
问题 1：怎么让进度条变颜色，而且不是只能从左往右长进度

解决方法：自定义一个进度条，继承自 `ProgressBar`，画一个矩形代表当前的进度，并根据进度给矩形填充不同的颜色，代码如下：

```
public class NewProgressBar : ProgressBar
{
    private Brush[] brushes = new Brush[] { Brushes.Red, Brushes.Yellow,
Brushes.Orange, Brushes.Lime, };
    public NewProgressBar()
    {
        this.SetStyle(ControlStyles.UserPaint, true);
    }

    protected override void OnPaint(PaintEventArgs e)
    {
        Rectangle rec = e.ClipRectangle;

        double rate = (double)Value / Maximum;
        rec.Width = (int)(rec.Width * rate) - 4;
        if (ProgressBarRenderer.IsSupported)
            ProgressBarRenderer.DrawHorizontalBar(e.Graphics, e.ClipRectangle);
        rec.Height = rec.Height - 4;
        if (rate >= 0.75)
            e.Graphics.FillRectangle(Brushes.Lime, 2, 2, rec.Width, rec.Height);
        else if (rate >= 0.5)
            e.Graphics.FillRectangle(Brushes.Yellow, 2, 2, rec.Width, rec.Height);
        else if (rate >= 0.25)
            e.Graphics.FillRectangle(Brushes.Orange, 2, 2, rec.Width, rec.Height);
        else
            e.Graphics.FillRectangle(Brushes.Red, 2, 2, rec.Width, rec.Height);
    }
}
```

问题 2：如何方便地给按钮添加背景图片并保证两两成对

解决方法：由于按钮数量很多，图片也很多，为了方便地根据下标选择图片，我不能直接把图片加载到资源 (Resources) 中，因为 Resources 中的图片只能根据名字读取，而名字用下标表示又不能区分水果的名字(个人的强迫症)，所以采用文件读取的方式获取图片，并存在一个 `Image` 数组中。为了保证图片两两成对出现，在使用了随机数的同时利用了 `Button` 的 `Visible` 属性，在设置背景图片之前 `Button` 的 `Visible` 设置为 `false`，设置背景图片之后将 `Visible` 设置为 `true`。代码如下：

```
int totalNum = n * n;
```

```

Random rnd = new Random();
while (totalNum > 0)
{
    int x = rnd.Next(n), y = rnd.Next(n), i = rnd.Next(m);
    Button btn = buttons[x, y];
    while (btn.Visible)
    {
        x = rnd.Next(n);
        y = rnd.Next(n);
        btn = buttons[x, y];
    }
    btn.BackgroundImage = images[i];
    btn.BackgroundImageLayout = ImageLayout.Zoom;
    btn.UseVisualStyleBackColor = true;
    btn.Visible = true;
    imageIndex[x, y] = i;
    while (btn.Visible)
    {
        x = rnd.Next(n);
        y = rnd.Next(n);
        btn = buttons[x, y];
    }
    btn.BackgroundImage = images[i];
    btn.BackgroundImageLayout = ImageLayout.Zoom;
    btn.UseVisualStyleBackColor = true;
    btn.Visible = true;
    imageIndex[x, y] = i;
    totalNum -= 2;
}

```

问题 3：如何判断两个图片可以消除

解决方法：为了方便处理边界上的图片，定义了一个 $(n+2)*(n+2)$ 的数组 `visible`，存储每个按钮是否已被消除， n 为每行每列图片的个数，周围多的一圈的 `visible` 都设为 `false`，表示已消除。首先判断两个图片是否为同一个，如果是同一个就不可达。然后依次判断两个图片是否直线可达、拐一个角可达、拐两个角可达。利用逻辑表达式的短路定理，只有直线不可达，才会判断是否拐一个角可达，只有拐一个角不可达，才会判断是否拐两个角可达。这样的话，拐一个角可达的条件是存在一个与第一个图片同行或同列的图片，它既与第一个图片直线可达，又与第二个图片直线可达(因为第一个图片和第二个图片直线不可达，所以这三个图片不可能共线)。同理拐两个角可达的条件是存在一个与第一个图片同行或同列的图片，它既与第一个图片直线可达，又与第二个图片拐一个角可达。代码实现了四个函数 `IsArrivable0(int x1, int y1, int x2,`

`int y2), IsArrivable1(int x1, int y1, int x2, int y2), IsArrivable2(int x1, int y1, int x2, int y2), IsArrivable (int x1, int y1, int x2, int y2)`，分别判断是否直线可达、拐一个角可达，拐两个角可达以及是否可以消除。

问题 4: SoundPlayer 不能同时播放背景音乐和游戏音效

解决方法：使用 `Microsoft.DirectX.DirectSound` 中的 `SecondaryBuffer` 缓冲区对象，但是用的过程中程序总是会卡住，后来更改了 `App.config`，将 `startup` 中的 `useLegacyV2RuntimeActivationPolicy` 设为“true”，然后程序到了定义设备对象的时候还是挂起，上网查一下之后，错误的原因是“检测到 LoaderLock Message Microsoft.DirectX.dll”正试图在 OS 加载程序锁内执行托管代码。”，把异常设置中 `Managed Debugging Assistant`s 去掉 `LoaderLock` 的引发复选框，然后就可以运行了。