

# コンピュータ概論 A/B

-- TeX (5) --  
(TeX マクロ)

数学科 栗野 俊一 (TA: 宮川 智行 [院生 2 年], 栗原 望 [院生 1 年])

2017/11/28 コンピュータ概

# 伝言

---

## 私語は慎むように !!

### □ 担任からの連絡

○ 学生証での出席は済ませましたか？

▶ 入口の脇の出席装置に学生証を翳す

### □ 席は自由です

○ できるだけ前に詰めよう

### □ 色々なお知らせについて

○ 栗野の Web Page に注意する事

<http://edu-gw2.math.cst.nihon-u.ac.jp/~kurino>

### □ VNC Server Address : 10.9.209.79

○ Password : vnc-2017

### □ ここ数週間の課題の評価が「未評価」になっています。

○ ごめんなさい、僕の単なるサボりです。気にしないでください。

# 前回(2017/11/21)の内容

---

## □ 前回(2017/11/21)の内容 : TeX による文章の作成

### ○ (pLa)TeX 形式で、文章を作成する

- ▶ 「%」から、行末迄は、コメント扱い(書いても無視される)
- ▶ 「{」と「}」(ブレース)は、「範囲指定」に利用される
- ▶ 「\」から始まる単語は「マクロ」とよばれ、特別な意味がある
- ▶ 三行の「おまじない」 : `\documentstyle{jsartcile}, \begin{document}, \end{document}`

### ○ 表紙 ( `\maketitle`, `\author`, `\date`, `\title` )

### ○ 数式 ( `$ ~ $` : 文中, `[ ~ ]` : 独立行, `\begin{eqnarray*} ~ \end{eqnarray*}` : 等式 )

- ▶ 数式処理中は数式モードという特別な状態になる (数式固有の表現が可能)

### ○ 文章構造 (章立て, 箇条書)

# 本日(2017/11/28)の予定

---

## □ 本日(2017/11/28)の予定

- 講義

- ▶ TeX マクロ

- 実習

- ▶ 実習 2: TeX でのマクロ利用

# 本日の課題 (2017/11/28)

---

## □ 前回 (2017/11/21) の課題

### ○ CST Portal に以下のファイルを提出しなさい

- ▶ ファイル名 : 20171121-QQQQ.tex (QQQQ は学生番号)
- ▶ 表題 : TeX による数学のレポート
- ▶ 内容 : TeX による数学のレポートを作成して提出
- ▶ 条件 : 名前と学生番号は自分のものにする
- ▶ 形式 : テキストファイル (sample-20171121.tex 参照)

## □ 今回 (2017/11/28) の課題

### ○ 次のファイルを提出しなさい

- ▶ 表題 : 自分の名前のロゴマクロ
- ▶ 内容 : 自分の名前のロゴマクロを定義したスタイルファイル
- ▶ ファイル名 : 20171128-QQQQ.sty (QQQQ は学生番号)
- ▶ 詳しくは、配布した sample-20171128.sty の内容を参照

# マクロ(Macro)

---

## □ Macro(マクロ) とは

○ Micro(マイクロ) の反対、では Micro は.. ? Macro の反対.. ;-D

▶ というの循環定義(論法)という.. (駄目な定義の典型例..)

○ 再度 Macro の意味 : 「大きな」という接頭語

▶ Macro : 大きい <-> Micro : 小さい

○ でも、何を基準に「大小」を考える .. ?

