

MathJax の使い方

1 MathJax とは？

MathJax はウェブページの製作者がページに数式を埋め込めるようにする JavaScript ライブラリです。閲覧者側では数式を閲覧するのに何も必要としません。

ブラウザ: MathJax は、Edge、Firefox、Chrome、Safari、Opera、ほとんどのモバイルブラウザを含む、すべてのモダンブラウザで動作します。

(MathJax ヘルプより)

MathJax はウェブページにきれいな数式を表示できるようにするためのソフトウェアです。2010年に最初のバージョンがリリースされ、現在 MathJax はウェブで数式を扱う方法としての標準になりつつあります。MathJax の本体は JavaScript で書かれていますが、MathJax を利用する上で JavaScript についての知識はとくに必要ありません。ソフトウェアのインストール等も必要ありませんので、MathJax はどなたでも手軽に利用することができます。以下で MathJax の簡単な使い方を見ていきます。

2 HTML ファイルの作り方

MathJax はウェブページで使用するものですので、ウェブページを作成するための HTML の知識が少し必要です。まずはもっとも簡単な HTML ファイルを書いてみましょう。テキストエディターを開いて次のコードを入力し、mathjax.html という名前で保存してみてください。(拡張子が .html になっていればファイル名はなんでもよいです。)

```
mathjax.html
<html>
<head>
<title>MathJax の練習</title>
</head>
<body>
MathJax の使い方を練習します。
</body>
</html>
```

ファイルを保存したら、このファイルをウェブブラウザで開いてみましょう。「MathJax の練習」というタイトルの付いたページが表示されるはずです。また、「MathJax の使い方を練習します。」という文章が表示されているはずです。以下では、`<body>~</body>` の部分に数式を表すコード (TeX コマンド) を入力し、文章だけでなく数式も表示できるようにしていきます。

3 MathJax の使い方

数式を入力できるようにするには先ほど作成した HTML ファイルを次のように修正します。(バックslash記号 \ が円記号 ¥ になる人は、\ を ¥ に読みかえてください。どちらを使っても同じです。)

```
mathjax.html
<html>
<head>
<script defer src="https://cdn.jsdelivr.net/npm/mathjax@4/tex-mml-cthtml.js">
</script>
<title>MathJax の練習</title>
</head>
<body>
MathJax の使い方を練習します。<br>
一次方程式  $(ax+b=0)$  の解は

$$x = -\frac{b}{a}$$

です。
</body>
</html>
```

- <head>~</head> の部分に <script defer src="https://cdn.jsdelivr.net/npm/mathjax@4/tex-mml-cthtml.js"></script> という命令を追加しました。これは MathJax を使うために必要な命令です。
- 文章中 (数式以外の場所) で改行したいときは、改行する場所で
 と入力します。
- 文章中に数式を入力したいときは \ (と \) で数式の両側を囲みます。
- 別行立ての数式を書きたい場合は、数式を \[と \] で囲みます。
- 分数を書きたい場合は $\frac{\text{分子}}{\text{分母}}$ というコマンドを使います。

HTML ファイルを上書き保存したら、ブラウザの更新ボタンを押して表示を更新してください。

表示結果

MathJax の使い方を練習します。

一次方程式 $ax + b = 0$ の解は

$$x = -\frac{b}{a}$$

です。

- 数式中では半角スペースや改行はいくつ入力しても表示結果に影響しませんので注意してください。
- 数式中に小さなスペースを入れたい場合、もっとも簡単には _ と入力します (_ は半角スペース)。
- 数式中で改行することはできません。複数行におよぶ数式を入力する方法については後で解説します。

上の例からわかるように、数式に関する特殊なコマンドは `\` から始まります（もしくは `¥`）。これらは TeX（テフ）という文書作成ソフトで使われるコマンドのため TeX コマンドと呼ばれることがあります。

次の例を入力してみてください。（例では `<body>~</body>` 以外の部分を省略しますが、実際に作成するファイルでは省略しないでください。）

```
mathjax.html
<body>
二次方程式  $(ax^2+bx+c=0)$  の解は
\[
x = \frac{-b \pm \sqrt{b^2-4ac}}{2a} \tag{1}
\]
です。
</body>
```

表示結果は次のようになるはずです。

表示結果

二次方程式 $ax^2 + bx + c = 0$ の解は

$$x = \frac{-b \pm \sqrt{b^2 - 4ac}}{2a} \tag{1}$$

です。

- 上付きの添え字（指数）は `^{\}` で出せます。例： $e^x \Rightarrow e^x$
- `\pm` : ±
- `\sqrt{\}` : $\sqrt{\quad}$ （なお n 乗根の記号 $\sqrt[n]{\quad}$ は `\sqrt[n]{\quad}` で出すことができます。）
- `\tag{\}` で別行立ての数式に数式番号を振ることができます。例えば `\tag{*}` と入力すれば、(*) という形の数式番号が付きます。

