

基于 QT 的俄罗斯方块游戏

叶丹 白俊鹤

成都锦城学院计算机与软件学院 四川 成都 611731

【摘要】俄罗斯方块是一款较为经典的休闲益智游戏，通过键盘控制方块的形状和位置，使方块填满一行以达到消除效果。虽然当今市场上不断出现各式各样的游戏，但是俄罗斯方块仍具有举足轻重的作用。本文主要阐述了基于 QT 和 C++ 技术在 Windows 开发环境下的一款俄罗斯方块游戏，将会简单介绍该游戏的历史，并详细描述其设计思路、界面布局以及部分功能的实现。

【关键词】QT; 俄罗斯方块; 游戏编程

1 引言

就现在较繁重的生活和工作压力之下，大多数人们在平时的闲暇时光，总是会选择玩一些益智休闲的小游戏来放松头脑，俄罗斯方块游戏正是一款符合条件的趣味性小游戏。评选史上最多平台的游戏，《俄罗斯方块》绝对有一席之地，三十年来横跨无数平台，但凡有游戏功能就少不了他的存在 [1]。俄罗斯方块游戏是在 1984 年由苏联科学院的一名普通软件工程师开发的一款电脑游戏，主要游戏方式是通过将不同形状的方块进行位置移动和形状变换，以此铺满一格或多格实现方块的消除。最主要的是该游戏符合“整理心理学”，即人们需要通过下落的新的方块来整理已有的混乱局面 [2]。该游戏规则简单，易于上手，特别适合人们平常消磨时间时玩的。本文将主要介绍俄罗斯方块游戏的设计流程、界面布局以及如何使用 QT 应用程序来对俄罗斯方块游戏进行开

发。

2 设计思想

首先游戏运行后其界面是静止的，点击左上方菜单栏，选择“游戏”按钮，点击“开始游戏”按钮，此时游戏正式开始。根据分析，“俄罗斯方块游戏”中的方块，共有七种基本形状，它们分别是“1 型”、“L 型”、“反 L 型”、“Z 型”、“反 Z 型”、“口型”、“T 型” [3]。每一种形状都由四块方格组成，游戏开始后在网格的顶部中央会随机出现这七种基本形状的某一种。玩家通过控制键盘的来对方块进行操作，当方块铺满一行就会自动消除，当落下的方格触碰网格顶部则游戏失败。暂停游戏和继续游戏放在菜单栏游戏选项中同一位置，两种状态可以自动切换。菜单栏“游戏”选项中还有一个按钮“结束游戏”，点击此按钮本局游戏结束。图 1 是俄罗斯方块游戏的设计流程图。

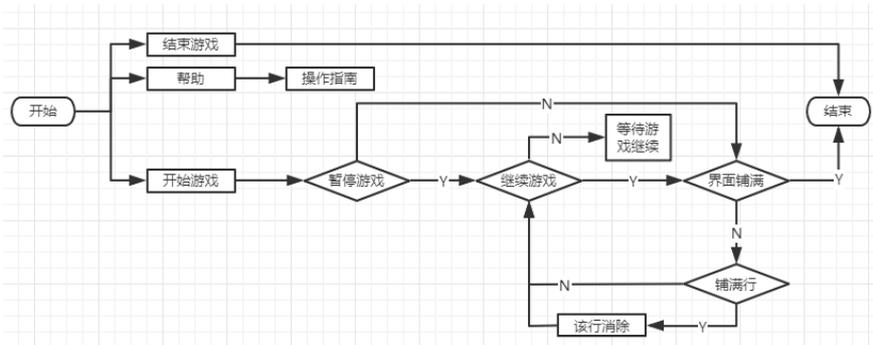


图 1 俄罗斯方块设计流程图

3 主要功能描述

本俄罗斯方块游戏的主要功能都放在了菜单栏里面，菜单栏分为“游戏”和“帮助”两个选项。“开始游戏”“暂停游戏”和“结束游戏”这三个功能都在“游戏”选项下面。游戏开始后点击“暂停游戏”按钮则游戏暂停，此时菜单栏中的“暂停游戏”按钮自动变为“继续游戏”按钮；点击则游戏继续，此时“继续游戏”按钮自动变为“暂停游戏”按钮。左边功能是主要显示游戏动态，方块随机生成形状，从网格顶部中央掉落，到网格底部停下，之后又重新生成新的方块。用户可以通过使用键盘 ASD 键，对方块进行左移、右移、下移这三种位置操作，以及通过使用空格键对方块进行四个方向的形状变换操

