

خروجی PDF زیر را در نظر بگیرید:

تاریخ: 18/11/1390
شماره پروژه: 56/4/3/2/1
اسلش: A/13/12
بك اسلش: 14\13\12
مساوي و جمع: 5=3+2
سمي گولون: ;1+1=2
دلار: \$12
كاما: 12,34,67
نقطه: 12.34
پرانتز: متن (ساده)

مشکلی را در آن مشاهده می‌کنید؟ اصل آن یا صحیح آن باید به شکل زیر باشد:

تاریخ: 1390/11/18
شماره پروژه: 1/2/3/4/56
اسلش: 12/A/13
بك اسلش: 12\13\14
مساوي و جمع: 2+3=5
سمي گولون: 2=1+1;
دلار: 12\$
كاما: 12,34,67
نقطه: 12.34
پرانتز: متن (ساده)

و این وارونه نمایش دادن‌ها، دقیقا مشکلی است که حین کار با iTextSharp برای نمایش متنی مثلا به همراه یک تاریخ شمسی وجود دارد. البته این مشکل هم اساسا به خود استاندارد یونیکد برمی‌گردد که یک سری کاراکتر را «[کاراکتر ضعیف](#)» معرفی کرده؛ برای مثال کاراکتر اسلش بکار رفته در یک تاریخ هم از این دست است. بنابراین PDF تولیدی توسط iTextSharp از دید استاندارد

یونیکد مشکلی ندارد، زیرا یک «نویسه ضعیف» مثل اسلش نمی‌تواند جهت را تغییر دهد؛ مگر اینکه از یک «[نویسه قوی](#)» برای دستکاری آن استفاده شود. برای مثال این نویسه‌ها قوی هستند:

```
U+202A: LEFT-TO-RIGHT EMBEDDING (LRE)
U+202B: RIGHT-TO-LEFT EMBEDDING (RLE)
U+202D: LEFT-TO-RIGHT OVERRIDE (LRO)
U+202E: RIGHT-TO-LEFT OVERRIDE (RLO)
U+202C: POP DIRECTIONAL FORMATTING (PDF)
```

برای رسیدن به تصویر صحیح نمایش داده شده در بالا، متد `FixWeakCharacters` زیر را تهیه کرده‌ام که حداقل با `iTextSharp` جواب می‌دهد:

```
using System;
using System.Diagnostics;
using System.IO;
using iTextSharp.text;
using iTextSharp.text.pdf;

namespace RleTests
{
    class Program
    {
        const char RightToLeftEmbedding = (char)0x202B;
        const char PopDirectionalFormatting = (char)0x202C;

        static string FixWeakCharacters(string data)
        {
            if (string.IsNullOrEmpty(data)) return string.Empty;
            var weakCharacters = new[] { @"\"", "/", "+", "-", "=", ";", "$" };
            foreach (var weakCharacter in weakCharacters)
            {
                data = data.Replace(weakCharacter, RightToLeftEmbedding + weakCharacter +
                PopDirectionalFormatting);
            }
            return data;
        }

        static void Main(string[] args)
        {
            using (var pdfDoc = new Document(PageSize.A4))
            {
                PdfWriter.GetInstance(pdfDoc, new FileStream("Test.pdf", FileMode.Create));
                pdfDoc.Open();

                FontFactory.Register("c:\\windows\\fonts\\Arial.ttf");
                Font tahoma = FontFactory.GetFont("Arial", BaseFont.IDENTITY_H);

                var table1 = new PdfPTable(1);
                table1.WidthPercentage = 100;

                var pdfCell = new PdfPCell
                {
                    RunDirection = PdfWriter.RUN_DIRECTION_RTL,
                    Border = 0,
                    Phrase = new Phrase(FixWeakCharacters(
                        "18/11/1390" + " " + "تاریخ: " + Environment.NewLine +
                        "56/4/3/2/1" + " " + "شماره پروژه: " + Environment.NewLine +
                        "12" + " " + "اسلش: A/13" + Environment.NewLine +
                        "14\\13\\12" + " " + "یک اسلش: " + Environment.NewLine +
                        "5=3+2" + " " + "مساوی و جمع: " + Environment.NewLine +
                        "1+1=2" + " " + "سمی گولون: " + Environment.NewLine +
                        "12" + " " + "$دلار: " + Environment.NewLine +
                        "12,34,67" + " " + "کاما: " + Environment.NewLine +
                        "12.34" + " " + "نقطه: " + Environment.NewLine +
                        "متن (ساده)" + " " + "پرانتز: " + Environment.NewLine +
                        ")",
                    tahoma)
                };

                table1.AddCell(pdfCell);
                pdfDoc.Add(table1);
            }
        }
    }
}
```

```
        }  
        Process.Start("Test.pdf");  
    }  
}
```

از این نوع مشکلات حین کار با HTML هم هست؛ وارونه نمایش داده شدن تاریخ فارسی در بین یک متن راست به چپ. البته در آنجا راه حل زیر هم توصیه شده (بدون نیاز به دستکاری نویسه‌ها):

```
<span dir="ltr" style="display:inline">1390/11/19</span>
```

