

# テープ作成ツール

オートマトンで使うテープのpdf図を作成する。

## 必要環境

- python 3
- Microsoft製 python vscode拡張機能
- OSはなんでもいい

## texツール

- luatex
- tikz
- unicode-math
- amsfonts
- jlreq
- standalone

## 実行方法と生成ファイル確認方法

1. リポジトリをクローン. VSCodeで開き, releaseブランチに変更
2. ターミナルを開き, 以下のコマンドを入力.

```
python main.py
```

すると, **CFL=NPDA**ディレクトリが作られる. その内部構成は以下の通り.

```
CFL=NPDA  : 親フォルダ名
└─L(G)inL(M)  : 子フォルダ名
    ├─auxs
    ├─logs
    └─output  : オートマトンのテープ図がここに出力される !
```

4. **output**ディレクトリを開いて, **cfg\_to\_npdaIntegrate.pdf**を開く.

※**cfg\_to\_npdaIntegrate.pdf**の内容

プログラミング言語の文法は文脈自由文法 **cfg** によって規定されている.  
**cfg**は, 非決定性プッシュダウンオートマトン **npda** とが受理する記号列と,  
全く同じ記号列を生成する事が知られている.[1]

[1] 『オートマトン・言語理論の基礎』米田政明, 広瀬貞樹, 大里延康, 大川知 著, 近代科学社, (2003)

## main.pyの編集方法

main.pyで、`make_automaton_template()`関数の引数を入れる。

## テープ2本の場合

```
make_automaton_template("親フォルダ名","子フォルダ名 (いらなかつたら何も書かない)", [
    ["内部状態 :str","input tapeの内容文字列 :str","input headの位置 :int . 0始まり.", "storage tapeの内部文字列 :str",
     "storage headの位置 :int. 0始まり. "], # 画像1枚目
    ["内部状態 :str","input tapeの内容文字列 :str","input headの位置 :int . 0始まり.", "storage tapeの内部文字列 :str",
     "storage headの位置 :int. 0始まり. "], # 画像2枚目
    ["内部状態 :str","input tapeの内容文字列 :str","input headの位置 :int . 0始まり.", "storage tapeの内部文字列 :str",
     "storage headの位置 :int. 0始まり. "], # 画像3枚目

    # .. 4枚目以降も可能.
],
[
    "画像1枚目から2枚目に遷移する時, 矢印の下に書く註釈テキスト",
    "画像2枚目から3枚目に遷移する時, 矢印の下に書く註釈テキスト",
    # .. 3枚目以降も可能.
],
"テープは1本(fa)か2本 (pda)か :bool (True:1本 or False : 2本)"⇒False,
"画像のファイル名 :str",
"jlreq or standalone :str",
"画像を1枚ずつ作るかどうか :bool (True : 作る or False : 作らない)",
"画像を矢印でつなぎファイル作るかどうか :bool (True : 作る or False : 作らない),",
"画像を矢印でつなぎ, さらに見切れないよう以下の引数で調整したファイル作る時, 個別のファイルを作るかどうか :bool (True : 作る or False : 作らない)",
"画像を矢印でつなぎ, さらに見切れないよう以下の引数で調整したファイル作るかどうか :bool (True : 作る or False : 作らない),",
"standalone option. 上のboolがFalseだと効果なし :str",
"画像を何枚目で改行するか: , , , , ,... :intの可変長引数. 0始まり. !最後の画像の数を描くと0徐算エラーになるので注意! ")
```