▶ 最小のモノ(基本命令/基本操作/基本関数)は Micro(マイクロ)

▶ それを「いくつかまとめたもの」→ Micro でない→ Macro

○ 計算機における「Macro」の意味

▶ 基本的なモノ(何が「基本」か ? は、システムによって異なる)を、いくつかまとめたモノ

▶ Macro 定義 : Macro に名前を付ける(事によって名前で Macro が参照できるようにする)事

# 実習 1 pLaTeX の利用 (復習)

---

## □[実習 1] pLaTeX の利用

- tex ファイルの作成 (tex-000.tex) : Text エディタ(サクラエディタ)を使う
  - ▶ 拡張子は「tex」にする
  - ▶ sample file をダウンロードしてもよい
- wincp によるファイル転送 : wincp で tex ファイルを ubuntu にコピーする
- platex による typeset
  - ▶ ubuntu を利用して「platex tex-000.tex」とする
  - ▶ 問題がなければ、「tex-000.dvi」(dvi 形式ファイル)が作られる
- dvi2pdf による pdf 形式への変換
  - ▶ ubuntu を利用して「dvi2pdf tex-000.dvi」とする
  - ▶ 問題がなければ、「tex-000.pdf」(pdf 形式ファイル)が作られる
- pdf ファイルの表示
  - ▶ ubuntu を利用して「evince tex-000.pdf」とする
- wincp によるファイル転送 : wincp で pdf ファイルを windows にコピーする
  - ▶ windows で tex-000.pdf を「開いて」もよい

# TeX のマクロ機能

---

## □ TeX のマクロ機能 (tex-001.tex)

### ○ マクロ機能

▶ TeX では、「『文字列』に『名前』を付けて、参照する」事ができる

### ○ マクロ名：文字列に付けられた名前

▶ 「名前」は習慣により「\`\`(バックスラッシュ)」で始まる物にする

### ○ マクロ定義：『文字列』に『名前』を付する事

▶ マクロ定義の表現：`\newcommand{新しい名前}{文字列}`

### ○ 例：`\newcommand{\MyJapaneseName}{栗野俊一}`

▶ 文字列「栗野俊一」に、マクロ名「`\MyJapaneseName`」をつけた

▶ マクロ「`\MyJapaneseName`」を、定義した(マクロ定義した)

▶ 文字列「栗野俊一」は、マクロ「`\MyJapaneseName`」の定義内容

### ○ マクロ展開(参照)

▶ マクロ定義済のマクロ名を記述すると「その定義内容」に置き換わる

▶ 例：「私の名前は`\MyJapaneseName` です」→「私の名前は栗野俊一です」

# Style ファイルとマクロの分離

---

- マクロは何度も「使い回す(再利用する)」事が多い
  - 毎回同じ事を書くのは面倒 / 一度作成した内容を使い回す
    - ▶ マクロ定義だけを記述した Style ファイルを作成して読み込む (`\input`)
- マクロ集としての **style** ファイル
  - TeX の特徴の一つ : 「内容」と「形式」が分離できる (内容だけに専念できる)
    - ▶ 「内容」は `.tex` ファイルに記述。では、「形式」は .. ? 実は **style** ファイルに記述されている。
  - 色々な **style** ファイル
    - ▶ 中に、色々なマクロが定義されている
    - ▶ 内容で、同じ名前のマクロを利用しても、マクロの内容が異れば、異なる結果になる
    - ▶ 「形式」の情報を「マクロ」にしておくことにより、形式と内容を分離できる
  - `\usepackage` マクロ : 実は、**style** ファイルを読み込んでいる

# pLaTeX vs TeX

---

## □ pLaTeX vs TeX

○ 今迄、pLaTeX は TeX の一種と言っていたが.. ?

▶ 実は、pLaTeX は TeX そのもの / 単に、pLaTeX 専用のマクロが事前に定義されて(読み込まれている..)だけ..

○ じゃあ、なぜ別の名前 ?

