

# PuyoTEX

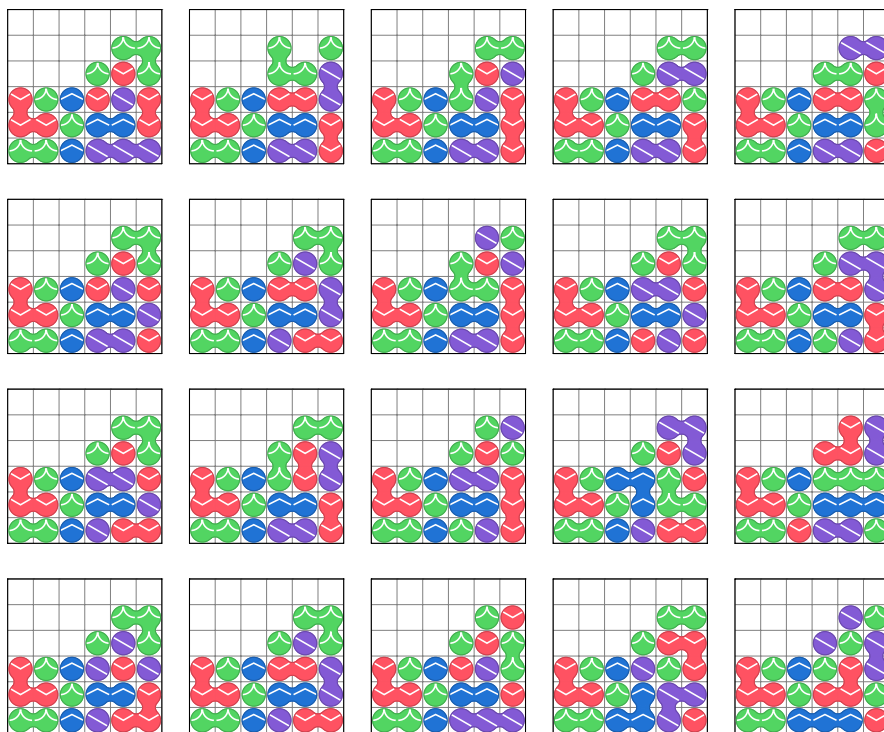
A  $\text{\LaTeX}$  package for quickly typesetting board states of Puyo Puyo games.

Supports large and small boards with arbitrary shape, hidden rows, current and next puyos, labels and move planning markers.

Source code available for download on [GitHub](#) or your favorite  $\text{\TeX}$  repository. Package requires [Python3](#) in support of scripts driven by [PythonTEX](#).

Move planning markers shown at left. Tailing variations on the Great Tanaka Rensa (GTR) shown below.

Created by [terramyst](#).



# 1 Installation

## 1.1 Python3 and PythonTeX

Installing this package should be straightforward through the user's T<sub>E</sub>X package manager. [Python3](#) will additionally need to be installed; it is recommended to use a virtual environment management tool such as [Anaconda](#). The Python environment will need packages [Pygments](#) and [Numpy](#) installed.

Compiling the document requires inserting a new command into the build sequence. [PythonT<sub>E</sub>X](#) must run `pythontex.py` in between two separate runs of the T<sub>E</sub>X engine (see documentation for more detail).

## 2 Usage

This package provides an environment and three macros for typesetting Puyo Puyo boards for inclusion in L<sup>A</sup>T<sub>E</sub>X documents.

### [puyotikz environment.](#)

The `puyotikz` environment is simply a wrapper for the `tikzpicture` environment. The user is therefore able to use additional [TikZ](#) macros inside this environment if they wish to augment whatever graphics are already provided by this package.

The `puyotikz` environment takes a single numeric argument which is passed to the `tikzpicture` environment as the scaling parameter. This package provides two scaling parameter definitions for convenience, `\puyosmallscale` and `\puyobigscale` (the default).

```
% example usage
\begin{puyotikz}[\puyosmallscale]
  % content...
\end{puyotikz}
```

For the traditional colors of red, yellow, green, blue, purple, and gray this package also provides new color definitions such as `bluepuyo` and `darkredpuyo`.

