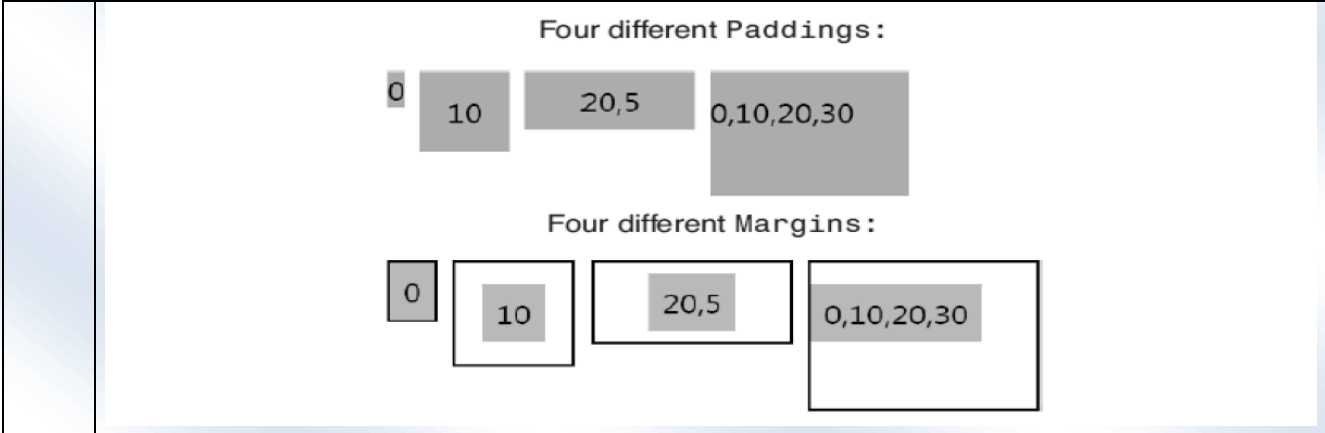


**CS411****Midterm upto 2015****MC130401237****Visual Programming****Subjective Solved****M. Kamran Khan**

No.	Question Statement
1	<p><b>Name the Built-in Transforms - Their Default Values?</b></p> <p><b>Name 5 built in 2D transform which are supported by "System.Windows.Media" namespace.</b></p>
Ans.	<p>1. Translate Transform</p> <p>2. Skew Transform - AngleX, AngleY, CenterX &amp; CenterY (default value = 0)</p> <p>3. Scale Transform</p> <p>4. Rotate Transform – Angle, Centrex, Centery = 0;</p> <p>5. Matrix Transform</p>
2	<p><b><u>Consider the following code snippet and write the output:</u></b></p> <pre> XmlDocument xmlDoc = new XmlDocument(); xmlDoc.LoadXml("&lt;uni name='Virtual'&gt;University Node&lt;/uni.&gt;"); console.WriteLine(xmlDoc.DocumentElement.Name); console.WriteLine(xmlDoc.DocumentElement.InnerText); console.ReadLine(); </pre>
Ans.	<p>Element Name: Virtual</p> <p>Inner Text: University Node</p>
3	<p><b>Which Option to be Used to Fill the Dock Panels?</b></p>
Ans.	<p>Dock panel allows easy docking of elements to an entire side.</p> <p>Dock attached property has left, right, top, bottom values.</p> <p>Last child fills space unless lastchildfill=false.<b>in</b></p> <p>We can also do this by setting Background = "Red"</p>
4	<p><b>Write Code to Delete Non-Empty Cell?</b></p>
Ans.	<pre> { cell.value = string.empty } </pre>
5	<p><b>Can BAML be Decompiled back into XAML? Explain with Valid Reason.</b></p>
Ans.	<p>Yes, BAML can be converted back to XAML.</p> <p>BAML is binary application markup language. It just a compressed representation of XAML. There is even a BAML reader available.</p>
6	<p><b><u>What are Default Values of These?</u></b></p> <ol style="list-style-type: none"> <li>1. Max Height</li> <li>2. Min Height</li> <li>3. Max Width</li> </ol>

