The l3sort package
Sorting lists*

The LaTeX3 Project†

Released 2017/01/28

1 l3sort documentation

LaTeX3 comes with a facility to sort list variables (sequences, token lists, or comma-lists) according to some user-defined comparison. For instance,

```
\clist_set:Nn \l_foo_clist { 3 , 01 , -2 , 5 , +1 }
\clist_sort:Nn \l_foo_clist
  {
    \int_compare:nNnTF { #1 } > { #2 }
      { \sort_reversed: }
      { \sort_ordered: }
  }
```

will result in `\l_foo_clist` holding the values `{ -2 , 01 , +1 , 3 , 5 }` sorted in non-decreasing order.

The code defining the comparison should perform `\sort_reversed:` if the two items given as `#1` and `#2` are not in the correct order, and otherwise it should call `\sort_ordered:` to indicate that the order of this pair of items should not be changed.

For instance, a ⟨*comparison code*⟩ consisting only of `\sort_ordered:` with no test will yield a trivial sort: the final order is identical to the original order. Conversely, using a ⟨*comparison code*⟩ consisting only of `\sort_reversed:` will reverse the list (in a fairly inefficient way).

TEXhackers note: Internally, the code from l3sort stores items in `\toks` registers allocated locally. Thus, the ⟨*comparison code*⟩ should not call `\newtoks` or other commands that allocate new `\toks` registers. On the other hand, altering the value of a previously allocated `\toks` register is not a problem.

`\seq_sort:Nn`
`\seq_gsort:Nn`

`\seq_sort:Nn` ⟨*sequence*⟩ {⟨*comparison code*⟩}

Sorts the items in the ⟨*sequence*⟩ according to the ⟨*comparison code*⟩, and assigns the result to ⟨*sequence*⟩.

*This file describes v6829, last revised 2017/01/28.

†E-mail: latex-team@latex-project.org

\tl_sort:Nn
\tl_gsort:Nn

`\tl_sort:Nn` ⟨*tl var*⟩ {⟨*comparison code*⟩}

Sorts the items in the ⟨*tl var*⟩ according to the ⟨*comparison code*⟩, and assigns the result to ⟨*tl var*⟩.

\clist_sort:Nn
\clist_gsort:Nn

`\clist_sort:Nn` ⟨*clist var*⟩ {⟨*comparison code*⟩}

Sorts the items in the ⟨*clist var*⟩ according to the ⟨*comparison code*⟩, and assigns the result to ⟨*clist var*⟩.

\tl_sort:nN ⋆

`\tl_sort:nN` {⟨*token list*⟩} ⟨*conditional*⟩

Sorts the items in the ⟨*token list*⟩, using the ⟨*conditional*⟩ to compare items, and leaves the result in the input stream. The ⟨*conditional*⟩ should have signature `:nnTF`, and return `true` if the two items being compared should be left in the same order, and `false` if the items should be swapped.

TEXhackers note: The result is returned within `\exp_not:n`, which means that the token list will not expand further when appearing in an `x`-type argument expansion.

Index

The italic numbers denote the pages where the corresponding entry is described, numbers underlined point to the definition, all others indicate the places where it is used.

⋆ indicates fully expandable functions, which can be used within an `x`-type argument (in plain TEX terms, inside an `\edef`), as well as within an `f`-type argument.