

文章编号: 2095-2163(2019)04-0212-04

中图分类号: TP317

文献标志码: A

高级打地鼠游戏的设计与实现

商利华

(无锡城市职业技术学院, 江苏 无锡 214153)

摘要: 本文基于 C# 开发环境, 开发了一款有趣味、有挑战、画面清新生动的高级打地鼠游戏。本游戏是通过地鼠的不断闪现和敲击, 根据玩家的击中数, 从而计算分数。其中, 最大的特色就是通过动态增加控件使得“田地”数目不断变化, 使得游戏等级逐步提高, 从而实现闯关、游戏计时计分等功能。基本此, 本文详尽提出动态增加控件的设计与实现, 同时对该游戏中的主界面设计和随机显示地鼠也进行了探讨与分析。

关键词: 打地鼠; C#; 随机数

Design and implementation of the advanced game of whack-a-mole

SHANG Lihua

(Wuxi City College of Vocational Technology, Wuxi Jiangsu 214153, China)

[Abstract] Based on the C # development environment, a whack-a-mole is developed which is fun, challenging, fresh and lively. While the game runs, through the constant flashing and tapping of the hamster, the score is calculated based on the number of hits by the player and the scores are gotten. The distinctive feature is that the number of "fields" is constantly changing by dynamically adding controls, game level is gradually improved and the functions such as game breakthrough, game timing and score are realized. This paper elaborates on the design and implementation of dynamically adding controls, and the main interface design and random display of the hamster in the game are also discussed and analyzed.

[Key words] whack-a-mole; C#; random number

0 引言

高级打地鼠是一款将休闲、趣味、益智精华完美融于其中的经典动作游戏。目前, 各平台上均已推出多种打地鼠游戏, 但游戏的质量参差不齐, 游戏模式单一, 而且也不具备各自的设计特色。基于此, 本文即使用 C# 语言和 GDI+ 技术共同研发一款高级打地鼠游戏。在此研发过程中, 将动态增加控件引入该款游戏中, 使得本款游戏不仅有着精美的画面、新颖的特效, 而且还有着十足引人入胜的挑战性和趣味性。本文拟对此展开研究论述如下。

1 设计思路

打地鼠游戏规则可表述为: 在游戏页面点击开始后, 即可直接进入游戏。游戏中, 地鼠会从一个个地洞中不经意地探出一个个脑袋, 企图躲过玩家的视线, 此时玩家出手要快、落锤要准, 力求击中地鼠次数能达到最多。本项目中, 研究使用多个按钮来模拟多个地鼠可能出现的位置, 通过按钮上显示图片的改变来反映地鼠的位置, 地鼠的位置按某个频率在一定的范围内随机出现。游戏可由 1~2 个玩

家合作使用技能消灭地洞里的地鼠, 除此之外游戏还可以计分, 达到一定的分数可以升级进入下一关, 每轮游戏都是要求在限定的时间内, 敲打的地鼠越多, 分数越高, 最后计算得到的总分。总分越高, 位列排行榜单的排名也会越高。游戏部分效果如图 1 所示。



图 1 高级打地鼠游戏界面、多地鼠设置、游戏不同关卡

Fig. 1 Advanced whack-a-mole game interface, multiple whack-a-mole settings and different levels of the game

作者简介: 商利华(1979-), 女, 讲师, 主要研究方向: .NET 平台开发、数据库应用。

收稿日期: 2019-05-18

2 游戏实现

2.1 制作打地鼠游戏闪屏 LOGO 和主页面静态界面

制作闪屏,将图片文件拷贝到 Debug 目录下,在程序中将此画面作为启动屏幕背景,如图 2 所示。设置 2 个 Boolean 型全局变量,分别作为不允许降低不透明度和允许增加不透明度。在窗体中拖放一个 Timer 控件,并将其 Enabled 设置为 True,在 timer1_Tick 事件处理程序的相关代码如下。

```
private void timer1_Tick(object sender,
EventArgs e)
{
    if (bAddAlpha)
        if (this.Opacity < 1.0)
        {
            this.Opacity += 0.1;
        }
        else
        {
            bAddAlpha = false;
            bSubAlpha = true;
        }
    if (bSubAlpha)
        if (this.Opacity > 0.1)
        {
            this.Opacity -= 0.1;
        }
        else
        {
            bSubAlpha = false;
            this.timer1.Enabled = false;
            new FormMain().Show();
        }
}
```

该段程序运行后,就实现了设计中的闪屏效果。



图 2 高级打地鼠游戏 LOGO

Fig. 2 Advanced whack-a-mole game LOGO

其次,在窗体上添加 3 个 Panel 控件,自上而下分别命名为:pnlTop、pnlBody 和 pnlBottom,其 Dock 属性值依次为:Top、Fill 和 Bottom。在 pnlTop 上添加 3 个 Label 控件,Name 属性依次为:lblLeftTime(显示游戏剩余时间)、lblClickNum(显示鼠标点击