	<b>4. Min Width</b>
<b>Ans.</b>	Min=0 and Max=Infinity by default. All framework elements have Height and Width properties of type double.
<b>7</b>	<b>How can we know which cells are directly affected by the resizing of "GridSplitters"?</b>
<b>Ans.</b>	It depends on Gridsplitter's alignment values. Default horizontal alignment=right and vertical alignment=stretch. When all proportionally sized, changes co-efficiently accordingly. When absolute sizing, changes only top or left of the cells. Remaining cells pushed down or right.
<b>8</b>	<b>Which provide multi-touch support, WPF3 or WPF4?</b>
<b>Ans.</b>	WPF 4.0 includes multi-touch support.
<b>9</b>	<b>Name of three type of row-definition and column-definition sizing that are supported by WPF "GRID" panel?</b>  <b>3 Ways of Rows and Columns Sizing?</b>
<b>Ans.</b>	1. Absolute sizing: device independent pixels means no grow shrink, 2. Auto-sizing: size to content. 3. Proportional or Star sizing: grows or shrinks.
<b>10</b>	<b>How to make xml document name? Write the syntax?</b>
<b>Ans.</b>	<code>Xmldocumentxmldoc = new xmldocument();</code>
<b>11</b>	<b>Types of Transform?</b>
<b>Ans.</b>	LayoutTransform RenderTransform. One applied before layout and one after.
<b>12</b>	<b><u>Convert Xml into C#?</u></b> Span id="IL_AD1" class="IL_AD">Button xmlns="http://schemas.microsoft.com/winfx/2006/xaml/presentation" Content="OK"/>
<b>Ans.</b>	1 System.Windows.Controls.Button b = new System.Windows.Controls.Button(); 2 b.Content = "OK";
<b>13</b>	<b>Write the C# code line to take any string as input from user and the write that string to a file named "Student.txt"?</b>
<b>Ans.</b>	using System; class Program { static void Main() { while (true) // Loop indefinitely

	<pre> { Console.WriteLine("Enter input:"); // Prompt string line = Console.ReadLine(); // Get string from user if (line == "exit") // Check string { break; } // Write the string to a file. { System.IO.StreamWriter file = new System.IO.StreamWriter("c:\\student.txt"); file.WriteLine(lines); } file.Close(); </pre>
<b>14</b>	<b>Which panel functionality is actually a superset of “stack panel” functionality?</b>
<b>Ans.</b>	Dock panel is a superset of stack panel, Dock attached property has left, right, top, bottom values.
<b>15</b>	<b>For what purpose XAML is used in WF and WCF?</b>
<b>Ans.</b>	<p>XAML is primarily used to describe interfaces in WPF and Silverlight.</p> <p>It is also used to express activities and configurations in workflow foundation (WF) and windows communication foundation (WCF).</p> <p>It's a common language for programmers and other experts e.g. UI design experts. Field specific development tools can be made. Field experts are graphic designers. They can use a design tool such as expression blend. Other than co-ordinating with designers, XAML is good for a concise way to represent UI or hierarchies of objects, encourages separation of front-end and back-end, tool support with copy and paste, used by all WPF tools.</p>
<b>16</b>	<b>WPF Panels are Derived from Which Class?</b>
<b>Ans.</b>	Panels are parents supporting layout of multiple children. They derive from the abstract System.Windows.Controls.Panel class.
<b>17</b>	<b>How matrix transform is work like translate transform and give example?</b>
<b>Ans.</b>	<p>Translate Transform has X — Amount to move horizontally (default value = 0), Y — Amount to move vertically (default value = 0). It has no effect as a layout transform.</p> <p>Matrix Transform has a single Matrix property (of type System.Windows.Media.Matrix) representing a 3x3 affine transformation matrix.</p>
<b>18</b>	<b>Padding and margin, illustrate 3 instances?</b>
<b>Ans.</b>	<p>Next is margin and padding. All framework elements have margin, controls (and Border) have padding. Margin is extra space “outside” and padding is “inside” edges of the element. They are of type Thickness that can represent 1, 2, or 4 values. Negative margin is possible. Label's default padding=5. Xaml specification is helped by type converter.</p>



**19** Write xml code of given C# code?  
 StakPanelstackpanel = new StakePanel  
 Button button = new system.window.control Button;  
 button.widh = 50;  
 button.Hight = 70  
 button.HOrizontelAlligenemtn = right;  
 stakepanel.child.add(button);

**Ans.** <?xml version="1.0" encoding="UTF-8" ?>  
 <root>  
 <0>  
 <StakPanelstackpanel = new StakePanel>  
 Button button = new system.window.control Button;  
 </StakPanelstackpanel = new StakePanel>  
 </0>  
 <1>  
 <StakPanelstackpanel = new StakePanel>  
 button.widh = 50;</StakPanelstackpanel = new StakePanel>  
 </1>  
 <2>  
 <StakPanelstackpanel = new StakePanel>  
 button.Hight = 70  
 </StakPanelstackpanel = new StakePanel>  
 </2>  
 <3>  
 <StakPanelstackpanel = new StakePanel>  
 button.HOrizontelAlligenemtn = right;  
 </StakPanelstackpanel = new StakePanel>

