

# tlarray.sty (v0.1)

## 簡易マニュアル

ワトソン

2015 年 11 月 9 日

## 1 定義

TlArray パッケージにおける配列とは、複数のトークン列を扱うことのできるデータ構造をさす。各配列は半角の英数字により構成される配列名をもち、また、各配列の保持する個々のデータ（トークン列）を要素と呼ぶ。要素には順序が存在しており、1つの要素には1つのインデックスが対応している。ただし、先頭要素のインデックスを0とし、以降の要素はそれぞれ続く自然数と対応する。

ある配列に対して、その保持する要素の数を長さまたはサイズという。長さが0の配列を空の配列と呼ぶ。配列が保持するデータに応じた“型”のようなものは存在しないが、一部の命令はすべての要素が整数（整数を表す文字列またはカウンタレジスタ）の場合にのみ動作する。そのため、本マニュアルにおいては、便宜的にすべての要素が整数であるものを整数配列と呼ぶことにする。

## 2 配列の宣言

すべての配列は `\NewArray` で宣言してから用いる必要がある。

```
\NewArray{<配列名>}
```

このように `\NewArray` は `<配列名>` を引数にとり、当該の名称をもつ配列を宣言する。`<配列名>` には基本的な半角英数字のみが使用可能であり、その他はサポート対象外とする（実際には使用可能である場合もある）。

以降に登場するすべての命令は、この `\NewArray` で宣言済みの配列に対してのみ正常に動作する。

## 3 配列の表示

`\LengthofArray` を用いると、指定した配列の長さを取得することができる。なお、`\LengthofArray` には `\SizeofArray` という別名も存在する。

```
\LengthofArray{<配列名>}
```

```
\SizeofArray{<配列名>}
```

配列の特定の要素を取得するには `\GetArray` 命令を使用する。

```
\GetArray{<配列名>}{<インデックス>}
```

さらに `\ShowArray` 命令を用いれば、すべての要素をカンマ区切りで出力することも可能である。

```
\ShowArray{<配列名>}
```

## 4 要素の挿入と取り出し

### 4.1 Insert

`\InsertArray` 命令を用いると、任意の配列の任意のインデックスに要素を追加することができる。

```
\InsertArray{<配列名>}{<インデックス>}{<値>}
```

なお、この命令は指定箇所に破壊的な代入を行いますので使用時には十分注意されたい。すなわち指定した配列の指定インデックスに既に要素が存在していた場合、その値を上書きする。

また、インデックスに当該配列の長さ以上の値を指定した場合、間の要素には `\relax` が代入された状態となり、配列の長さは指定インデックスから 1 を減じた値に変更される。

※ `TlArray` パッケージでは `\relax` を一般的な言語における `null` に相当するものと捉えている。したがって、`\InsertArray` 命令の第三引数である `<値>` に `\relax` を指定することで、`\InsertArray` 命令を任意の要素の“削除”にも利用することが可能である。

### 4.2 Push と Pop

`\PushArray` 命令は、指定した配列の末尾に要素を追加する。

```
\PushArray{<配列名>}{<値>}
```

一方、`\PopArray` 命令は末尾の要素を取得（出力）し、その要素を配列から除去する。

```
\PopArray{<配列名>}
```

### 4.3 Shift と Unshift

`\ShiftArray` 命令は先頭の要素を取得（出力）し、その要素を配列から除去する。その後、配列に要素が残存していた場合にはすべての要素のインデックスが 1 つ前にずらされる。

```
\ShiftArray{<配列名>}
```

`\UnshiftArray` 命令は、指定した配列の先頭に要素を追加する。このとき、元々存在していた要素のインデックスは 1 つずつ後ろにずらされる。

```
\UnshiftArray{<配列名>}{<値>}
```

※これらの命令は内部的に重い処理を行うため（特に大きな配列に対しては）あまり乱用しない方がよい。

## 5 配列に対する操作

### 5.1 要素の交換

`\SwapArray` を用いると、特定の配列の 2 つの要素を入れ替えることができる。

```
\SwapArray{<配列名>}{<インデックス 1>}{<インデックス 2>}
```

さらに `\XSwapArray` を用いると、異なる配列間で要素を入れ替えることも可能である。

```
\XSwapArray{<配列名 1>}{<インデックス 1>}{<配列名 2>}{<インデックス 2>}
```

### 5.2 配列の反転

`\ReverseArray` 命令は、指定した配列の要素の順序を逆転させる。

```
\ReverseArray{<配列名>}
```

### 5.3 整数配列のソート

未実装。

## 6 配列と文字列の変換

`\StringtoArray` 命令で、与えた文字列を 1 文字ずつに分解して、指定した配列の末尾に付加させることが可能である。

```
\StringtoArray{<配列名>}{<文字列>}
```

逆に、`\toStringArray` 命令を用いると、指定した配列のもつ要素をすべてつなげた文字列を出力することができる。ただし、要素に制御綴が含まれる場合には、その展開および実行が行われる。

```
\toStringArray{<配列名>}
```

## 7 配列と制御綴

配列の要素に制御綴が含まれる場合、当該の制御綴は配列中では一切展開されず、そのままの状態では保持される。