LaTeX3 入门

ljguo

2022 年 8 月 4 日

1 function

Function ▶\cs_new:Npn(Nn|cn|cpn) ①

```
\ExplSyntaxOn
\cs_new:Npn \l_my_cmd:n#1 {i~love~#1!}
\l_my_cmd:n{fish}
\ExplSyntaxOff
```

i love fish!

Function ▶\cs_new:Npx(Nx|cx|cpx) ②

```
\ExplSyntaxOn
\cs_new:Npx \l_my_cmdx:n#1 {i~love~#1!}
\l_my_cmd:n{fish}
\ExplSyntaxOff
```

i love fish!

演示 3

Function ▶\cs_set:Npn(Nn|cn|cpn) 3

```
\ExplSyntaxOn
\cs_set:Npn \foo_cmd:n #1 {i~love~#1}
\foo_cmd:n{fish}
\ExplSyntaxOff
```

i love fish

2 COLOR

Function ▶\color_group_begin: \color_group_end: 4

```
\ExplSyntaxOn
\cs_meaning:N \color_group_begin: \par
\cs_meaning:N \color_group_end:
\ExplSyntaxOff
```

\begingroup

\endgroup

Function ▶\color_set:nn{<name>}{<color expression>} 5

```
\ExplSyntaxOn
\color_set:nn{main}{red!30!cyan}
\color_select:n{main}
\centering
\LaTeX
\hrule
\ExplSyntaxOff
```

LaTeX

Function ▶\int_format:n \fp_format:n 6

```
\ExplSyntaxOn
\seq_new:N \l_before_seq
\seq_new:N \l_after_seq
\cs_generate_variant:Nn \regex_extract_all:nnN {nxN}
\cs_set:Npn \int_format:n #1#2 {
  \regex_extract_all:nnN {\d{1,#2}} {#1} \l_tmpa_seq
  \seq_use:Nn \l_tmpa_seq{~}
}
\cs_set:Npn \fp_format:n #1#2 {
  \regex_split:nnN {\.} {#1} \l_tmpb_seq
  \regex_extract_all:nxN {\d{1,#2}} {\seq_item:Nn \l_tmpb_seq{1}} \l_before_seq
  \regex_extract_all:nxN {\d{1,#2}} {\seq_item:Nn \l_tmpb_seq{2}} \l_after_seq
  \seq_use:Nn \l_before_seq{~} . \seq_use:Nn \l_after_seq{~}
}
\int_format:n{123456789}{3}\par
\fp_format:n{152.354126859}{4}\par
\int_format:n{23354862}{2}\par
\fp_format:n{0.1010010001000010}{3}
\ExplSyntaxOff
```

123 456 789

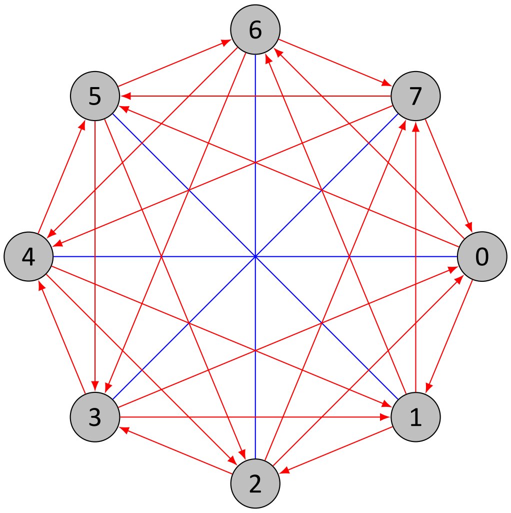
152.3541 2685 9

23 35 48 62

0.101 001 000 100 001 0

Function ▶picture 7

```
\ExplSyntaxOn
\centering
\begin{tikzpicture}
\int_step_inline:nnn{0}{7}
{
  \node[circle,fill=gray!50,draw] (a#1) at (-#1*45\c_colon_str 3){#1};
  \draw[blue] (0,0) -- (a#1);
}
\int_step_inline:nnn{0}{7}
{
  \draw[-latex,red] (a#1) -- (a\int_mod:nn{#1+1}{8});
}
\int_step_inline:nnn{0}{7}
{
  \int_step_inline:nnn{#1+5}{#1+6}
  {
    \draw[-latex,red] (a#1) -- (a\int_mod:nn{##1}{8});
  }
}
\end{tikzpicture}
\ExplSyntaxOff
```