数式コマンドの後ろにアルファベットが続く場合は、そのコマンドの直後に半角スペースを入れる必要がありますので注意してください。

例：`\pm a` $\Rightarrow \pm a$ （スペースを入れずに `\pma` と書くとエラーになります。）

ただしコマンドのあとに記号や数字が続く場合は、スペースを入れなくても大丈夫です。数式中での半角スペースは無視されるので、スペースを入れても入れなくても、表示結果に差はありません。

4 数式入力の例

練習 1

ソースコード

```
\[
\sum_{k=1}^n a_{k} = a_{1} + a_{2} + \dots + a_{n}
\]
```

表示結果

$$\sum_{k=1}^n a_k = a_1 + a_2 + \dots + a_n$$

- `\sum_{k=1}^n` : 総和記号 (シグマ) 和の下限 `_k=1` や上限 `^n` が必要ない場合はそれらを省略します。
- 下付きの添え字は `_k` で出せます。
- `\dots` : 点々

練習 2

ソースコード

```
ガウス積分
\[
\int_{-\infty}^{\infty} e^{-x^2} \, dx = \sqrt{\pi}
\]
```

表示結果

ガウス積分

$$\int_{-\infty}^{\infty} e^{-x^2} dx = \sqrt{\pi}$$

- `\int_{-\infty}^{\infty}` : 積分記号 \int (積分の下限や上限が必要ない場合はそれらを省略できます。)
- `\infty` : 無限大記号 ∞
- `\,` : すごく小さいスペース。このスペースを入れるかどうかは好みです。入れなくても構いません。いろいろな大きさのスペースを入力する方法については付録を参考にしてください。
- `\pi` : π その他のギリシア文字については付録を参考にしてください。

練習 3

ソースコード

関数 $f(x)$ の導関数は

$\left[$

$$f'(x) = \lim_{\Delta x \rightarrow 0} \frac{f(x + \Delta x) - f(x)}{\Delta x}$$

\backslash

である。

表示結果

関数 $f(x)$ の導関数は

$$f'(x) = \lim_{\Delta x \rightarrow 0} \frac{f(x + \Delta x) - f(x)}{\Delta x}$$

である。

- $\lim_{\{}$: 極限
- \to : 矢印
- Δ , \varDelta : Δ , Δ (ギリシア文字の大文字は **var** を付けると斜体になります。)

練習 4

ソースコード

$\left[$

$$\int \tan \theta \, d\theta = \int \frac{\sin \theta}{\cos \theta} \, d\theta = -\log |\cos \theta| + C$$

\backslash

表示結果

$$\int \tan \theta \, d\theta = \int \frac{\sin \theta}{\cos \theta} \, d\theta = -\log |\cos \theta| + C$$

- \sin , \cos , \tan , \log : \sin , \cos , \tan , \log
(バックスラッシュや円記号を付けずに、単に **sin** とだけ書くと *sin* のようにイタリック体で表示され、 $s \times i \times n$ の意味となってしまいます。)
- θ : θ

練習 5

ソースコード

```
\begin{align}
\cos 2\theta &= \cos^2 \theta - \sin^2 \theta \\
&= 2\cos^2 \theta - 1 \\
&= 1 - 2\sin^2 \theta
\end{align}
```

表示結果

$$\begin{aligned}\cos 2\theta &= \cos^2 \theta - \sin^2 \theta \\ &= 2\cos^2 \theta - 1 \\ &= 1 - 2\sin^2 \theta\end{aligned}$$

- `\begin{align}~\end{align}` : 複数行からなる別行立ての数式を書くことができます。改行の命令は `\\` です。また、`&` を置くとそこで数式の表示位置をそろえてくれます。もし数式どうしの間隔を変更したい場合は、改行の命令を `\\[5pt]` や `\\[-3pt]` 等へ変更することで調整できます。

練習 6

ソースコード

```
\[
|x| = \begin{cases}
x & x \geq 0 \text{ のとき} \\
-x & x < 0 \text{ のとき}
\end{cases}
\]
```

表示結果

$$|x| = \begin{cases} x & x \geq 0 \text{ のとき} \\ -x & x < 0 \text{ のとき} \end{cases}$$