	<pre> &lt;/3&gt;  &lt;4&gt; &lt;StakPanelstackpanel = new StakePanel&gt; stakepanel.child.add(button); &lt;/StakPanelstackpanel = new StakePanel&gt; &lt;/4&gt; &lt;/root&gt; </pre>
<b>20</b>	<b>Name the Properties Supported by Scale Transform, write their default values?</b>
<b>Ans.</b>	<p>Scale Transform has  scalex, scaley = 1; by default.  centerx, centery = 0; by default.</p>
<b>21</b>	<b>Write names of 3 children of an object?</b>
<b>Ans.</b>	<p>An object element can have three types of children:</p> <ul style="list-style-type: none"> <li>- A value for a content property</li> <li>- Collection items</li> <li>- A value that can be type-converted to the object element</li> </ul>
<b>22</b>	<b>When Context Overflow in Built-in Panel Arise, How to Fix That?</b>
<b>Ans.</b>	<p>Content overflow is when parents and children can't agree on the use of available space. Content overflow can be dealt with Clipping, Scrolling, Scaling, Wrapping and Trimming.</p>
<b>23</b>	<b>Logical Tree of the Code?</b>
<b>Ans.</b>	<p>Logical tree exists even if there is no xaml. Properties, events, resources are tied to logical trees. Properties propagated down and events can be routed up or down the tree. Its a simplification of whats actually going on when rendered. Visual tree can be thought of as an extension of the logical tree though some things can be dropped as well. Visual tree exposes visual implementation details e.g. A listbox is a border, two scrollbars and more. Only things from visual or visual3d appear in a visual tree. Avoid depending on visual tree in your code.</p>
<b>24</b>	<b>Syntax of CDATA Section?</b>
<b>Ans.</b>	<pre> &lt;x:Code&gt; &lt;![CDATA[ Void button_Click(object sender, routedeventargs e) { This.Close(); } ]]&gt; &lt;/x:Code&gt; </pre>

<b>25</b>	<b>Explain Routed Events?</b>
<b>Ans.</b>	<p>Routed events are like dependency properties on top of .net properties. Similarly routed events are a layer on top of .net events. They can travel up or down a visual or logical tree. Helps apps remain oblivious to the visual tree. E.g. Button exposes Click based on mouseleftbuttondown and keydown but actually buttonchrome or textblock visual child fires it. Event travels up the tree and Button eventually sees it. Or like in a previous example rectangle on top of button. So arbitrary complex content but Click still raised. Otherwise would require custom code on inner content or consumers. Very similar to dependency property concept.</p>
<b>26</b>	<b>WPF Read-Only?</b>
<b>Ans.</b>	<p>Read only properties:</p> <ol style="list-style-type: none"> <li>1. Desiredsize</li> <li>2. Rendersize</li> <li>3. Actualheight</li> <li>4. Actualwidth.</li> </ol> <p>They are useful to act programmatically on final size.</p>
<b>27</b>	<b>What We Achieve from ZIndex Codes?</b>
<b>Ans.</b>	<p>ZIndex is an integer with a default value of 0 that you can set to any number (positive or negative). Elements with larger ZIndex values are rendered on top of elements with smaller ZIndex values, so the element with the smallest value is in the back, and the element with the largest value is in the front.</p>
<b>28</b>	<b>Why Most WPF Classes are Inherently Thread-Unsafe?</b>
<b>Ans.</b>	<p>Since WPF classes derive from dispatcherobject therefore they are inherently thread-unsafe. Reason: Dispatcherobject is the base class meant for any object that wishes to be accessed only on the thread that created it.</p>
<b>29</b>	<b>Which transforms all framework elements have?</b>
<b>Ans.</b>	<ol style="list-style-type: none"> <li>1. Layout Transform</li> <li>2. Render Transform</li> </ol>
<b>30</b>	<b>What will be the easiest way if someone wants to know whether a certain string is a number or not? Elaborate your answer with an example?</b>
<b>Ans.</b>	<p>IsNumeric(String s) This method is used to find if the string is numeric or not.</p>
<b>31</b>	<b>Write the XAML code to create a “DockPanel” having four buttons with dock settings i.e. Top, Bottom, Right and Left.</b>
<b>Ans.</b>	<pre>&lt;DockPanel&gt; &lt;Button DockPanel.Dock="Top" Background="Red"&gt;1 (Top)&lt;/Button&gt; &lt;Button DockPanel.Dock="Left" Background="Orange"&gt;2 (Left)&lt;/Button&gt; &lt;Button DockPanel.Dock="Right" Background="Yellow"&gt;3 (Right)&lt;/Button&gt; &lt;Button DockPanel.Dock="Bottom" Background="Lime"&gt;4 (Bottom)&lt;/Button&gt; &lt;Button Background="Aqua"&gt;5&lt;/Button&gt; &lt;/DockPanel&gt;</pre>

