

利用VB2005制作颜色渐变的进度条计算机二级考试 PDF转换  
可能丢失图片或格式，建议阅读原文

[https://www.100test.com/kao\\_ti2020/572/2021\\_2022\\_\\_E5\\_88\\_A9\\_E7\\_94\\_A8VB20\\_c97\\_572485.htm](https://www.100test.com/kao_ti2020/572/2021_2022__E5_88_A9_E7_94_A8VB20_c97_572485.htm) 现在Windows环境下的应用程序经常要用到进度条，用以指示出一些复杂进程的运行状态。目前我们在窗体设计时，使用的都是单色进度条。本文讲述在WindowsXP环境下，使用VB2005制作一个漂亮的颜色渐变的进度条组件。

一、技术细节 这种进度条的显示相当于使用许多颜色相近的“画笔”在窗体上绘制方框。这些“画笔”就相当于一个画笔数组。首先要建立颜色数组，初始的时候放入一些颜色值，然后根据平滑度的设置来重新设置颜色数组。依次取出颜色数组中两个相邻的颜色值，计算这两个颜色经过混合后的颜色，再放入到两个颜色的中间。例子中颜色数组中有8种颜色。平滑度的值从0开始循环，最高是7。每经过一次循环，颜色数组中就增加了一些过渡颜色。平滑度设置的越大，则颜色过渡的越柔和。产生的效果越漂亮。颜色数组生成之后，画笔数组就确定了。它的大小和颜色数组一致。有了画笔数组，下面的问题就是在窗体上进行绘图操作了。

二、实现原理 在绘图操作之前。所以要重载基类中的窗体的消息处理事件WndProc，窗体大小变化事件OnResize和窗体重绘事件OnPaint。加入以下属性设置：Minimum。表示进度条的最小值。默认是0。Maximum。表示进度条的最大值。默认是100。Value。表示了进度条的当前值。默认是0。Smoothness。表示了进度条颜色渐变的平滑度。默认是0。在初始化的时候，设置了以上四个属性后，相应的画笔数组产生了。当前值的每次改变都要触发OnPaint事件。触发事

件后，首先绘制进度条的外部边框，接着得到内部边框的总宽度和高度。根据当前值，最小值和最大值计算完成百分比，由完成百分比乘以总宽度得到当前值占用的宽度。总宽度除以画笔数组的个数得到每个画笔所占用的宽度。然后就是从初始宽度循环到当前值占用的宽度，从画笔数组中依次使用画笔按照每个画笔所占用的宽度和高度填充颜色。循环结束后，如果百分比为100%。则使用画笔数组中的最后一个画笔绘制颜色。进度条显示完毕。

### 三、实现步骤

打开VS2005，点击“文件”下的“新建项目”，在打开的窗体中，“项目类型”选择“Visual Basic”。“模板”选择“Windows 控件库”。在左下角，“名称”改为“GradientProcessBar”。按下“确定”按钮，新建工程。在“解决方案资源管理器”窗口中，将文件名UserControl1.vb 改为 GradientProcessBar.vb。将下面完整的代码加入到GradientProcessBar.vb文件中。

```
Option Strict On
Option Explicit On
Imports System.Drawing
Imports System.Windows.Forms

Public Class GradientProcessBar Inherits UserControl
    Public Const MinSmoothness As Integer = 0 ' 最小平滑度
    Public Const MaxSmoothness As Integer = 7 ' 最大平滑度
    Private Const BorderWidth As Integer = 2 ' 边界宽度
    Private lstDefault As List(Of Color) ' 颜色数组
    Private lstBrushes As List(Of SolidBrush) ' 画笔数组
    Private m_Value As Integer = 0 ' 当前进度值
    Private m_Minimum As Integer = 0 ' 最小进度值
    Private m_Maximum As Integer = 100 ' 最大进度值
    Private m_Smoothness As Integer = 0 ' 平滑度 ' 进度值设置
    Public Property Value() As Integer
        Get Return m_Value End Get
        Set(ByVal value As Integer) m_Value = value
        If m_Value gt.
```

```
m_Maximum Then m_Value = m_Maximum Me.Invalidate(False)
End Set End Property ' 最小值设置 Public Property Minimum()
As Integer Get Return m_Minimum End Get Set(ByVal value As
Integer) m_Minimum = value If m_Minimum lt. m_Minimum
Then m_Value = m_Minimum Me.Invalidate(False) End Set End
Property ' 最大值设置 Public Property Maximum() As Integer
Get Return m_Maximum End Get Set(ByVal value As Integer)
m_Maximum = value If m_Minimum gt. m_Maximum Then
m_Value = m_Maximum Me.Invalidate(False) End Set End
Property ' 平滑度设置 Public Property Smoothness() As Integer
Get Return m_Smoothness End Get Set(ByVal value As Integer) If
value gt. GradientProcessBar.MaxSmoothness Then value =
MaxSmoothness m_Smoothness = value BuildColorList(1stDefault)
Me.Invalidate(False) End Set End Property ' 交换值 Private Sub
Swap(ByRef val1 As Integer, ByRef val2 As Integer) Dim temp As
Integer temp = val1 val1 = val2 val2 = temp End Sub Public Sub
New() InitializeComponent() Me.BorderStyle =
Windows.Forms.BorderStyle.Fixed3D ' 设置窗体为凹陷的效果
' 最简单的实现双缓冲绘图 ' AllPaintingInWmPaint 忽略系统
消息,防止图像闪烁 ' DoubleBuffer 设置双缓冲,防止图像
抖动 ' UserPaint 自己为控件做所有的绘图操作 ' Opaque 使
窗体绘制时,不绘制背景 100Test 下载频道开通,各类考试题目
直接下载。详细请访问 www.100test.com
```