The inputnormalization package*

Marcel Krüger
tex@2krueger.de

July 5, 2021

Add support for normalising input files for LuaTeX and provide a common interface for LuaTEX and XeTEX.

1 Motivation

Modern TEX engines like XeTEX or LuaTEX natively accept Unicode input. Unicode is a rather special encoding since many characters can be encoded in different ways which are officially considered equivalent. This can sometimes lead to surprising behavior since many parts of TEX are not aware of this equivalences and therefore treats different encodings as different strings. This can show itself during rendering when some the same text might appear in different ways depending on the input, but it might also show itself in macro or option names: When e.g. non-English macro names are in use, the different encodings of the same name can name different macros, leading to hard to understand and solve errors.

Unicode defines a mechanism to solve such issues: It defines the normalization forms NFC and NFD. When text is normalized to one of these forms, then two equivalent strings are always encoded in the same way, leading to unique names and consistent rendering.

This package provides a uniform way to enable input normalization to either of these forms in both XeTEX and LuaTEX.

2 Usage

Just loading the package is enough to enable NFC normalization. This is the right option for almost all users:

```
\documentclass{article}
\usepackage{inputnormalization}
\begin{document}
Everything here gets normalized before it's processed by \TeX.
\end{document}
```

*This document corresponds to inputnormalization v0.2, dated 2021/07/05.

If you are a plain LuaTeX/XeTeX user, you can use

```
\input inputnormalization
Everything here gets normalized before it's processed by \TeX.
\bye
```

instead.

3 Advanced usage

In addition to enabling NFC normalization by default, the package makes `\Uinputnormalization` available as a cross engine version of `\XeTeXinputnormalization` to make the normalization controllable. See the XeTeX documentation for detailed usage. E.g. you could write

```
\documentclass{article}
\usepackage{inputnormalization}
\begin{document}
Everything here gets normalized to NFC before it's processed by \TeX.

\Uinputnormalization=0
Now normalization is disabled.

\Uinputnormalization=2
Here we normalize to NFD instead.
\end{document}
```

Warning: It is almost never a good idea to use different kinds of normalization in the same document, therefore you should set one kind of normalization directly after loading the package and not modify it afterwards.

Additionally NFC works much better in a TeX context than NFD, so you should not set this at all unless you know exactly what you are doing.

4 The implementation

```
⟨*package⟩
\NeedsTeXFormat{LaTeX2e}
\ProvidesPackage
  {inputnormalization}
  [2021/07/05 v0.2 Unicode input normalization]
⟨/package⟩
```

Only LuaTeX and XeTeX are supported. For other engines we show an error.

```
\ifx\directlua\undefined
```

```
\ifx\XeTeXinputnormalization\undefined
⟨*tex-package⟩
  \begingroup
    \ifx\PackageError\undefined
      \def\PackageError#1#2#3{\errhelp{#3}\errmessage{#1: #2}}
    \fi
⟨/tex-package⟩
  \PackageError{inputnormalization}{LuaTeX or XeTeX required}%
     {inputnormalization requires LuaTeX or XeTeX.
      Maybe you forgot to switch the engine in your editor?}
⟨*tex-package⟩
  \endgroup
⟨/tex-package⟩
  \else
```

First deal with XeTeX: Define `\Uinputnormalization` as an alias for `\XeTeXinputnormalization`. Make sure that `ltluatex` is loaded.

```
    \let\Uinputnormalization\XeTeXinputnormalization
  \fi
  \else
```

In LuaTeX we emulate `\Uinputnormalization` using a `process_input_buffer` callback. First ensure that `ltluatex` is loaded to have proper callback handling:

```
    \ifx\newluafunction\@undefined
      \input ltluatex
    \fi
```

We need a integer register to control the normalization and then the actual implementation of the callback. Nothing particularly interesting is happening here, the actual normalization is handled by `lua-uni-algos`.

```
    \newcount\Uinputnormalization
    \directlua{
      local getcount = tex.getcount
      local function ident(buf) return buf end
      local uni_normalize = require'lua-uni-normalize'
      local normalize = {[0] = ident, uni_normalize.NFC, uni_normalize.NFD}
      luatexbase.add_to_callback('process_input_buffer', function(buf)
        return normalize[getcount(\the\allocationnumber)](buf)
      end, 'inputnormalization')
    }
  \fi
```

Finally we enable NFC normalization as a reasonable default:

```
  \Uinputnormalization=1
  \endinput
```

Change History