- `\begin{cases}~\end{cases}` : 場合分け 各行に `&` を置くと、その場所に区切りが入り、そこで位置をそろえてくれます。また、`\\` で次の行へ移ります。
- `\lt`, `\gt`, `\le`, `\ge` : 不等号 $<$, $>$, \leq , \geq

練習 7

ソースコード

```
\(n \times n\) 行列  
\[  
A =  
\begin{pmatrix}  
a_{11} & a_{12} & \dots & a_{1n} \\ a_{21} & a_{22} & \dots & a_{2n} \\ \vdots & \vdots & \ddots & \vdots \\ a_{n1} & a_{n2} & \dots & a_{nn}\end{pmatrix}  
\]
```

が逆行列 (A^{-1}) をもつための必要十分条件は、 $(\det A \neq 0)$ である。

表示結果

$n \times n$ 行列

$$A = \begin{pmatrix} a_{11} & a_{12} & \dots & a_{1n} \\ a_{21} & a_{22} & \dots & a_{2n} \\ \vdots & \vdots & \ddots & \vdots \\ a_{n1} & a_{n2} & \dots & a_{nn} \end{pmatrix}$$

が逆行列 A^{-1} をもつための必要十分条件は、 $\det A \neq 0$ である。

- `\times` : \times
- `\begin{pmatrix}`~`\end{pmatrix}` : 行列 & で列を区切り `\\` で次の行へ移ります。
- `\dots`, `\vdots`, `\ddots` : 点々
- `\det` : det (行列式)
- `\neq` : \neq

行列を囲む括弧の形は次のように変えることができます。

- `pmatrix` : 丸括弧 $()$
- `bmatrix` : 角括弧 $[\]$
- `Bmatrix` : 波括弧 $\{ \}$
- `vmatrix` : 縦棒 $| |$
- `Vmatrix` : 二重の縦棒 $|| | |$
- `matrix` : 括弧なし

5 MathJax をカスタマイズする

MathJax の設定を変更する方法について簡単に紹介します。まず、次の命令が書き込まれた設定ファイル `config.js` を作成してみてください。(mathjax.html と同じディレクトリに保存します。)

```
config.js
window.MathJax = {
  tex: {
    tags: "ams",
    macros: {
      x: "\\times",
      bm: ["\\boldsymbol{#1}",1],
      dd: ["\\frac{\\partial #1}{\\partial #2}",2]
    }
  },
  chtml: {
    scale: 0.9,
    mtextInheritFont: true
  }
};
```

この設定ファイルでは以下のような設定を行っています。

- `tags` の行は、数式番号を自動的に振ってくれるようにする設定になっています。別行立ての数式を書くときに、`\[` と `\]` の代わりに `\begin{equation}` と `\end{equation}` を使うと、数式の右端に自動的に数式番号が振られるようになります。
- `macros` の部分では「マクロ」を定義しています。
 - `x` の行では `\times` と入力するのが長いので `\x` と入力するだけでよいようにしています。
 - `bm` の行では、引数を 1 つとる命令 `\bm{}` を定義しています。`\boldsymbol{A}` と入力する代わりに `\bm{A}` と入力すればよくなります。(`\boldsymbol{}` は太字を出力する命令です。)
 - `dd` の行では、引数を 2 つとる命令 `\dd{ }{ }` を定義しています。`\frac{\\partial A}{\\partial B}` と入力する代わりに `\dd{A}{B}` と入力すればよくなります。(`\partial` は偏微分記号 ∂ です。)マクロの名前は自分で自由につけられますが、記号や数字を含んだ名前にすることはできません。
- `scale` の行では、数式環境中の文字の大きさをデフォルトの 90% に変更しています。
- `mtextInheritFont` の行では、数式環境中の数式以外のフォント（日本語や数式番号）が本文中のフォントと同じになるように設定しています。この項目のデフォルトは `false` です。

他にもいろいろな設定を行うことができますが、詳しくは MathJax のドキュメントを読んでみてください。

- <https://docs.mathjax.org/en/latest/options/index.html>
- <https://docs.mathjax.org/en/latest/options/input/tex.html>
- <https://docs.mathjax.org/en/latest/options/output/index.html>

では、上で定義したマクロを使って次のような HTML ファイルを作成してみましょう。

```
mathjax.html
<html>
<head>
<script defer src="config.js"></script>
<script defer src="https://cdn.jsdelivr.net/npm/mathjax@4/tex-mml-ctml.js">
</script>
<title>MathJax の練習</title>
</head>
<body>
ベクトル場  $\mathbf{B}(x,y,z)$  が
\begin{equation}
\mathbf{B} = \nabla \times \mathbf{A} \label{a}
\end{equation}
という形に書けると、その発散
\begin{equation}
\nabla \cdot \mathbf{B} = \frac{\partial B_x}{\partial x} + \frac{\partial B_y}{\partial y} + \frac{\partial B_z}{\partial z}
\end{equation}
は 0 になる。式\eqref{a}に現れる  $\mathbf{A}$  を、 $\mathbf{B}$  のベクトルポテンシャルという。
</body>
</html>
```