作。右边区域的功能主要是显示下一个将出现的方块形状、当前得分这两种关键信息。

4 界面设计

本俄罗斯方块游戏的界面设计较为简洁，只有一个主界面。顶部是黑色方块的标志，窗口名为“俄罗斯方块”。左上方工具栏包含“游戏”和“帮助”选项，“游戏”选项下面有“开始游戏”“暂停游戏”和“结束游戏”这三个按钮。“帮助”选项下只有“操作指南”按钮，点击按钮后将会有弹窗提示。左边是该程序的标志，右边分别纵向显示：左移 A 键、右移 D 键、下移 S 键、变形空格键等操作规则。游戏界面的左半边是一个网格

图，主要是显示游戏时玩家操作方块落下的状态，也就是主要的游戏界面。右边的界面是由上下两个控件组成，上方的控件显示的是下一个出现的方块形状，下方的控件显示当前得分。界面主基调是黑色，网格为白色，方块为黄色，数据面板为绿色。图2为俄罗斯方块游戏的界面效果图。

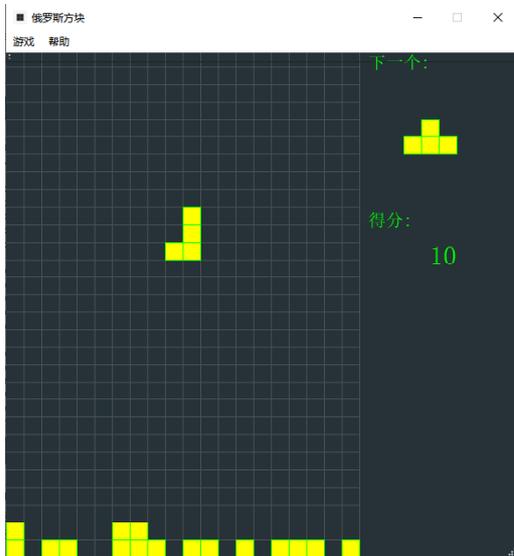


图2 俄罗斯方块游戏界面效果图

5 部分功能实现

5.1 主场景功能实现

主场景功能主要是在初始化函数 `init()` 中实现，该函数将包含了菜单初始化函数以及看板初始化函数，同时创建画笔和画刷。使用 `setWindowIcon()` 函数给窗口设置一个图标，使用 `setWindowTitle()` 函数将窗口名设置为“俄罗斯方块”。在游戏配置文件中固定了窗口高宽度、游戏窗口高宽度以及网格宽度。`initMenu()` 函数中，使用 `menuBar()->addMenu()` 语句将“游戏”和“帮助”选项添加到菜单栏中。其中二级选项中“开始游戏”“暂停游戏”“结束游戏”和“帮助”这四个按钮，运用到了信号和槽函数的关联来实现对事件的控制的知识。`initScoreBoard()` 函数中，使用 `setColor()` 语句将字体颜色置为绿色，设置下一个方块标签、分数标签以及分数值标签的文本内容字体颜色，还有这几个控件的几何结构。

5.2 旋转方块功能实现

上文曾介绍过俄罗斯方块具有“I型”、“L型”、“反L型”、“Z型”、“反Z型”、“口型”、“T型”这七种基本形状。我们可以通过分析得出，“L型”、“反L型”和“T型”具有上下左右四种旋转状态，可以使用 `switch` 语句分别描述上、下、左、右四种情况，再绘制二维数组中的形状。“Z型”、“反Z型”和“I型”因为其左右旋转状态一致以及上下旋转状态一致，所以只有横纵两种旋转状态，一个 `if` 语句判断是否为纵向，然后在二维数组中绘制其纵向的状态，`else` 语句中绘制横向的状态。其中“口型”上下左右形态一致，所以只有一种旋转状态。