<b>32</b>	<b>Write two differences between WPF and DirectX?</b>
<b>Ans.</b>	<p>WPF is made for designers to create vector-based applications using XML and managed code. It uses DirectX as the rendering engine under the covers.</p> <p>DirectX is a set of APIs that abstracts access to modern graphics rendering hardware, as well as other technologies such as sound or input devices.</p> <p>WPF is not designed for any significant game programming, other than, say, a simple card game.</p> <p>DirectX is designed with the intention of allowing you to create a 3D user interface into your game.</p>
<b>33</b>	<b>When two or more independent grids need to be synchronized, then how can we synchronize them?</b>
<b>Ans.</b>	Sharedsizegroup is used to synchronized when they are docked.
<b>34</b>	<b>What is the biggest feature of dependency property?</b>
<b>Ans.</b>	Its built-in ability to provide change notification.
<b>35</b>	<b>What is the purpose of using xaml extensions?</b>
<b>Ans.</b>	To extend the expressiveness of XAML.
<b>36</b>	<b>WPF combines the best attribute of some systems. write the names of three?</b>
<b>Ans.</b>	<p>WPF aims to combine the best attributes of systems such as:</p> <ul style="list-style-type: none"> <li>- DirectX (3D and hardware acceleration),</li> <li>- Windows Forms (developer productivity),</li> <li>- Adobe Flash (powerful animation support)</li> <li>- HTML (declarative markup)</li> </ul>
<b>37</b>	<b>What is the purpose of WPF 2D transform classes?</b>
<b>Ans.</b>	To rotate, scale, move (translate), and skew FrameworkElement objects.
<b>38</b>	<b>For which class WPF are derived?</b>
<b>Ans.</b>	using System.Windows;
<b>39</b>	<b>Why C# let us Develop devices resulation independent graphics?</b>
<b>Ans.</b>	C# enable us to develop device independent graphics so that same application/ GUI have same aspect ratio and appearance on all kind of displays. e.g. Windows phone, Windows PC etc.
<b>40</b>	<b>Which term is used to refer to an object that derives from "UIElement" or "FrameworkElement"?</b>
<b>Ans.</b>	<p>Uielement is the base class for all 2D visual objects. System.Windows.UIElement</p> <p>Frameworkelement is the base class that adds support for styles, data binding, resources etc. System.Windows.FrameworkElement</p>
<b>41</b>	<b>Name the three routed strategies?</b>
<b>Ans.</b>	<ul style="list-style-type: none"> <li>• <b>Tunneling</b>—The event is first raised on the root, then on each element down the tree until the source element is reached (or until event is marked as handled).</li> </ul>

	<ul style="list-style-type: none"> <li>• <b>Bubbling</b>—The event is first raised on the source element, then on each element up the tree until the root is reached (or until the event is marked as handled).</li> <li>• <b>Direct</b>—The event is only raised on the source element. Only source element is given the chance to invoke handlers in response.</li> </ul>
<b>42</b>	<b>Name the three parameters of Handlers for routed events?</b>
<b>Ans.</b>	Two are known - third is unknown. <b>1. Sender</b> Type: System.Object - The source of the event. <b>2. E</b> Type: System.EventArgs - An object that contains no event data.
<b>43</b>	<b>Why "Grid" panel is the best choice for most complex layout scenarios?</b>
<b>Ans.</b>	The grid is a layout panel that arranges its child controls in a tabular structure of rows and columns. Its functionality is similar to the HTML table but more flexible. A cell can contain multiple controls, they can span over multiple cells and even overlap themselves.
<b>44</b>	<b>Why C# let us Develop devices resolution independent graphics?</b>
<b>Ans.</b>	Fortunately for us, C# makes it very easy. The File class, from the System.IO namespace comes with pretty much everything we could possibly want, making it very easy to do simple reading and writing of a file. File.ReadAllText File.WriteAllText      Also      AppendAllText
<b>45</b>	<b>What happens when an event is moved up?</b>
<b>Ans.</b>	It's actually called bubbling in routed events. <b>Bubbling</b> —The event is first raised on the source element, then on each element up the tree until the root is reached (or until the event is marked as handled).
<b>46</b>	<b>Why events are used in pairs? explain with example?</b>
<b>Ans.</b>	In VPL, a program is created by assembling pairs of one event and one or more action blocks. For example, a pair composed of a <i>button</i> event and a <i>motors</i> action tells the robot to move when a button is touched on the robot.

**Little Research is Required at Your End as Some of the Answers are Not Absolutely Perfect**