- `\nabla` : ナブラ演算子 ∇
- `\cdot` : 中黒 \cdot

`<head>~</head>` の中で MathJax を読み込む前に `config.js` を読み込ませています。HTML ファイルと設定ファイルの両方に間違いがなければ、次のような結果が表示されると思います。

表示結果

ベクトル場 $\mathbf{B}(x,y,z)$ が

$$\mathbf{B} = \nabla \times \mathbf{A} \tag{1}$$

という形に書けると、その発散

$$\nabla \cdot \mathbf{B} = \frac{\partial B_x}{\partial x} + \frac{\partial B_y}{\partial y} + \frac{\partial B_z}{\partial z} \tag{2}$$

は 0 になる。式 (1) に現れる \mathbf{A} を、 \mathbf{B} のベクトルポテンシャルという。

- 式に `\label{}` で任意の名前を付け `\eqref{}` で数式番号を参照することができます。(上の例では最初の式に「a」という名前を付け、`\eqref{a}` でその名前の式の番号を参照しています。)
- 数式番号を自動で振ってほしくない式がある場合は、その行の最後に `\notag` と書きます。
- 実際にブラウザの表示を確認すると、式番号に使われているフォントが本文のフォントと同じになっていることがわかると思います。また、数式環境中の文字の大きさが少し小さくなっているはずですが。

6 演習問題

ヒントや付録を参考にコードを入力し、以下の数式を表示してみましょう。

問題 1

オイラーの公式

$$e^{i\theta} = \cos \theta + i \sin \theta$$

問題 2

テイラー展開

$$f(x) = \sum_{n=0}^{\infty} \frac{f^{(n)}(a)}{n!} (x-a)^n$$

問題 3

正規分布

$$f(x) = \frac{1}{\sqrt{2\pi\sigma^2}} \exp\left(-\frac{(x-\mu)^2}{2\sigma^2}\right)$$

- `\pi`, `\sigma`, `\mu` : ギリシア文字 π , σ , μ
- `\exp` : \exp
- `\left(` `\right)` : 大きさを自動で調整してくれる括弧、`\left` が左用、`\right` が右用、丸括弧 `()` を角括弧 `[]` や波括弧 `\{\}` などに置き換えることもできます。必ず左右セットで使ってください。

問題 4

ニュートンの運動方程式

$$m \frac{d^2 \vec{r}}{dt^2} = \vec{F}$$

- `\vec{}{}` : 上に付く矢印
- `\overrightarrow{}{}` : 上に付く長い矢印 (記号の大きさに合わせて長さが変わる。例: \overrightarrow{AB})

問題 5

ラグランジュの運動方程式

$$\frac{d}{dt} \left(\frac{\partial \mathcal{L}}{\partial \dot{q}} \right) - \frac{\partial \mathcal{L}}{\partial q} = 0$$

- `\partial` : ∂
- `\mathcal{}{}` : カリグラフィックフォント (この問題では `\mathcal{L}`)
- `\dot{}{}` : 記号の上に付く点

問題 6

フーリエ変換

$$\hat{f}(\xi) = \int_{\mathbb{R}^n} f(x) e^{-2\pi i x \cdot \xi} dx$$

- `\hat{}` : 記号の上に付く ^ (ハット)
- `\xi` : ξ
- `\mathbb{}` : 黒板太文字
- `\cdot` : 中黒

問題 7

コーシーの積分公式

$$f(\alpha) = \frac{1}{2\pi i} \oint_C \frac{f(z)}{z - \alpha} dz$$

- `\alpha` : α
- `\oint` : \oint

問題 8

ガウスの定理

$$\iiint_V \nabla \cdot \mathbf{A} dV = \iint_{\partial V} \mathbf{A} \cdot \mathbf{n} dS$$

- `\iint`, `\iiint`, `\iiint` : \iint , \iiint , \iiint
- `\nabla` : ナブラ演算子 ∇
- `\boldsymbol{}` : 太文字

問題 9

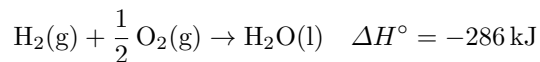
シュレーディンガー方程式

$$i\hbar \frac{\partial}{\partial t} \psi(\mathbf{r}, t) = \left(-\frac{\hbar^2}{2m} \nabla^2 + V(\mathbf{r}, t) \right) \psi(\mathbf{r}, t)$$