▶ マクロの量が多く、また、利用形式が独特なので、まるで違ったように見える

▶ 違う物は違う名前の方が混乱が少ない(cf. 役者はドーランを塗ったら別人)

# 色々な TeX マクロ

---

## □ 表示位置を変更するマクロ

### ○ 文字の位置を上にする

▶ `\raise<サイズ>\hbox{<文字列>}`

▶ 例: `\raise.4ex\hbox{野}`: 「野」の表示位置を x の高さ(ex)の 0.4 倍だけ上に

### ○ 文字の位置を下にする

▶ `\lower<サイズ>\hbox{<文字列>}`

▶ 例: `\lower.2ex\hbox{俊}`: 「俊」の表示位置を x の高さ(ex)の 0.2 倍だけ下に

### ○ 文字と文字の間隔を調整する

▶ `\kern<サイズ>`

▶ 例: `俊\kern.1em一`: 「俊」と「一」の間を m の幅(em)の 0.1 倍だけ広げる

▶ サイズに負の数を指定すると幅を狭める事もできる

# 実習 2: TeX マクロ

---

## □[実習 2.1] マクロを利用する

- `tex-001.tex` の `\MyJapaneseName` の定義を変更して試してみる

## □[実習 2.2] style ファイルの作成

- `tex-001.tex` の `\MyJapaneseName` の定義を切り取り、別のファイル(`myinfo.sty`) に保存

- `tex-001.tex` のプリアンブルに `\input{myinfo.sty}` を追加

▶ `tex-002.tex` を DL してもよい

## □[実習 2.3] 名前ロゴの作成

- `tex-003.tex` を参考にロゴを作るマクロを作成する

## □[実習 2.4] 引数付きマクロの作成

- `tex-004.tex` を参考に引数付きマクロを作ってみる

## □[実習 2.5] 課題の作成

- 自分ロゴのマクロを作る

▶ `sample-20171128.tex`, `sample-20171128.sty` を参考に `sample-20171128.sty` を作る

▶ `sample-20171128.tex` を `typeset` して、きちんと、pdf が自分の情報になっている事を確認

# 引数付きマクロの定義

---

## □ 引数付きマクロ

- マクロの内容を一部後から指定したい場合がある

- ▶ ほとんど同じだが、一部だけ異なる文字列を何度も利用したい場合

- ▶ 引数付きマクロによって対応可能

## □ 引数付きマクロの定義(三つのポイント)

- 引数付きマクロの場合は、引数の個数を「[n] (n は個数)」で指定する

- ▶ 例 : `\newcommand{\ThreeArgMacro}[3]{ .. }` : 引数は三つある

- 内容の中に「#1 ~ #n」を含める

- ▶ 例 : 上記で { 最初は #1, 次は #2, 最後が #3 } とする

- マクロを利用する時にマクロ名の後ろに n 個の引数を指定する

- ▶ 例1 : `\ThreeArgMacro{芋虫}{蛹}{蝶}`

- ▶ 例2 : `\ThreeArgMacro{おたまじゃくし}{足が生えてきて}{蛙}`

- ▶ 例3 : `\ThreeArgMacro{ちよろちよろ}{ぱっぱ}{子供が泣いても蓋とるな}`

# 引数付きマクロの作り方

---

## □ 何時、引数付きマクロを作るか (When) ?

- 似ているマクロが出て来た時 ( cf. 連分数を沢山定義してみる .. )

  - ▶ # 「『今』でしょう」とは.. 言わない... :-)

- 初めっから引数付きマクロを作る必要はない

  - ▶ 初めっから複数の似ているマクロを作る場合はある..(その時には最初から..)

## □ どうやって、引数付きマクロを作るか (How to) ?

- 基本は、似ているマクロの統合(抽象化)

  - ▶ 似ているマクロの形を揃えて、「同じ所」と「違う所」が解るようにする

  - ▶ 「同じ所」はそのままで、「違う所」を引数参照 (#n) に変更

  - ▶ 新しい(引数付き)マクロを定義し、それに引数の個数(#n の個数)を追加

  - ▶ 「違う所」はマクロ参照の引数(マクロの後ろに{}で記述..)で指定する

# 実習 3: TeX の引数付きマクロ

---

□ [実習 3] 引数付きマクロを作成してみる

○ sample-20171128.tex の中に色々な連分数を定義して表示してみる

▶ 参考 : <http://d.hatena.ne.jp/rikunora/20090830>