Function ▶expand \d 8

```
\def\a{aaa}
\def\b{\a\a}
\def\c{\b\b}
展开一次:
\expandafter\def\expandafter\d\expandafter{\c}
\meaning\d \par
```

```
展开两次:
\expandafter\expandafter\expandafter\expandafter\expandafter\expandafter\expandafter
\def\expandafter\expandafter\expandafter\expandafter\expandafter\expandafter\expandafter
\d\expandafter\expandafter\expandafter\expandafter\expandafter\expandafter\expandafter
{\expandafter\expandafter\c}
\meaning\d \par
递归展开:
\edef\d{\c}
\meaning\d
```

```
展开一次： macro:->\b \b
展开两次： macro:->\a \a \a \a
递归展开： macro:->aaaaaaaaaaaa
```

Function ▶noexpand \b 9

```
\def\a{a}
\def\b{b}
\def\c{c}
(方法一)
\edef\d{\a\noexpand\b\c}
\meaning\d \par
(方法二)
\toks0={\b}
\edef\d{\a\the\toks0\c}
\meaning\d \par
(方法三)
\expandafter\expandafter\expandafter
\def\expandafter\expandafter\expandafter
\d\expandafter\expandafter\expandafter
{\expandafter\a\expandafter\b\c}
\meaning\d
```

```
(方法一) macro:->a\b c
(方法二) macro:->a\b c
(方法三) macro:->a\b c
```

Function ▶\items 10

```
\newcount\ljguo
\ljguo = 1
{\catcode`\-=13
\catcode`\*=13
\gdef\begitem{\par\noindent\begingroup\catcode`\-=13\catcode`\*=13
\def- {\par\noindent\the\ljguo.\advance\ljguo by 1\hskip 1pt\ignorespaces}
\def* {\par\noindent\hskip4pt$\bullet$\hskip3pt\ignorespaces}}}
\def\enditem{\par\endgroup}
```

```
{\LARGE Markdown}
\vskip 0.3cm
\begitem
- this is item one.
- this is item two.
 * this is iitem one.
 * this is iitem two.
- this is item three.
* this is iitem one.
* this is iitem two.
- this is item four.
\enditem
```

Markdown

1. this is item one.
2. this is item two.
   - this is iitem one.
   - this is iitem two.
3. this is item three.
   - this is iitem one.
   - this is iitem two.
4. this is item four.

Function ▶\getlength 11

```
\def\length#1{\count0=0 \getlength#1\end \number\count0}
\def\getlength#1{\ifx#1\end \let\next=\relax
\else\advance\count0 by 1\relax \let\next=\getlength\fi \next}
\length{asdasf}
```

6

Function ▶\tokenfill 12

```
\def\tokenfill#1{\leaders\hbox to 0.8em{\hss #1 \hss}\hfill}
\noindent\tokenfill{$\triangleright$} \par
begin \tokenfill{$\cdot$} end
```

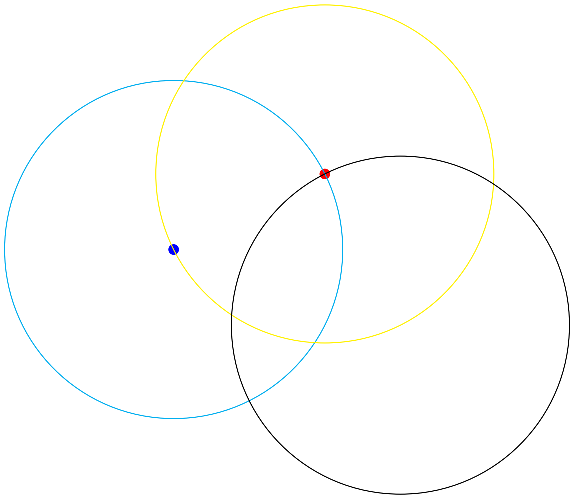
▷ ▷

begin · end

Function ▶\length 13

```
\ExplSyntaxOn
\makeatletter
\cs_set:Npn \pgf_length:N #1 {
  \fp_set:Nn \l_tmpa_fp {\dim_to_fp:n{\pgf@x}}
  \fp_set:Nn \l_tmpb_fp {\dim_to_fp:n{\pgf@y}}
  \tl_set:Nx #1{
    \fp_to_dim:n{\fp_eval:n{({\l_tmpa_fp}^2+{\l_tmpb_fp}^2)^(0.5)}}
  }
}
\cs_set_eq:NN \length \pgf_length:N
\def\getlengthtomarco#1#2{
  \path #1;
  \length#2
}
\makeatother
\ExplSyntaxOff

\begin{tikzpicture}
  \fill[red] (3,2) coordinate(a) circle(2pt);
  \fill[blue] (1,1) coordinate(b) circle(2pt);
  \coordinate(c) at ($(b)-(a)$);
  \path (c);
  \length\r %\r = length "(b) - (a)"
  \draw[cyan] (b) circle (\r);
  \draw[yellow] (a) circle (\r);
  \getlengthtomarco{(1,2)}{\r}
  \draw[] (4,0) circle (\r);
\end{tikzpicture}
```