- `\hbar` : \hbar
- `\psi` : ψ
- `\biggl(` `\biggr)` : 大きな括弧、左括弧は `\biggl`, `\Bigl`, `\biggl`, `\Biggl` の順番で大きくできます。右括弧に対しては `\biggr`, `\Bigr`, `\biggr`, `\Biggr` を使ってください。また、括弧内に大きい記号を置きたい場合は `\bigm`, `\Bigm`, `\biggm`, `\Biggm` を使います (例えば長い縦棒 $|$ を括弧の真ん中に置きたい場合)。

問題 10

熱化学方程式



- `\mathrm{}` : ローマン体フォント (数式中では自動的にイタリック体が使われるため、元素記号や単位などを書く場合は `\mathrm{}` で囲みます。)
- `\circ` : 小さな白丸 例: 45°
- `\,` : 小さなスペース 数値と単位の間に入れるとよいでしょう。

問題 11

集合記号など

$$A \cap B = \{x \mid x \in A \wedge x \in B\}$$

- `\{ \}` : 波括弧 `{}` (波括弧は特別で `{}` と書くだけでは何も表示されません。)
- `\cap`, `\cup`, `\wedge`, `\vee` : \cap , \cup , \wedge , \vee
- `\in`, `\ni`, `\notin`, `\subset`, `\supset` : \in , \ni , \notin , \subset , \supset
- `\emptyset`, `\forall`, `\exists`, `\neg` : \emptyset , \forall , \exists , \neg

問題 12

二項係数

$${}_n C_r = \binom{n}{r} = \frac{n!}{r!(n-r)!}$$

- 記号の左に添え字を付けたい場合は、空の波括弧を使うとよいでしょう。例: $\binom{x}{z}^A \Rightarrow \frac{A}{z} X$
- `\binom{x}{y}` : 二項係数 $\binom{x}{y}$

問題 13

マクスウェル方程式

$$\nabla \cdot \mathbf{E} = \frac{\rho}{\epsilon_0},$$

$$\nabla \cdot \mathbf{B} = 0,$$

$$\nabla \times \mathbf{E} = -\frac{\partial \mathbf{B}}{\partial t}$$

$$\nabla \times \mathbf{B} = \mu_0 \mathbf{i} + \frac{1}{c^2} \frac{\partial \mathbf{E}}{\partial t}$$

- `\rho`, `\varepsilon`, `\mu` : ρ , ϵ , μ
- `\times` : \times
- `\begin{align} \sim \end{align}` の環境で `&` を複数書くと、複数の位置で数式をそろえられます。ただし、このような使い方をする場合には、隣り合った数式どうしの区切りを示すために、それらの間へもう 1 つ `&` を置く必要があります。(この問題では 1 行につき 3 つの `&` を使っています。)

A 付録：数式コマンド一覧

よく使われるコマンドをまとめておきます。ここで紹介するもの以外にも多くのコマンドがあります。

スペース

<code>\l</code>	小さいスペース
<code>\quad</code>	大きいスペース
<code>\qqquad</code>	<code>\quad</code> の 2 倍
<code>\,</code>	<code>\quad</code> の 3/18 倍
<code>\:</code>	<code>\quad</code> の 4/18 倍
<code>\;</code>	<code>\quad</code> の 5/18 倍
<code>\!</code>	<code>\quad</code> の -3/18 倍 (負のスペース)

ギリシア文字 (小文字)

<code>\alpha</code>	α	<code>\zeta</code>	ζ	<code>\lambda</code>	λ	<code>\varpi</code>	ϖ	<code>\upsilon</code>	υ
<code>\beta</code>	β	<code>\eta</code>	η	<code>\mu</code>	μ	<code>\rho</code>	ρ	<code>\phi</code>	ϕ
<code>\gamma</code>	γ	<code>\theta</code>	θ	<code>\nu</code>	ν	<code>\varrho</code>	ϱ	<code>\varphi</code>	φ
<code>\delta</code>	δ	<code>\vartheta</code>	ϑ	<code>\xi</code>	ξ	<code>\sigma</code>	σ	<code>\chi</code>	χ
<code>\epsilon</code>	ϵ	<code>\iota</code>	ι	<code>o</code>	o	<code>\varsigma</code>	ς	<code>\psi</code>	ψ
<code>\varepsilon</code>	ε	<code>\kappa</code>	κ	<code>\pi</code>	π	<code>\tau</code>	τ	<code>\omega</code>	ω

