用 QList 储存行、列与数字信息，当有操作发生时将新动作储存在 QList 中，如果要撤销则去掉链表最后一个节点并重新绘制。我创建了一个 backup 类，将去掉的节点放入 backup 中，当恢复时就从 backup 中提取数据。删除同理。当重新开始游戏时，我清空 QList 中的信息，并依据默认数字重新绘制。

6、 与主界面的交互

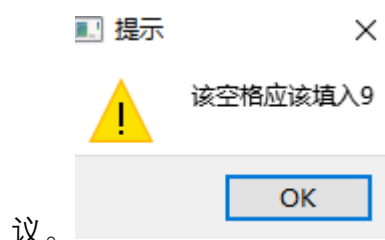
当玩家点击画布时，出发 mousePressEvent 向主界面发出信号，传递当前位置的数字信息。主界面根据这个数字信息调整按钮情况。如若格子中有 1,2 两个数字，则 1,2 数字按钮高亮。当主界面的按钮被按下时，改变的数字信息再传回 Frame 类中，主画布更新信息并重绘。若每个行列宫都完成了则胜利。

7、 计时器

我在程序中加入了计时器，同时在读取数独的时候也能读取时间信息，从上一次的时间继续游戏。游戏中可以暂停，调用一个空的 QFrame 窗口覆盖掉主画布并暂停计时，点击重新显示画布。

8、 提示功能

主界面除了基础按钮，我还实现了提示功能。根据每格判断 esitwin 信息，如果有 m 则有错，如果行列宫中都有 9 则提出建



9、 储存读取

我将当前数独的数字、默认数字和当前时间信息以一定格式写入 txt 文件中。读取同理。

四、 反思展望

这次大作业虽然基本完成，但是这个程序仍有许多可以完善的地方。首先是程序的 ui 不够美观，可以加入图片、按钮图标、音乐等丰富程序。其次是在数独生成的时候没有考虑是否有多解或无解的情况。我初步考虑运用 DFS，在交换挖空后的数独中进行 estiwin 判断，寻找仅有一个相同数字的格子，并填入该数字。然后再依据填入数字后的数独再次重复 estiwin 判断，不断填入新数字。最后生成数独，若无法完成则无解或有多解，放弃并重新生成新数独。求解器同理。