The argproc package
Handle user input into integer, dimen, skip, and muskip expressions

Oliver Beery

Version 0.0.0 5 February 2025

1 About

The argproc package provides expl3 programming and macro support for creating user interfaces that take user input into integer, dimen, skip, and muskip expressions. The corresponding ε-TEX primitives—`\numexpr`, `\dimexpr`, `\glueexpr`, and `\muexpr`—expand and evaluate the following tokens and leave the remaining tokens, if any, in the input stream. This can be problematic if the user provides additional tokens that are not needed in the expression and can lead to unexpected results and low-level errors. This package provides commands that issue an error if the user inputs any extra trailing tokens that were not processed by the expression. This package also allows the programmer to append a default unit of measurement in dimen and skip expressions for user convenience.

2 Loading the package

Requirements:

- LaTeX 2_ε version 2023-11-01 or newer
- l3kernel version 2023-10-10 or newer

You may need to ensure that your LaTeX installation is up-to-date before using this package.

3 Argument processors

This section documents the argument processors provided by the argproc package. For more information on argument processors, see usrguide.[1]

`\argproc_int:n` `\argproc_int:n {⟨integer expression⟩}`

Argument processor for use in `\NewDocumentCommand` and friends. Evaluates ⟨*integer expression*⟩ and stores the result in `\ProcessedArgument`. Issues an error if ⟨*integer expression*⟩ contains any extra trailing tokens that were not processed by ⟨*integer expression*⟩.

[1] https://ctan.org/pkg/usrguide

\argproc_dim:nn `\argproc_dim:nn {⟨unit⟩} {⟨dimen expression⟩}`
\argproc_dim:n `\argproc_dim:n {⟨dimen expression⟩}`

Argument processor for use in `\NewDocumentCommand` and friends. Evaluates ⟨*dimen expression*⟩, appending a default unit of ⟨*unit*⟩, and stores the result in `\ProcessedArgument`. When fully expanded, ⟨*unit*⟩ must either match a valid unit of measurement or expand to nothing. Issues an error if ⟨*dimen expression*⟩ contains any extra trailing tokens that were not processed by ⟨*dimen expression*⟩, appending a default unit of ⟨*unit*⟩. `\argproc_dim:n` does not append a default unit.

\argproc_skip:nn `\argproc_skip:nn {⟨unit⟩} {⟨skip expression⟩}`
\argproc_skip:n `\argproc_skip:n {⟨skip expression⟩}`

Argument processor for use in `\NewDocumentCommand` and friends. Evaluates ⟨*skip expression*⟩, appending a default unit of ⟨*unit*⟩, and stores the result in `\ProcessedArgument`. When fully expanded, ⟨*unit*⟩ must either match a valid unit of measurement or expand to nothing. Issues an error if ⟨*skip expression*⟩ contains any extra trailing tokens that were not processed by ⟨*skip expression*⟩, appending a default unit of ⟨*unit*⟩. `\argproc_skip:n` does not append a default unit.

\argproc_muskip:n `\argproc_muskip:n {⟨muskip expression⟩}`

Argument processor for use in `\NewDocumentCommand` and friends. Evaluates ⟨*muskip expression*⟩ and stores the result in `\ProcessedArgument`. Issues an error if ⟨*muskip expression*⟩ contains any extra trailing tokens that were not processed by ⟨*muskip expression*⟩.

4 Setting integer, dimen, skip, and muskip variables from user input

4.1 Setting integer variables from user input

\argproc_int_set_from_user:Nn `\argproc_int_set_from_user:Nn ⟨integer variable⟩ {⟨integer expression⟩}`
\argproc_int_set_from_user:(cn|NV|cV)
\argproc_int_gset_from_user:Nn
\argproc_int_gset_from_user:(cn|NV|cV)

Sets ⟨*integer variable*⟩ to the value of ⟨*integer expression*⟩. ⟨*integer expression*⟩ should contain the tokens input by the user. Issues an error if ⟨*integer expression*⟩ contains any extra trailing tokens that were not processed by ⟨*integer expression*⟩.

\argproc_int_add_from_user:Nn `\argproc_int_add_from_user:Nn ⟨integer variable⟩ {⟨integer expression⟩}`
\argproc_int_add_from_user:(cn|NV|cV)
\argproc_int_gadd_from_user:Nn
\argproc_int_gadd_from_user:(cn|NV|cV)

Identical to `\argproc_int_set_from_user:Nn`, but sets ⟨*integer variable*⟩ equal to its current value plus the result of ⟨*integer expression*⟩.

