

コンピュータ概論 A/B

-- TeX Macro --

数学科 栗野 俊一 (TA: 浜津 翔 [院生 1 年])

2013/11/05 コンピュータ概

伝言

私語は慎むように !!

- 席は自由です : できるだけ前に詰めよう
- すぐやること
 - Note-PC の電源ケーブルを継ぎ、電源を入れ、今週の資料を読む
- 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>
- 【お詫び】先週(2013/11/05)は、OHP 資料の作成が間に合いませんでした(すみません)
- 【注意】
 - 講義の速度が早過ぎたら
 - ▶ 「遅くするように / もう一度説明するように」と申し出る
 - 今聞いた内容を、友達に聞きたければ
 - ▶ 「友達に確認したいので、時間が欲しい」と申し出る(勝手に話をしない)
 - 来週(11/11)は補習を予定しています
 - ▶ 教室は 1214 GW 教室になります

前回(2013/10/29)の復習

□ 講義

- システム : (興味)の「対象」と対象の「操作」からなる (cf. 親と子の関係)
 - ▶ 「操作」を学ぶ事により「対象」を望みの形にする事が可能になる
- メタシステム : 「操作」を「操作の対象」にするシステム (cf. 祖父母と親の関係)
 - ▶ 「操作」を操作する事により、「間接的」に「対象」を操作する
 - ▶ 「間接的」なので「難しい」が、「とても強力」な仕組
- 計算機に於けるメタシステム : 対象はデータで、操作はソフト
 - ▶ プログラミング : ソフトを「作る」作業 (メタシステム)

□ 実習

- Mathematica で「数」を作る
 - ▶ ペアノの公理に基づく「自然数」と「四則」の実現

本日(2013/11/05)の予定

□ 講義

- 拡張子
- TeX のマクロ

□ 実習

- [演習 1] TeX の復習
- [演習 2] TeX のマクロの作成
- [演習 3] 課題の作成

本日の課題 (2013/11/05)

□ 前回 (2013/10/29) の課題

○ 前は課題がありませんでした

□ 今回 (2013/11/05) 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

▶ ファイル名 : 20131105-QQQQ.sty (QQQQ は学生番号)

▶ 詳しくは、配布した sample-20131105.sty の内容を参照

ファイルシステム

□ ファイル

○ 計算機の外部記憶装置(通常はハードディスク)には、沢山の情報が記録できる

▶ ファイルは、それに記録されている「まとまりのある情報の塊」の事

▶ # 外部記憶装置の記憶領域が細かく分割されており、その区画の一部にファイルの中身が記録されている

○ 複数のファイルの「容れ物」として「フォルダ」がある

▶ 「ファイルシステム」はファイルを管理するための仕組みで木構造になっている

○ ファイル名

▶ 個々のファイル/フォルダに付けられた名前

□ ファイルシステム(Name/ID)とパス名(Address)

○ パス名：ルートフォルダから、そのファイルの位置までの経路(パス)を表現した名前

▶ ファイル名はフォルダ内でのファイルの区別に利用される / 異なるフォルダにある同じファイル名は異なるファイルを示している(ファイルの位置を示す)

▶ パス名は一意性をもつ：同じパス名を持つファイルは一つの計算機内(ファイルシステム内)に一つしかない

ファイル名と拡張子

□ ファイル名

- ファイルにつけられた名前:ファイルの内容を区別するために付ける

- ▶ 基本は「ベース名」と「拡張子」(1 ~ 3 文字)からなり「.」で区切られる

- 拡張子の扱い

- ▶ ファイル名中に「.」が二つ以上含まれる場合は最後の「.」から後ろが「拡張子」

- ▶ 通常ファイルはファイルの中身を表す拡張子を付ける*習慣*がある / フォルダには、通常、拡張子は付けない

- ▶ ※ ファイル名を二つに分けるのは単なる「名付け規則」ではあるのだが、歴史的な意味があり、特別な扱いがされる

□ 拡張子と関連付け

- 拡張子は、「ファイル形式」を表す物をつける習慣がある(メタ情報)

- 「ファイル形式」は、「その形式を扱う(事ができる)ソフト」と関連付けが可能

- ▶ 「ダブルクリックで開く」のは、「関連付け」がされているため

- ▶ 「拡張子」を変更するとアイコンが変化するのも「関連付け」のため

実習 1: 拡張子

□[実習 1.1] 拡張子の表示

○フォルダオプションを変更して、拡張子が表示されるようにする

▶[コンピュータ]→[コントロールパネル]→[デスクトップのカスタマイズ]→

▶[フォルダオプション]→[表示タブ]→[登録されている拡張子は表示しない]

□[実習 1.2] 拡張子の変更

○拡張子を変更する

▶アイコンが変化したり、「開く」ソフトが変化する事を確認する

□[実習 1.3] 関連付けの変更

○関連付けを変更する

▶アイコンが変化したり、「開く」ソフトが変化する事を確認する

TeX の復習(1)

□ pLaTeX システムとは(What)

- tex 形式の文章ファイル(*.tex)を typeset(整形)して dvi 形式ファイルを作るソフト
 - ▶ dvi 形式のファイルは更に dvipdfmx で pdf 形式に変換できる
 - ▶ 一度、pdf 形式にすれば、表示も印刷も綺麗にできる

□ TeX を利用する理由(Why)

- 印刷物(「内容」と「形式」の二つの情報からなる)の作成を「内容」だけに専念したい
 - ▶ 「形式」は、TeX が対処してくれる(形式は style ファイルの形で別に記述されているので、その御仕着せに頼る)
- tex 形式は基本 Text 形式である
 - ▶ 「編集」を「Text Editor(サクラ)」で行いたい / 高速にかつ汎用で、しかも、手慣れている
 - ▶ pLaTeX がなくても、ある程度、中身が理解できる (e-mail での利用)
- 色々なシステムで動く
 - ▶ MS-Windows は勿論、Linux (栗野愛用)や、MacOS でも利用可能
- 数式を使いたい
 - ▶ 数式を扱いたければ、(残念な事に、未だに..) 一番便利

実習 2: pLaTeX の利用 (復習)

□[実習 2] 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

○ platex による typeset

- ▶ コマンドプロンプトを利用して「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

- ▶ コマンドプロンプトを利用して「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

- ▶ コマンドプロンプトを利用して「tex-000.pdf」とする
- ▶ 普通に、tex-000.pdf

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：文字列につけられた名前

▶ 「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
 - 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 の一種といていたが.. ?

▶ 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すくない(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

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}`

メタ, 関数定義, マクロ定義

□ 計算機操作におけるメタ

○ ファイル(対象), ソフト(操作)

▶ メタは?: インストール / プログラム作成

□ プログラミング

○ 計算機操作におけるメタの典型

▶ Mathematica の関数 / TeX のマクロ

□ コンピュータ概論の*裏*目標

○ 「計算機に使われる」のではなく「計算機が使える」人材の育成

▶ ソフトを使うだけの人 (実は、ソフトに使われている人: ×)

○ プログラミングができる人

▶ ソフトを作る人: つまり、「計算機が使える」人

○ 講義では「概念」だけを述べている

▶ 自分で色々試してみよう (結局、何時ものセリフ)

実習 3: TeX マクロ

□[実習 3.1] 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 3.2] 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 3.3] 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 3.4] 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 3.5] 課題の作成

- 自分ロゴのマクロを作る

▷ `sample-20131105.tex`, `sample-20131105.sty` を参考に `sample-20131105.sty` を作る

▷ `sample-20131105.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