```csharp
using System;
using System.Drawing;
using MonoTouch.Foundation;
using MonoTouch.UIKit;
using System.Threading;

namespace ControlDemo_LMT
{
    public partial class ProgressView : UIViewController
    {
        UIProgressView _progView;
        PDFCodeView _pdfView;
        string _pdfFile = "ProgressView.pdf";

        public ProgressView () : base ("ProgressView", null)
        {
        }

        public override void DidReceiveMemoryWarning ()
        {
            // Releases the view if it doesn't have a superview.
            base.DidReceiveMemoryWarning ();

            // Release any cached data, images, etc that aren't in use.
        }

        public override void ViewDidLoad ()
        {
            base.ViewDidLoad ();
            showProgressBtn.TouchUpInside += HandleShowProgressTouchUpInside;
        }

        void HandleShowProgressTouchUpInside (object sender, EventArgs e)
        {
            _progView = new UIProgressView ();
            _progView.Frame = new RectangleF (0, 0, View.Frame.Width - 20, 100);
            _progView.Center = View.Center;
            _progView.Style = UIProgressViewStyle.Default;

            View.AddSubview (_progView);

            Thread t = new Thread (DoSomethingElse);
            t.Start ();

            pdfbtn.TouchUpInside += delegate {
                _pdfView = new PDFCodeView(_pdfFile);
```

```
                this.PresentViewController(_pdfView, true, null);
            };
        }

        void DoSomethingElse ()
        {
            int n = 5;

            for (int i = 0; i < n; i++) {
                Thread.Sleep (1000);

                using (var pool = new NSAutoreleasePool ()){
                    this.InvokeOnMainThread (delegate {
                        _progView.Progress = (float)(i + 1) / n;
                    });
                }
            }
        }
    }
}
```