\argproc_int_sub_from_user:Nn
\argproc_int_sub_from_user:(cn|NV|cV)
\argproc_int_gsub_from_user:Nn
\argproc_int_gsub_from_user:(cn|NV|cV)

\argproc_int_sub_from_user:Nn ⟨*integer variable*⟩ {⟨*integer expression*⟩}

Identical to \argproc_int_set_from_user:Nn, but sets ⟨*integer variable*⟩ equal to its current value minus the result of ⟨*integer expression*⟩.

4.2 Setting dimen variables from user input

\argproc_dim_set_from_user:Nnn
\argproc_dim_set_from_user:(cnn|NVn|cVn)
\argproc_dim_gset_from_user:Nnn
\argproc_dim_gset_from_user:(cnn|NVn|cVn)
\argproc_dim_set_from_user:Nn
\argproc_dim_set_from_user:(cn|NV|cV)
\argproc_dim_gset_from_user:Nn
\argproc_dim_gset_from_user:(cn|NV|cV)

\argproc_dim_set_from_user:Nnn ⟨*dimen variable*⟩ {⟨*dimen expression*⟩} {⟨*unit*⟩}
\argproc_dim_set_from_user:Nn ⟨*dimen variable*⟩ {⟨*dimen expression*⟩}

Sets ⟨*dimen variable*⟩ to the value of ⟨*dimen expression*⟩, appending a default unit of ⟨*unit*⟩. ⟨*dimen expression*⟩ should contain the tokens input by the user. When fully expanded, ⟨*unit*⟩ must either match a valid unit of measurement or expand to nothing. Issues an error if ⟨*dimen expression*⟩ contains any extra trailing tokens that were not processed by ⟨*dimen expression*⟩, appending a default unit of ⟨*unit*⟩. \argproc_dim_set_from_user:Nn and friends do not append a default unit.

\argproc_dim_add_from_user:Nnn
\argproc_dim_add_from_user:(cnn|NVn|cVn)
\argproc_dim_gadd_from_user:Nnn
\argproc_dim_gadd_from_user:(cnn|NVn|cVn)
\argproc_dim_add_from_user:Nn
\argproc_dim_add_from_user:(cn|NV|cV)
\argproc_dim_gadd_from_user:Nn
\argproc_dim_gadd_from_user:(cn|NV|cV)

\argproc_dim_add_from_user:Nnn ⟨*dimen variable*⟩ {⟨*dimen expression*⟩} {⟨*unit*⟩}
\argproc_dim_add_from_user:Nn ⟨*dimen variable*⟩ {⟨*dimen expression*⟩}

Identical to \argproc_dim_set_from_user:Nnn, but sets ⟨*dimen variable*⟩ equal to its current value plus the result of ⟨*dimen expression*⟩, appending a default unit of ⟨*unit*⟩. \argproc_dim_add_from_user:Nn and friends do not append a default unit.

\argproc_dim_sub_from_user:Nnn
\argproc_dim_sub_from_user:(cnn|NVn|cVn)
\argproc_dim_gsub_from_user:Nnn
\argproc_dim_gsub_from_user:(cnn|NVn|cVn)
\argproc_dim_sub_from_user:Nn
\argproc_dim_sub_from_user:(cn|NV|cV)
\argproc_dim_gsub_from_user:Nn
\argproc_dim_gsub_from_user:(cn|NV|cV)

\argproc_dim_sub_from_user:Nnn ⟨*dimen variable*⟩ {⟨*dimen expression*⟩} {⟨*unit*⟩}
\argproc_dim_sub_from_user:Nn ⟨*dimen variable*⟩ {⟨*dimen expression*⟩}

Identical to \argproc_dim_set_from_user:Nnn, but sets ⟨*dimen variable*⟩ equal to its current value minus the result of ⟨*dimen expression*⟩, appending a default unit of ⟨*unit*⟩. \argproc_dim_sub_from_user:Nn and friends do not append a default unit.

4.3 Setting skip variables from user input

`\argproc_skip_set_from_user:Nnn`	`\argproc_skip_set_from_user:Nnn` ⟨*skip variable*⟩ {⟨*skip expression*⟩} {⟨*unit*⟩}
`\argproc_skip_set_from_user:(cnn|NVn|cVn)`	
`\argproc_skip_gset_from_user:Nnn`	`\argproc_skip_set_from_user:Nn` ⟨*skip variable*⟩ {⟨*skip expression*⟩}
`\argproc_skip_gset_from_user:(cnn|NVn|cVn)`	
`\argproc_skip_set_from_user:Nn`	
`\argproc_skip_set_from_user:(cn|NV|cV)`	
`\argproc_skip_gset_from_user:Nn`	
`\argproc_skip_gset_from_user:(cn|NV|cV)`	

Sets ⟨*skip variable*⟩ to the value of ⟨*skip expression*⟩, appending a default unit of ⟨*unit*⟩. ⟨*skip expression*⟩ should contain the tokens input by the user. When fully expanded, ⟨*unit*⟩ must either match a valid unit of measurement or expand to nothing. Issues an error if ⟨*skip expression*⟩ contains any extra trailing tokens that were not processed by ⟨*skip expression*⟩, appending a default unit of ⟨*unit*⟩. `\argproc_skip_set_from_user:Nn` and friends do not append a default unit.