5.3 悬空坠落功能实现

当一行或多行产生消除后，需要将消除行的上方

元素向下移动。若第 n 行被消除时，其上一行是第 $n-1$ 行，从第 $n-1$ 行到第一行，都需要依次直接拷贝其前一行的数据，来覆盖本行数据，此时第 0 行则生成一个空行。我们可以从网格底部开始遍历每行元素，设置一个整型 k 来记录每行的 0 值。若 k 的值等于该行的大小，也就是该行全为 0，则跳出循环不再遍历。若不全为 0，则 `blocks[x][j] = blocks[x - 1][j]`，也就是将上行数据拷贝给本行，实现悬空坠落功能。

5.4 碰撞判断功能实现

就是将当前俄罗斯方块的局势信息保存在一个二维数组中，当前方块的任何移动操作都需要先与残局中的数据进行对比，防止产生碰撞。首先用一个 `for` 循环实现对当前方块的四个方格位置的遍历，若 `if` 语句判断出当前方块的位置在二维数组的值为 1，则说明方块重合。碰撞检测除了需要检测方块之间的重合还需要考虑边界情况。可以使用两个 `if` 语句分别判断当前方块的位置，一个语句 `if` 判断是否靠在 x 边界，另一个语句 `if` 判断是否靠在 y 边界。

5.5 消除判断功能实现

当一行或多行产生消除后，需要判断此时是否形成了可消除行。判断是可消除行，则循环遍历该行将二维数组的该行数据置为 0。可消除行需要被方块填满，因此只有当前方块穿插的行，才有可能形成可消除行。其判断依据就是二维数组中任意一行元素是否含 0，如果包含则此时无可消除行。

6 功能测试及后期改进

该游戏完成后，我对俄罗斯方块游戏的基本功能进行了一系列的测试。在对游戏功能进行测试时，我发现该项目还存在一部分漏洞。第一个漏洞是在暂停游戏后，直接点击“开始游戏”按钮，游戏会重开，此时应该设置一个弹窗提醒。第二个漏洞是如果在点击“开始游戏”按钮之前点击“暂停游戏”按钮，程序会直接引发异常，这两个漏洞应该都是代码逻辑造成的问题。

后期对俄罗斯方块游戏有一些具体改进方案，首先是修复测试时遇到的漏洞，然后预计在菜单栏新增“设置”以及“模式”选项。计划在“设置”中加入“游戏难度”选项。难度分为高、中、低三个层次，玩家可以手动调整三个层次的数据。在点击开始游戏前，玩家可以根据自己的水平高低在设置中选择适应的游戏难度。“模式”选项中包含“经典模式”也就是现在的模式，以及“上推模式”。同时也计划连接数据库，实现保存游戏分数等信息的功能。

7 结束语

本文具体介绍了如何使用 QT 应用程序开发俄罗斯方块这款游戏，该项目可以实现对俄罗斯方块游戏的开始、暂停、继续、退出功能。其中游戏开始后，可以通过键盘对方块进行左右下的移动操作以及对四个方向的旋转操作，方块满行后自动进行消除操作。但相比较于其他更加完善的俄罗斯方块游戏，本俄罗斯方块游戏还存在一定的缺陷。例如：游戏画面缺乏设计感和层次感、无法手动设置游戏困难等级、游戏模式单一等问题，这些都可以作为下一阶段游戏完善和改进的地方。总的来说，

本文简单介绍的俄罗斯方块游戏部分代码逻辑，易于理解且此项目基本功能较为完善，适合初学者学习使用。

【参考文献】

[1] 刘峰国. 漫谈经典游戏系列的起源 [J]. 计算机与网络, 2015, 41(17):20-21.

[2] 赵素萍. 基于 C 语言的俄罗斯方块游戏的设计与实现 [J]. 电子技术与软件工程, 2020(09):32-33.

[3] 邱菊, 唐凯军, 徐立希, 牛小梅. 基于枚举算法设计“俄罗斯方块”游戏 [J]. 电脑知识与技术, 2009, 5(14):3710-3712.