### puyoboard macro.

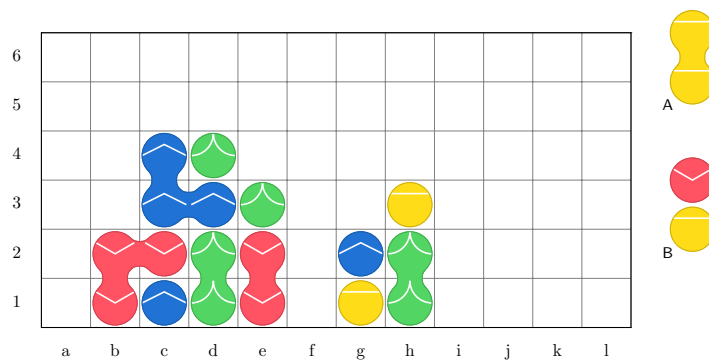
The `\puyoboard` macro takes 2 arguments and 4 optional parameters and must be used inside the `puyotikz` environment.

Puyos are specified in format strings by column (bottom-up, left-to-right). Each puyo is identified by a single letter (`r`, `y`, `g`, `b`, `p`, `n`). Columns (or next puyos) are separated by `/`. Columns are labeled numerically and rows and next puyos are labeled alphabetically (lowercase and uppercase, respectively).

```
% parameters
% - ncols: number of columns (default 6)
% - nrows: number of visible rows (default 12)
% - nhidrows: number of hidden rows (default 1)
% - showlabels: display coordinate labels (default True)

% arguments
% - #1: board format string
% - #2: next puyos format string

% example usage (see image below)
\begin{puyotikz}
  \puyoboard[ncols=12, nrows=6, nhidrows=0]
    {/rr/brbb/ggbg/rrg//yb/ggy}
    {yy/yr}
\end{puyotikz}
```



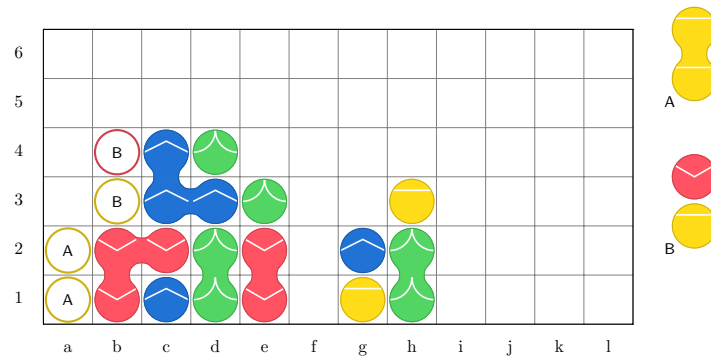
**puyomarker macro.**

The `\puyomarker` macro takes 1 argument and must be used inside the `puyotikz` environment following a `\puyoboard` macro.

Puyo markers are specified in a single format string, separated by /. A single marker is a concatenation of the row label (lowercase letter), column label (number), puyo color (lowercase letter) and puyo label (uppercase letter).

```
% arguments
%   - #1: marker format string

% example usage (see image below)
\begin{puyotikz}
  \puyoboard[ncols=12, nrow=6, nhidrows=0]
    {/rr/brbb/ggbg/rrg//yb/ggy}
    {yy/yr}
  \puyomarker{a1yA/a2yA/b3yB/b4rB}
\end{puyotikz}
```



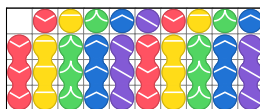
**puyogrid macro.**

The `\puyogrid` macro is provided as an example derived from `\puyoboard`. It assumes a `\puyosmallscale puyotikz` environment and only provides a single argument for the board format string (assuming no next puyos and no markers). It also assumes zero hidden rows and turns off labeling, however any parameters available to set in `\puyoboard` are also available in `\puyogrid`.

The user is encouraged to construct their own derived macros in similar fashion for their intended purpose.

```
% source code \puyogrid definition from puyotex.sty
\newcommand{\puyogrid}[2][]{
  \setkeys{puyoboard}{nhidrows=0, showlabels=False}
  \begin{puyotikz}[\puyosmallscale]
    \puyoboard[#1]{#2}{ }
  \end{puyotikz}
}

% example usage (see image below)
\puyogrid[nrows=4, ncols=10]
{rrr/yyr/gggy/bbbg/pppb/rrrp/yyr/gggy/bbbg/pppb}
```



## 3 Limitations

### 3.1 TikZ

The graphics provided in this package are created using [TikZ](#). Having many graphics compiled in a single document will result in a long compilation time.

It is recommended in this case to generate images in separately compiled standalone documents and include them in the main document via `\includegraphics`. [PythonTeX](#) is not compatible with `\tikzexternalize`.

### 3.2 PythonTeX

The [PythonTeX](#) documentation gives a warning which bears repeating: compiling a document that uses [PythonTeX](#) involves executing Python code on your computer therefore you should only compile documents from sources you trust.