数)、lblHitNum(显示击中地鼠数)和 lblLevel(显示当前关数)。在 pnlBottom 上添加一个 Button 控件,Name 属性设置为 btnStart。运行程序,即可得到如图 1 所示的设计效果。

2.2 类的继承的设计实现

本次项目是用按钮控件来模拟“田地”,程序中需要记录某一时刻田地是否有地鼠出现的状态,为此可利用 Button 控件的 Tag 属性来进行控制:当 Tag = 0 时无地鼠,Tag = 1 时有地鼠。本次任务中,则将继承 Button 类,通过在派生类里增加新的属性的方式来记录某个时刻田地是否有地鼠的状态。研发中,会用到如下程序代码。

```
class ClsButtonX: Button
{
    bool isMouse;
    public bool IsMouse
    {
        get { return isMouse; }
        set { isMouse = value; }
    }
}
```

2.3 动态增加控件

随着游戏等级的提高,游戏中的“田地”数也要逐渐增多,这样就涉及到在游戏过程中,升级后所派生出的按钮控件,就要按一定数量动态地添加到窗体中。程序界面布局如图 3 所示。由图 3 可知,文本框中输入每行显示的 Button 数,点击“按要求重新添加控件”按钮将添加指定数量的 Button 控件至窗体上。给按钮控件添加的与此功能相关的 Click 事件,其程序代码详见如下。



图 3 高级打地鼠游戏动态添加控件

Fig. 3 Dynamical adding controls in advanced game of whack-a-mole

```
private void btnStart_Click ( object sender,
EventArgs e)
{
    guanNum = Int32.Parse(tbNum.Text.
ToString());
```

```

    cellSize = pnlBody.Width / (guanNum);
    // 老鼠的尺寸
    pnlBody.Controls.Clear();
    // 先清除 pnlBody 上原有的控件
    for (int i = 0; i < guanNum; i++)
        for (int j = 0; j < guanNum; j++)
        {
            Button btnMouse = new Button();
            // 初始化 Button 的对象
            pnlBody.Controls.Add(btnMouse);
            // 添加控件到 pnlBody 容器上
            btnMouse.Width = btnMouse.Height =
cellSize;
            btnMouse.Top = j * cellSize;
            btnMouse.Left = i * cellSize;
            // 指定 btnMouse 的大小及显示位置
        }
    // 生成新的游戏界面(动态增加控件)
private void GenMouseCell()
{
    pnlBody.Controls.Clear(); // 先清除 pnlBody 上
原有的控件
    for (int i = 0; i <= guanNum; i++)
        for (int j = 0; j <= guanNum; j++)
        {
            ClsButtonX btnMouse = new ClsButtonX();
            // 初始化 ClsButtonX 的对象
            pnlBody.Controls.Add(btnMouse);
            // 添加控件到 pnlBody 容器上
            btnMouse.Width = btnMouse.Height =
cellSize;
            btnMouse.Top = j * cellSize;
            btnMouse.Left = i * cellSize;
            // 指定 btnMouse 的大小及显示位置
            btnMouse.IsMouse = false;
            // 默认为非老鼠状态
            btnMouse.Click += new EventHandler(button
X_Click);
            // 添加自定义事件
        }
}

```

2.4 随机显示老鼠

打老鼠游戏难点之一就是老鼠将会在指定的范

围之内随机出现,本项目将在按钮控件上显示老鼠图片的方式来标识老鼠的位置。首先,要产生老鼠随机位置,则要初始化一个随机类的实例,再在指定的范围内产生随机数。本项目引入了游戏等级的要求,而随着游戏等级的变化,即使得“田地”数目也将随之变化,故随机数的产生范围也并非固定不变。接下来,就要研究老鼠出现的时间间隔。采用 Timer 控件 *tmrHamster*, 初始将该控件属性设置为 300。最后用图片的方式显示老鼠。研究中,可以利用 Button 控件的 *image* 属性来显示图片,本项目中动态添加的控件是从 Button 类中派生出来,因而该控件也具有 *image* 属性。添加 *tmrHamster* 控件 Tick 事件的代码具体如下。

```

private void tmrHamster_Tick(object sender,
EventArgs e)
{
    ((ClsButtonX)pnlBody.Controls[rdNum]).Image
= null;
    ((ClsButtonX)pnlBody.Controls[rdNum]).
IsMouse = false;
    // 清除原来老鼠标记
    rdNum = rnd.Next((guanNum + 1) *
(guanNum + 1));
    // 产生新的老鼠的位置
    ((ClsButtonX)pnlBody.Controls[rdNum]).
Image = imageList1.Images[0];
    ((ClsButtonX)pnlBody.Controls[rdNum]).
IsMouse = true;
    // 新位置设置老鼠标记
}