ギリシア文字 (大文字)

<code>\Gamma</code>	Γ	<code>\Lambda</code>	Λ	<code>\Sigma</code>	Σ	<code>\Psi</code>	Ψ
<code>\varGamma</code>	\varGamma	<code>\varLambda</code>	\varLambda	<code>\varSigma</code>	\varSigma	<code>\varPsi</code>	\varPsi
<code>\Delta</code>	Δ	<code>\Xi</code>	Ξ	<code>\Upsilon</code>	Υ	<code>\Omega</code>	Ω
<code>\varDelta</code>	\varDelta	<code>\varXi</code>	\varXi	<code>\varUpsilon</code>	\varUpsilon	<code>\varOmega</code>	\varOmega
<code>\Theta</code>	Θ	<code>\Pi</code>	Π	<code>\Phi</code>	Φ		
<code>\varTheta</code>	\varTheta	<code>\varPi</code>	\varPi	<code>\varPhi</code>	\varPhi		

括弧

<code>(x)</code>	(x)	<code>\langle x \rangle</code>	$\langle x \rangle$	<code> x </code>	$ x $
<code>[x]</code>	$[x]$	<code>\lfloor x \rfloor</code>	$\lfloor x \rfloor$	<code>\ x\ </code>	$\ x\ $
<code>\{x\}</code>	$\{x\}$	<code>\lceil x \rceil</code>	$\lceil x \rceil$		

※ 左括弧の前に `\bigl`, `\Bigl`, `\biggl`, `\Biggl` を置くと、この順番で括弧が大きくなります。右括弧に対しては `\bigr`, `\Bigr`, `\biggr`, `\Biggr` を使います。また、`\left` や `\right` を左括弧や右括弧の前に置くと、適切な括弧の大きさを自動的に選んでくれます。

二項演算子

<code>+</code>	$+$	<code>\ast</code>	$*$	<code>/</code>	$/$	<code>\cap</code>	\cap	<code>\otimes</code>	\otimes
<code>-</code>	$-$	<code>\star</code>	\star	<code>\backslash</code>	\backslash	<code>\cup</code>	\cup	<code>\oslash</code>	\oslash
<code>\pm</code>	\pm	<code>\cdot</code>	\cdot	<code>\setminus</code>	\setminus	<code>\sqcap</code>	\sqcap	<code>\odot</code>	\odot
<code>\mp</code>	\mp	<code>\bullet</code>	\bullet	<code>\wr</code>	\wr	<code>\sqcup</code>	\sqcup	<code>\dagger</code>	\dagger
<code>\times</code>	\times	<code>\circ</code>	\circ	<code>\wedge</code>	\wedge	<code>\oplus</code>	\oplus	<code>\ddagger</code>	\ddagger
<code>\div</code>	\div	<code>\bigcirc</code>	\bigcirc	<code>\vee</code>	\vee	<code>\ominus</code>	\ominus	<code>\amalg</code>	\amalg

関係演算子

<code>=</code>	$=$	<code>\cong</code>	\cong	<code>\lt</code>	$<$	<code>\lesssim</code>	\lesssim
<code>\neq</code>	\neq	<code>\propto</code>	\propto	<code>\gt</code>	$>$	<code>\gtrsim</code>	\gtrsim
<code>\doteq</code>	\doteq	<code>\varpropto</code>	\varpropto	<code>\ll</code>	\ll	<code>\subset</code>	\subset
<code>\doteqdot</code>	\doteqdot	<code>\perp</code>	\perp	<code>\gg</code>	\gg	<code>\supset</code>	\supset
<code>\equiv</code>	\equiv	<code>\mid</code>	\mid	<code>\lll</code>	\lll	<code>\subseteq</code>	\subseteq
<code>\sim</code>	\sim	<code>\shortmid</code>	\shortmid	<code>\ggg</code>	\ggg	<code>\supseteq</code>	\supseteq
<code>\backsimeq</code>	\backsimeq	<code>\parallel</code>	\parallel	<code>\le, \leq</code>	\leq	<code>\subseteqq</code>	\subseteqq
<code>\simeq</code>	\simeq	<code>\shortparallel</code>	\shortparallel	<code>\ge, \geq</code>	\geq	<code>\supseteqq</code>	\supseteqq
<code>\backsimeq</code>	\backsimeq	<code>\therefore</code>	\therefore	<code>\leqq</code>	\leqq	<code>\in</code>	\in
<code>\eqsim</code>	\eqsim	<code>\because</code>	\because	<code>\geqq</code>	\geqq	<code>\ni</code>	\ni
<code>\approx</code>	\approx	<code>\risingdotseq</code>	\risingdotseq	<code>\leqslant</code>	\leqslant	<code>\notin</code>	\notin
<code>\approxeq</code>	\approxeq	<code>\fallingdotseq</code>	\fallingdotseq	<code>\geqslant</code>	\geqslant	<code>\backepsilon</code>	\backepsilon