Function ▶\@ifnextchar 14

```
\makeatletter
\def\@cmd[#1]#2{\textcolor{red}{#1} and (#2)}
\def\cmd{
  \@ifnextchar[{\@cmd}{\@cmd[default]}
}
```

```
\cmd{aaa}\par
\cmd[bbb]{aaa}

\def\@cmd[#1]#2{\c@md{} and \textcolor{red}{#1} and (#2)}
\def\cmd#1{
  \def\c@md{#1}\@ifnextchar[{\@cmd}{\@cmd[default]}
}
\cmd{aaa}{ccc}\par
\cmd{aaa}[bbb]{ccc}
\makeatother
```

default and (aaa)

bbb and (aaa)

aaa and default and (ccc)

aaa and bbb and (ccc)

Function ▶\@ifstar 15

```
\catcode`@=11
\def\cmd{\@ifstar{\textcolor{red}}{\textcolor{blue}}}
\cmd*{aaa}\par
\cmd{aaa}
\catcode`@=12
```

aaa

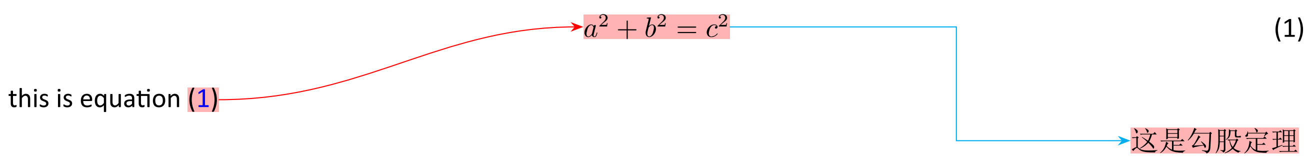
aaa

Function ▶\mark 16

```
\def\mark#1#2{%
\tikz[remember picture,baseline]{%
\node[inner sep=0pt,outer sep=0pt,anchor=base,fill=red!30,align=center] (#1){#2};}}

\begin{align}\label{eq:1}
  \mark{label}{$a^2 + b^2 = c^2$}
\end{align}
this is equation \mark{ref}{\eqref{eq:1}}
\tikz[overlay,remember picture]\draw[red,Stealth-](label.180) to[out=180,in=0] (ref.0);

\hfill \mark{margin}{这是勾股定理}
\tikz[overlay,remember picture]\draw[cyan,-Stealth](label.0) --++(3,0)|- (margin.180);
```



Function ▶Array 17

```
\ExplSyntaxOn
\seq_new:N \l_node_row_seq
\seq_new:N \l_node_tmp_seq
\cs_set:Npn \GetArray #1 {
  \seq_set_split:Nnn \l_node_row_seq {;} {#1}
  \int_step_inline:nn{\seq_count:N \l_node_row_seq}
  {
    \seq_if_exist:cF {l_node_row_##1_seq}
    {
      \seq_new:c {l_node_row_##1_seq}
    }
    \exp_args:Ncx\seq_set_from_clist:Nn {l_node_row_##1_seq} {\seq_item:Nn \l_node_row_seq{##1}}
  }
}

\cs_set:Npn \PrintArray [#1][#2] {
  \tl_if_empty:nTF { #1 }
  {
    \tl_if_empty:nTF { #2 }
    {
      \seq_use:Nn \l_node_row_seq {,}
    }
    {
      \int_step_inline:nn{\seq_count:N \l_node_row_seq}
      {
        \seq_put_right:Nn \l_node_tmp_seq {\seq_item:cn {l_node_row_##1_seq}{#2}}
      }
      \seq_use:Nn \l_node_tmp_seq {,}
    }
  }
  {
    \tl_if_empty:nTF {#2}
    {
      \seq_use:cn {l_node_row_#1_seq}{,}
    }
    {
      \seq_item:cn {l_node_row_#1_seq}{#2}
    }
  }
}
\ExplSyntaxOff

\GetArray
{
  A,B,C;
  D,E,F;
  G,H,I
}
\PrintArray[][]\par
\PrintArray[1][]\par
\PrintArray[][2]\par
\PrintArray[1][2]
```

A,B,C,D,E,F,G,H,I

A,B,C

B,E,H

B