```objc
@implementation QuartzPDFView

-(id)initWithFrame:(CGRect)frame
{
    self = [super initWithFrame:frame];
    if(self != nil)
    {
        CFURLRef pdfURL = CFBundleCopyResourceURL(CFBundleGetMainBundle(),
CFSTR("Quartz.pdf"), NULL, NULL);
        pdf = CGPDFDocumentCreateWithURL((CFURLRef)pdfURL);
        CFRelease(pdfURL);
    }
    return self;
}

-(void)dealloc
{
    CGPDFDocumentRelease(pdf);
    [super dealloc];
}

-(void)drawInContext:(CGContextRef)context
{
    // PDF page drawing expects a Lower-Left coordinate system, so we flip the coordinate system
    // before we start drawing.
    CGContextTranslateCTM(context, 0.0, self.bounds.size.height);
    CGContextScaleCTM(context, 1.0, -1.0);

    // Grab the first PDF page
    CGPDFPageRef page = CGPDFDocumentGetPage(pdf, 1);
    // We're about to modify the context CTM to draw the PDF page where we want it, so save the
graphics state in case we want to do more drawing
    CGContextSaveGState(context);
    // CGPDFPageGetDrawingTransform provides an easy way to get the transform for a PDF page. It
will scale down to fit, including any
    // base rotations necessary to display the PDF page correctly.
    CGAffineTransform pdfTransform = CGPDFPageGetDrawingTransform(page, kCGPDFCropBox,
self.bounds, 0, true);
    // And apply the transform.
    CGContextConcatCTM(context, pdfTransform);
    // Finally, we draw the page and restore the graphics state for further manipulations!
    CGContextDrawPDFPage(context, page);
    CGContextRestoreGState(context);
}

@end
```