※ `\not` で斜線を重ねることができます。例：`\not\equiv` ⇒ \neq

関数等

<code>\sin</code>	\sin	<code>\sinh</code>	\sinh	<code>\deg</code>	\deg	<code>\inf</code>	\inf
<code>\cos</code>	\cos	<code>\cosh</code>	\cosh	<code>\dim</code>	\dim	<code>\sup</code>	\sup
<code>\tan</code>	\tan	<code>\tanh</code>	\tanh	<code>\hom</code>	hom	<code>\liminf</code>	\liminf
<code>\cot</code>	\cot	<code>\coth</code>	\coth	<code>\ker</code>	\ker	<code>\limsup</code>	\limsup
<code>\sec</code>	\sec	<code>\exp</code>	\exp	<code>\bmod</code>	mod	<code>\det</code>	\det
<code>\csc</code>	\csc	<code>\log</code>	\log	<code>\pmod{n}</code>	$(\text{mod } n)$	<code>\gcd</code>	\gcd
<code>\arcsin</code>	\arcsin	<code>\ln</code>	\ln	<code>\lim</code>	\lim	<code>\Pr</code>	Pr
<code>\arccos</code>	\arccos	<code>\lg</code>	\lg	<code>\min</code>	\min		
<code>\arctan</code>	\arctan	<code>\arg</code>	\arg	<code>\max</code>	\max		

記号

<code>\emptyset</code>	\emptyset	<code>\Bbbk</code>	\mathbb{k}	<code>\triangle</code>	\triangle	<code>\top</code>	\top
<code>\varnothing</code>	\varnothing	<code>\varkappa</code>	\varkappa	<code>\square</code>	\square	<code>\bot</code>	\perp
<code>\infty</code>	∞	<code>\ell</code>	ℓ	<code>\blacksquare</code>	\blacksquare	<code>\diagup</code>	\diagup
<code>\aleph</code>	\aleph	<code>\Re</code>	\Re	<code>\bigstar</code>	\bigstar	<code>\diagdown</code>	\diagdown
<code>\complement</code>	\complement	<code>\Im</code>	\Im	<code>\spadesuit</code>	\spadesuit	<code>\forall</code>	\forall
<code>\partial</code>	∂	<code>\mho</code>	\mho	<code>\heartsuit</code>	\heartsuit	<code>\exists</code>	\exists
<code>\digamma</code>	\digamma	<code>\eth</code>	\eth	<code>\diamondsuit</code>	\diamondsuit	<code>\nexists</code>	\nexists
<code>\hbar</code>	\hbar	<code>\prime</code>	\prime	<code>\clubsuit</code>	\clubsuit	<code>\neg, \lnot</code>	\neg
<code>\hslash</code>	\hslash	<code>\backprime</code>	\backprime	<code>\angle</code>	\angle	<code>\sharp</code>	\sharp
<code>\imath</code>	\imath	<code>\surd</code>	\surd	<code>\measuredangle</code>	\measuredangle	<code>\flat</code>	\flat
<code>\jmath</code>	\jmath	<code>\nabla</code>	∇	<code>\sphericalangle</code>	\sphericalangle	<code>\natural</code>	\natural

大きな記号

<code>\sum</code>	\sum	<code>\bigsqcup</code>	\bigcup	<code>\int</code>	\int
<code>\prod</code>	\prod	<code>\bigwedge</code>	\bigwedge	<code>\oint</code>	\oint
<code>\coprod</code>	\coprod	<code>\bigvee</code>	\bigvee	<code>\iint</code>	\iint
<code>\bigcap</code>	\bigcap	<code>\bigoplus</code>	\bigoplus	<code>\iiint</code>	\iiint
<code>\bigcup</code>	\bigcup	<code>\bigotimes</code>	\bigotimes	<code>\iiiiint</code>	\iiiiint
<code>\biguplus</code>	\biguplus	<code>\bigodot</code>	\bigodot	<code>\idotsint</code>	$\int \cdots \int$