```

2.5 增加游戏计时与积分

本项目在游戏计时的基础上增加游戏积分的功能,达到一定积分可以进入下一关继续游戏,达不到则继续本关游戏。首先添加一个 Timer 控件,命名为 *tmrTimeLeft*, 设置其 *Interval* 属性值为 1 000。其次,添加 Tick 事件,设计代码见如下。

```

// 游戏计时
private void tmrLeftTime_Tick(object sender,
EventArgs e)
{
    time--;
    lblLeftTime.Text = " 剩余时间: " + time.
ToString();
    progressBar1.Value = time;
}

```

```

//初始滚动条的 Value 属性
if (time == 0)
{
    if (hitMouse * 10 > ClickNum)
        //若击中地鼠数大于鼠标总点击数的 1/10
    则进入下一关
    {
        btnStart.Text = "继续下一关(&N)";
        guanNum ++;
        lblMarks.Text = "关数:" + guanNum.
ToString();
        cellSize = pnlBody.Width / (guanNum + 1);
        //更改“田地”的尺寸
    }
    else
        btnStart.Text = "继续本关(&N)";
        tmrLeftTime.Enabled = false;
        tmrHamster.Enabled = false;
        btnStart.Enabled = true;
        pnlBody.Visible = false;
    } }

```

为增加可视性,本次任务同时引入了 ProgressBar 控件计时。ProgressBar 控件是较长操作的状态的可视化表示,由其指示用户正在进行某个操作,用户应等待。ProgressBar 控件工作时要设置 Minimum 和 Maximum 属性。这些属性对应于进度指示器的最左端 (Minimum) 和最右端 (Maximum)。设置 Step 属性,以确定每次调用 PerformStep 方法时数值的增量。Value 属性返回 ProgressBar 的当前值。

(上接第 211 页)

控难度。拼多多可以构建一个物流配送网络或与已具备成熟物流网络的商业公司展开合作,共建物流网络,提升物流速度。

(3) 加强社交电商平台建设,强化平台监管力度。加强平台的建设可以进一步地完善平台售后服务、系统管理等功能,通过对平台管理人员进行技术培训等方式提高管理人员的素质。并加大平台的监管力度,强化商家入驻的审批流程,避免商品质量低劣、售卖假货等问题,提升平台的信誉,为消费者营造良好有序的消费环境。

参考文献

[1] 刘混尘,康星. 基于 3W2H 模型的拼多多商业模式研究[J]. 中

游戏结束弹出保存玩家姓名和分数的窗口,玩家输入姓名后点击确定,保存到数据库中,玩家可以通过排行榜分数查看从高到低排名的榜单。运行程序,效果如图 4 所示。

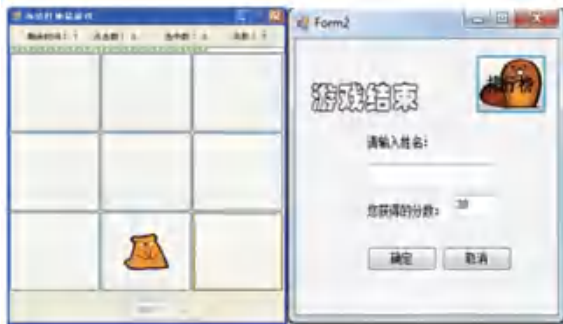


图 4 高级打地鼠游戏进度条设置、游戏排行榜

Fig. 4 Progress bar settings and game rankings of advanced game of whack-a-mole

3 结束语

本文开发的高级打地鼠游戏特色是地鼠多,关卡多,是一款可玩性颇高的电脑单机游戏。除实现基本功能外,还设置了不同的背景音乐和游戏音效,让玩家在轻松愉快的环境中体验打地鼠的快乐。

参考文献

- [1] 夏普,周靖. Visual C# 2005 从入门到精通[M]. 北京:清华大学出版社,2008.
- [2] 蔡继文. 21 天学通 ASP.NET[M]. 北京:电子工业出版社,2009.
- [3] 王兴晶. Visual Basic.NET 数据库开发典型实例[M]. 北京:电子工业出版社,2002.
- [4] 胡木林. 基于遗传算法的在线考试系统组卷设计[J]. 中国教育技术装备,2018(12):23-27.

国市场,2019(12):21-23.

- [2] 张倩丽. 拼多多营销模式分析[J]. 合作经济与科技,2019(3):100-101.
- [3] 黄婧. 互联网时代下社交电商的营销价值创造过程研究—以拼多多为例[J]. 商场现代化,2018(24):7-10.
- [4] 杨柳. 中国白酒业的核心竞争力研究[J]. 酿酒,2006,33(1):88-90.
- [5] 李紫轩,王萌萌,王娟. 互联网经济下企业商业模式转型研究—以娃哈哈集团为例[J/OL]. 电子商务:1-6[2018-12-17]. <http://kns.cnki.net/kcms/detail/11.4499.TN.20181217.1059.002.html>.
- [6] 余雪琼. 拼凑:拼多多的商业模式创新[J]. 财经界,2018(11):86-88.
- [7] 闫凯璐. 拼多多平台发展状况分析及对策研究[J]. 现代商业,2018(28):123-124.