## テープ1本の場合

```
make_automaton_template("親フォルダ名","子フォルダ名 (いらなかつたら何も書かない)", [
    ["内部状態 :str","storage tapeの内容文字列 :str","input headの位置 :int . 0始まり."], # 画像1枚目
    ["内部状態 :str","storage tapeの内容文字列 :str","input headの位置 :int . 0始まり."], # 画像2枚目
    ["内部状態 :str","storage tapeの内容文字列 :str","input headの位置 :int . 0始まり."], # 画像3枚目

    # .. 4枚目以降も可能.
],
[
    "画像1枚目から2枚目に遷移する時, 矢印の下に書く註釈テキスト",
    "画像2枚目から3枚目に遷移する時, 矢印の下に書く註釈テキスト",
    # .. 3枚目以降も可能.
```

```
[,
"テープは1本(fa)か2本 (pda)か :bool (True:1本 or False : 2本)"⇒True,
"画像のファイル名 :str",
"jlreq or standalone :str",
"画像を1枚ずつ作るかどうか :bool (True : 作る or False : 作らない)",
"画像を矢印でつないだファイル作るかどうか :bool (True : 作る or False : 作らない),,
"画像を矢印でつなぎ, さらに見切れないよう以下の引数で調整したファイル作る時, 個別のファイルを作るかどうか
:bool (True : 作る or False : 作らない)",
"画像を矢印でつなぎ, さらに見切れないよう以下の引数で調整したファイル作るかどうか :bool (True : 作る
or False : 作らない),,
"standalone option. 上のboolがFalseだと効果なし :str",
"画像を何枚目で改行するか: , , , , ,... :intの可変長引数. 0始まり. !最後の画像の数を描くと0徐
算エラーになるので注意! ")
```

## サンプルコード

### 1. 全部連結したファイル作る

```
make_automaton_template(
    "kSNAclosure",
    "concatenation",
    [
        ["q_{\triangleleft}", "\triangleright\cdots\sigma\triangleleft", 3, "\triangleright\cdots\gamma_3\cdots", 2],
        ["\textcolor{magenta}{p_{\triangleleft}}", "\triangleright\cdots\sigma\triangleleft", 3,
        "\triangleright\cdots\gamma_3\cdots", 2], #p_{\triangleleft}\in Q_{acc}
    ],
    [
        "\triangleleftで作業して受理状態に変化"
    ],
    False,
    "move30fM_1",
    "jlreq",
    False,
    False,
    False,
    True,
    "scale=0.5",
)
```

他の例⇒main.pyにある通り。

## 注意

tex記法で変更していくが、以下の点には注意。

- ＼から始まる文字やコマンドは\\から始める。
  - 例: ギリシャ文字のγである\gammaは\\gamma、σである\sigmaは\\sigmaと入力。
  - 例: 半角シャープ#は\\#と入力。
- uppertrackとlowertrackの表記は、[^{上側文字}\_{下側文字}]と書く。

。例::上に $\sigma$ 、下に $\gamma$ と書きたいときは[ $^{\{\backslash\backslash\sigma\}}_{\{\backslash\backslash\gamma\}}$ ]と書く。

- $\$\\\$aiueo$ は $\backslash\backslash aiueo$ に省略可能
- 添字倍加機能は添字部分に中括弧{}がないといけない

## エラーでた時の対処法

### index out of range

- オートマトンの画像情報は二次元配列になっている。1次元配列で書いていないか確認
- ksdaとkldaで二次元配列の項目が違うことに注意。
- 画像と画像を結ぶ矢印のテキスト書く所は一次元配列。かつ、必ず画像枚数分要素数が必要! 記入漏れはある?

### undefined control sequence

- 例えば $\backslash\backslash sigma$ とか、スラッシュの後の文字書いた後は半角空白が必要。 $\backslash\backslash simga$ 上とかだとエラーになる。 $\backslash\backslash sigma$ 上のと書く必要あり。

### ! Package graphics Error: Division by 0.

- テープ図の分割ミス

### Extra }, or forgotten \$.

- このエラーが出ている上のファイル名をクリックして、baselineが入っているtikzpicture環境を見ると、そこから下のコードの色合いが変になっている。適切な場所に $}$ を入れると、解決

### error: (file rm-lmr10) (type 3): font rm-lmr10 at 600 not found

→texliveが正しくインストールされてない可能性あり

- 参考リンク1
- 参考リンク2
- 参考リンク3

## 元ネタ

[pzuehlke/Turing-machine-tape-generator](#)のコードを一部改変したのが本作。