※ `_{\{ }^{\{ }` で下限や上限を付けることができます。

上に付く記号

<code>\vec{x}</code>	\vec{x}	<code>\acute{x}</code>	\acute{x}	<code>\overrightarrow{xyz}</code>	\overrightarrow{xyz}
<code>\bar{x}</code>	\bar{x}	<code>\grave{x}</code>	\grave{x}	<code>\overleftarrow{xyz}</code>	\overleftarrow{xyz}
<code>\tilde{x}</code>	\tilde{x}	<code>\dot{x}</code>	\dot{x}	<code>\widetilde{xyz}</code>	\widetilde{xyz}
<code>\breve{x}</code>	\breve{x}	<code>\ddot{x}</code>	\ddot{x}	<code>\widehat{xyz}</code>	\widehat{xyz}
<code>\hat{x}</code>	\hat{x}	<code>\ddd{x}</code>	\ddd{x}	<code>\overline{xyz}</code>	\overline{xyz}
<code>\check{x}</code>	\check{x}	<code>\ddddot{x}</code>	\ddddot{x}	<code>\underline{xyz}</code>	\underline{xyz}

点々

<code>\cdots</code>	\cdots
<code>\ldots</code>	\ldots
<code>\vdots</code>	\vdots
<code>\ddots</code>	\ddots

矢印

<code>\rightarrow, \to</code>	\rightarrow	<code>\uparrow</code>	\uparrow	<code>\Leftrightarrow</code>	\Leftrightarrow
<code>\leftarrow, \gets</code>	\leftarrow	<code>\downarrow</code>	\downarrow	<code>\Uparrow</code>	\Uparrow
<code>\longrightarrow</code>	\longrightarrow	<code>\updownarrow</code>	\updownarrow	<code>\Downarrow</code>	\Downarrow
<code>\longleftarrow</code>	\longleftarrow	<code>\upuparrows</code>	\upuparrows	<code>\Updownarrow</code>	\Updownarrow
<code>\leftrightarrows</code>	\leftrightarrows	<code>\downdownarrows</code>	\downdownarrows	<code>\rightharpoonup</code>	\rightharpoonup
<code>\longleftarrow</code>	\longleftrightarrow	<code>\nearrow</code>	\nearrow	<code>\rightharpoondown</code>	\rightharpoondown
<code>\mapsto</code>	\mapsto	<code>\searrow</code>	\searrow	<code>\leftharpoonup</code>	\leftharpoonup
<code>\longmapsto</code>	\longmapsto	<code>\nwarrow</code>	\nwarrow	<code>\leftharpoondown</code>	\leftharpoondown
<code>\hookrightarrow</code>	\hookrightarrow	<code>\swarrow</code>	\swarrow	<code>\rightleftharpoons</code>	\rightleftharpoons
<code>\hookleftarrow</code>	\hookleftarrow	<code>\Rightarrow</code>	\Rightarrow	<code>\leftrightharpoons</code>	\leftrightharpoons
<code>\rightleftarrows</code>	\rightleftarrows	<code>\Leftarrow</code>	\Leftarrow	<code>\upharpoonleft</code>	\upharpoonleft
<code>\leftrightarrows</code>	\Leftrightarrow	<code>\Longrightarrow</code>	\Longrightarrow	<code>\upharpoonright</code>	\upharpoonright
<code>\rightrightarrows</code>	\Rrightarrow	<code>\Longleftarrow</code>	\Leftrightarrow	<code>\downharpoonleft</code>	\downharpoonleft
<code>\leftleftarrows</code>	\Leftrightarrow	<code>\Leftrightarrow</code>	\Leftrightarrow	<code>\downharpoonright</code>	\downharpoonright

書体の変更

<code>ABC</code>	ABC	<code>\mathrm{ABC}</code>	ABC	<code>\mathcal{ABC}</code>	ABC
<code>\boldsymbol{ABC}</code>	\mathbf{ABC}	<code>\mathsf{ABC}</code>	ABC	<code>\mathfrak{ABC}</code>	\mathfrak{ABC}
<code>\mathbf{ABC}</code>	\mathbf{ABC}	<code>\mathtt{ABC}</code>	ABC	<code>\mathbb{ABC}</code>	\mathbb{ABC}

その他

<code>\frac{x}{y}</code>	$\frac{x}{y}$
<code>\sqrt{x}</code>	\sqrt{x}
<code>\sqrt[n]{x}</code>	$\sqrt[n]{x}$
<code>\binom{x}{y}</code>	$\binom{x}{y}$

※ `\frac{}{}` は、本文中の数式や別行立ての数式等、状況に応じて自動的に分数の大きさを調整してくれますが、`\dfrac{}{}` とすることで常に大きい分数表示、`\tfrac{}{}` とすることで常に小さい分数表示にすることもできます。また、連分数を書きたいときは `\cfrac{}{}` を使うとバランスよく表示されます。