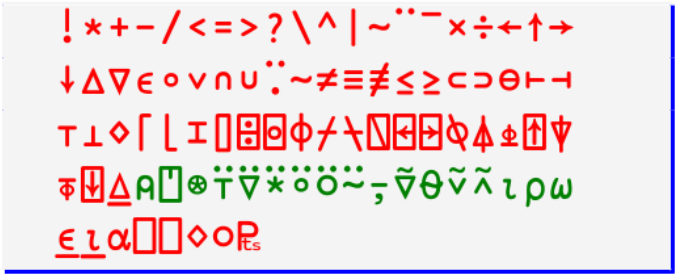

UTF-8 APL characters within a LaTeX `lstlisting` environment. Click for `*.tex` source code.

Typesetting UTF-8 APL code with the LaTeX lstlisting package

Posted: 16 Aug 2011 02:27:57

Typesetting APL source code has always been a pain in the ass! In the dark ages, (the 1970's), you had to fiddle with APL type-balls and live without luxuries like *lower case letters*. With the advent of general outline fonts it became *technically possible* to render APL glyphs on standard display devices provided you:

1. Designed your own APL font.
2. Mapped the *atomic vector* of your APL to whatever encoding your font demanded.
3. Wrote WSFULL's of junk transliteration functions to dump your APL objects as font encoded text.

It's a testament to either the talent, or pig headedness of APL programmers, that many actually did this. We all hated it! We still hate it! But, like an abused spouse, we kept going back for more. *It's our fault; if we loved APL more it would stop hitting us!*

When Unicode appeared APL'ers cheered — our long ASCII nightmare was ending. The more politically astute worked to include the APL characters in the Unicode standard. Hey if Klingon is there why not APL? Everyone thought it was just a matter of time until APL vendors abandoned their nonstandard atomic vectors and fully embraced Unicode. With a few notable exceptions we are still waiting. *While we wait the problem of typesetting APL source code festers.*

My preferred source code listing tool is the LaTeX lstlisting package. lstlisting works well for standard ANSI source code. I use it for J, C#, SQL, C, XML, Ocaml, Mathematica, F#, shell scripts and LaTeX source code, i.e. everything except APL!

lstlisting is an eight bit package; it will not handle arbitrary Unicode out of the box. I didn't know how to get around this so I handled APL by enclosing UTF-8 APL text in plain `\begin{verbatim} … \end{verbatim}` environments. This works for XeLaTeX and LuaLaTeX but you lose all the lstlisting goodies. Then I saw an interesting tex.stackexchange.com posting about The 'listings' package and UTF-8. One solution to the post's "French ligature problem" showed how to force Unicode down lstlisting's throat. I wondered if the same method would work for APL. It turns out that it does!

If you insert the following snippet of TeX code in your document preamble LuaLaTeX and XeLaTeX will properly process UTF-8 APL text in lstlisting environments. You will need to download and install the APL385 Unicode font if it's not on your system. **A test LaTeX document illustrating this hack is available here. The compiled PDF is available here.**

```
% set lstlisting to accept UTF-8 APL text
\makeatletter
\lst@InputCatcodes
\def\lst@DefEC{%
 \lst@CCECUse \lst@ProcessLetter
  ^^80^^81^^82^^83^^84^^85^^86^^87^^88^^89^^8a^^8b^^8c^^8d^^8e^^8f%
  ^^90^^91^^92^^93^^94^^95^^96^^97^^98^^99^^9a^^9b^^9c^^9d^^9e^^9f%
  ^^a0^^a1^^a2^^a3^^a4^^a5^^a6^^a7^^a8^^a9^^aa^^ab^^ac^^ad^^ae^^af%
  ^^b0^^b1^^b2^^b3^^b4^^b5^^b6^^b7^^b8^^b9^^ba^^bb^^bc^^bd^^be^^bf%
  ^^c0^^c1^^c2^^c3^^c4^^c5^^c6^^c7^^c8^^c9^^ca^^cb^^cc^^cd^^ce^^cf%
  ^^d0^^d1^^d2^^d3^^d4^^d5^^d6^^d7^^d8^^d9^^da^^db^^dc^^dd^^de^^df%
  ^^e0^^e1^^e2^^e3^^e4^^e5^^e6^^e7^^e8^^e9^^ea^^eb^^ec^^ed^^ee^^ef%
  ^^f0^^f1^^f2^^f3^^f4^^f5^^f6^^f7^^f8^^f9^^fa^^fb^^fc^^fd^^fe^^ff%
  ^^^^20ac^^^^0153^^^^0152%
  ^^^^20a7^^^^2190^^^^2191^^^^2192^^^^2193^^^^2206^^^^2207^^^^220a%
  ^^^^2218^^^^2228^^^^2229^^^^222a^^^^2235^^^^223c^^^^2260^^^^2261%
  ^^^^2262^^^^2264^^^^2265^^^^2282^^^^2283^^^^2296^^^^22a2^^^^22a3%
  ^^^^22a4^^^^22a5^^^^22c4^^^^2308^^^^230a^^^^2336^^^^2337^^^^2339%
  ^^^^233b^^^^233d^^^^233f^^^^2340^^^^2342^^^^2347^^^^2348^^^^2349%
  ^^^^234b^^^^234e^^^^2350^^^^2352^^^^2355^^^^2357^^^^2359^^^^235d%
  ^^^^235e^^^^235f^^^^2361^^^^2362^^^^2363^^^^2364^^^^2365^^^^2368%
  ^^^^236a^^^^236b^^^^236c^^^^2371^^^^2372^^^^2373^^^^2374^^^^2375%
  ^^^^2377^^^^2378^^^^237a^^^^2395^^^^25af^^^^25ca^^^^25cb%
  ^^00}
\lst@RestoreCatcodes
\makeatother
```

From the blog: Analyze the Data not the Drivel
John D. Baker — revised: September 30, 2020