`\argproc_skip_add_from_user:Nnn`	`\argproc_skip_add_from_user:Nnn` ⟨*skip variable*⟩ {⟨*skip expression*⟩} {⟨*unit*⟩}
`\argproc_skip_add_from_user:(cnn|NVn|cVn)`	
`\argproc_skip_gadd_from_user:Nnn`	`\argproc_skip_add_from_user:Nn` ⟨*skip variable*⟩ {⟨*skip expression*⟩}
`\argproc_skip_gadd_from_user:(cnn|NVn|cVn)`	
`\argproc_skip_add_from_user:Nn`	
`\argproc_skip_add_from_user:(cn|NV|cV)`	
`\argproc_skip_gadd_from_user:Nn`	
`\argproc_skip_gadd_from_user:(cn|NV|cV)`	

Identical to `\argproc_skip_set_from_user:Nnn`, but sets ⟨*skip variable*⟩ equal to its current value plus the result of ⟨*skip expression*⟩, appending a default unit of ⟨*unit*⟩. `\argproc_skip_add_from_user:Nn` and friends do not append a default unit.

`\argproc_skip_sub_from_user:Nnn`	`\argproc_skip_sub_from_user:Nnn` ⟨*skip variable*⟩ {⟨*skip expression*⟩} {⟨*unit*⟩}
`\argproc_skip_sub_from_user:(cnn|NVn|cVn)`	
`\argproc_skip_gsub_from_user:Nnn`	`\argproc_skip_sub_from_user:Nn` ⟨*skip variable*⟩ {⟨*skip expression*⟩}
`\argproc_skip_gsub_from_user:(cnn|NVn|cVn)`	
`\argproc_skip_sub_from_user:Nn`	
`\argproc_skip_sub_from_user:(cn|NV|cV)`	
`\argproc_skip_gsub_from_user:Nn`	
`\argproc_skip_gsub_from_user:(cn|NV|cV)`	

Identical to `\argproc_skip_set_from_user:Nnn`, but sets ⟨*skip variable*⟩ equal to its current value minus the result of ⟨*skip expression*⟩, appending a default unit of ⟨*unit*⟩. `\argproc_skip_sub_from_user:Nn` and friends do not append a default unit.

4.4 Setting muskip variables from user input

`\argproc_muskip_set_from_user:Nn` `\argproc_muskip_set_from_user:Nn` ⟨*muskip variable*⟩ {⟨*muskip expression*⟩}
`\argproc_muskip_set_from_user:(cn|NV|cV)`
`\argproc_muskip_gset_from_user:Nn`
`\argproc_muskip_gset_from_user:(cn|NV|cV)`

Sets ⟨*muskip variable*⟩ to the value of ⟨*muskip expression*⟩. ⟨*muskip expression*⟩ should contain the tokens input by the user. Issues an error if ⟨*muskip expression*⟩ contains any extra trailing tokens that were not processed by ⟨*muskip expression*⟩.

`\argproc_muskip_add_from_user:Nn` `\argproc_muskip_add_from_user:Nn` ⟨*muskip variable*⟩ {⟨*muskip expression*⟩}
`\argproc_muskip_add_from_user:(cn|NV|cV)`
`\argproc_muskip_gadd_from_user:Nn`
`\argproc_muskip_gadd_from_user:(cn|NV|cV)`

Identical to `\argproc_muskip_set_from_user:Nn`, but sets ⟨*muskip variable*⟩ equal to its current value plus the result of ⟨*muskip expression*⟩.

`\argproc_muskip_sub_from_user:Nn` `\argproc_muskip_sub_from_user:Nn` ⟨*muskip variable*⟩ {⟨*muskip expression*⟩}
`\argproc_muskip_sub_from_user:(cn|NV|cV)`
`\argproc_muskip_gsub_from_user:Nn`
`\argproc_muskip_gsub_from_user:(cn|NV|cV)`

Identical to `\argproc_muskip_set_from_user:Nn`, but sets ⟨*muskip variable*⟩ equal to its current value minus the result of ⟨*muskip expression*⟩.