نظرات خوانندگان

نویسنده: فرهاد یزدان پناه
تاریخ: ۱۳۹۰/۱۱/۱۹ ۲۱:۱۶:۴۲

ممنون. جناب نصیری.
آقای حاجلو هم مطلبه بسیار مفیدی در همین موضوع دارند.
[/http://hajloo.wordpress.com/2009/03/02/persian-text-problem-in-ltr-forms](http://hajloo.wordpress.com/2009/03/02/persian-text-problem-in-ltr-forms)

نویسنده: linux
تاریخ: ۱۳۹۰/۱۱/۲۲ ۲۳:۱۹:۴۵

برای جدا سازی اجزای تاریخ شمسی، ماه، روز و سال نباید از / استفاده کرد در unicode برای این کار از Unicode Character (U+0600 'ARABIC DATE SEPARATOR') استفاده کنید برای دیدن جزئیات بیشتر
<http://www.fileformat.info/info/unicode/char/60d/index.htm>

نویسنده: وحید نصیری
تاریخ: ۱۳۹۰/۱۱/۲۳ ۰۰:۰۰:۴۲

شاید در زبان عربی اینطور باشه. حتما می‌دونید که نحوه نمایش و نویسه‌های اعداد 4 و 6 عربی و فارسی یکی نیست. ک و ی عربی و فارسی هم یکی نیست. حتی ممیز فارسی هم شیوه خاص خودش را دارد و کلا بحث من اینجا در مورد نحوه متداول ورود اطلاعات در زبان فارسی است؛ در مورد هزاران هزار سطر موجود. ضمن اینکه اگر به مثال دقت کرده باشید یک شماره پروژه‌ای هم این وسط هست که الگویی شبیه به تاریخ ندارد؛ به علاوه یک سری نویسه ضعیف دیگر مثل مساوی و جمع و منها و غیره. به علاوه بحث من در مورد کتابخانه تولید PDF ذکر شده است و راه حلی که با آن جواب بدهد. راه حل بالایی که من مطرح کردم در نمایش هیچ تغییری ایجاد نمی‌کنه. این حرف بکارگرفته شده، نامرئی هستند. PDF هم یک لایه Presentation است. بنابراین زمانیکه اطلاعاتی را درست نمایش می‌دهد، یعنی هدف اصلی خودش را برآورده کرده.

نویسنده: Nasser Mansouri
تاریخ: ۱۳۹۰/۱۲/۰۸ ۱۵:۲۹:۴۷

سلام، می‌خواستم اگر ممکن باشه من رو راهنمایی کنید، من می‌خوام با استفاده از itextsharp محتوای یک فایل پی دی اف رو به صورت txt ذخیره کنم، با زبانهای چپ به راست خیلی آسون هست ولی در زبان راست به چپ، کلمات از آخر به اول نوشته می‌شن مثلاً کلمه ارزیابی به صورت "یبایزرا" خوانده می‌شه، ممنون می‌شم من رو راهنمایی بکنید.

```
(private static string ParsPDFToString
}
;("PdfReader reader = new PdfReader("c:/2v.pdf
;(PdfReaderContentParser parser = new PdfReaderContentParser(reader
;())StringBuilder sb = new StringBuilder
;())Console.WriteLine("reader.NumberOfPages : " + reader.NumberOfPages.ToString
;())Console.ReadKey
(++for (int i = 1; i <= reader.NumberOfPages; i
}
)ITextExtractionStrategy strategy = parser.ProcessContent
(i , new SimpleTextExtractionStrategy
;
;())sb.Append(strategy.GetResultantText
{
;())return sb.ToString
```

{

نویسنده: وحید نصیری
تاریخ: ۱۳۹۰/۱۲/۰۸ ۲۰:۰۶:۵۱

بستگی داره نرم افزار تولید PDF از چه روشی استفاده کرده باشه. بعضی‌ها از چرخاندن حروف استفاده می‌کنند و این روش بسیار متداولی هست. یعنی مشکل از iTextSharp نیست. در اصل به همین ترتیب حروف ذخیره شدن. الگوریتم اولیه به همین صورت بوده.

نویسنده: وحید نصیری
تاریخ: ۱۳۹۰/۱۲/۰۹ ۱۲:۲۹:۱۸

دو مورد تکمیلی:
- کار این چرخاندن‌ها توسط دو کلاس ArabicLigaturizer و BidilLine در iTextSharp انجام می‌شود. سورس کتابخانه را دریافت و این دو کلاس را مطالعه کنید (ضمن اینکه PDF های فارسی هم وجود دارند که اصلا با این الگوریتم‌ها تهیه نشده‌اند و خلاصه راه سختی را پیش رو دارید). iTextSharp انجمنی نداره ولی یک mailing list فعال داره:
<https://lists.sourceforge.net/lists/listinfo/itext-questions>