

# コンピュータ概論 A/B

-- TeX & Mathematica --

数学科 栗野 俊一 (TA: 浜津 翔 [院生 1 年])

2013/06/18 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>

**TeX** のインストールをしておきましょう

□ VNC の IP Address : 10.9.209.107

# 前回(2013/06/11)の内容

---

## □ [注意]

- TeX のインストールが終っていない人がいます

- ▶ すんでいない人は、昼休みか 5 限に TA に申し出てください

- ▶ TeX のインストールが動かないと前期の試験の半分は答えられません

- ▶ 講義中に、「TeX のインストール」の質問はしないでください(他人の迷惑)

## □ 前回(2013/06/11)の内容

- TeX による文章の作成

## □ 講義内容

- TeX による文章の作成手順を説明

# 本日(2013/06/18)の予定

---

## □ 本日(2013/06/18)の予定

- 前回、ほとんど進まなかったなので、今回、もう一度やる
- pLaTeX での文章の作成
- 式の扱い

## □ 本日(2013/06/18)の目標

- pLaTeX の利用方法を学ぶ

## □ 演習

- [演習 1] pLaTeX による文章の作成
- [演習 2] ファイルに分割と Mathematica の結果の利用
- [演習 3] pLaTeX で色々な数式を記述する

# 本日の課題 (2013/06/18)

---

## □ 前回 (2013/06/11) の課題

- 前回の課題提出はなし

## □ 今回 (2013/06/18) の課題

- 前回の課題を行う/ファイル名なども前回(20130611)を利用する
- ファイル名 : 20130611-QQQQ.tex (QQQQ は学生番号)
- 表題 : TeX で Mathematica の図を利用する
- 内容 : TeX で Mathematica で作成した図を利用する
- 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-20130611.tex 参照)

# VNC

---

## □ VNC とは

### ○ Virtual Network Computing の略

- ▶ ヴァーチャル・ネットワーク・コンピューティング
- ▶ ネットワーク経由で、他人の画面が見れる/操作できる !!

### ○ 構成

- ▶ vncserver : ネットワークからその PC が操作できるようにする(危険)
- ▶ vncviewer : ネットワーク上の他の PC が操作/観察できるようにする(便利)

## □ VNC のインストール

### ○ tightvnc-2.7.7-setup-64bit.msi をダウンロード

- ▶ 普通に「開け(ダブルクリックすれ)」ばインストールが始まる
- ▶ 自分の PC の Server のパスワードは自分で決める
- ▶ このパスワードがバレると他人から、自分の PC が操作されてしまう

## □ vncviewer の利用

### ○ コンピュータの操作内容を手元で見たい場合は vncviewer を開く

- ▶ IP Address と Port 番号は、毎回黒板に提示する
- ▶ パスワード : comp2013

# 木 (Tree)

---

## □ 構造

- いくつかのノード(点/要素)をアーク(線/関係)で結んだもの

## □ 木(き)/木構造(もくこうぞう)

- 階層を表す構造の一つで、次の様に再帰的に定義される

- ▶ 単独のノードは木である(この木の根は、この単独のノード自身である)
- ▶ 新しいノードと複数の木の根を結んだものは木である(根は新しいノード)
- ▶ 上記の二つの規則で作られたものだけが木である

- cf. 「自然数の定義」(ペアノの公理の一部)

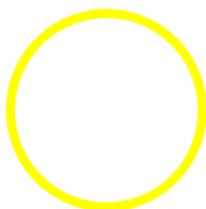
- ▶ 1 は自然数である
- ▶  $n$  が自然数ならば、 $n+1$  も自然数である
- ▶ 上記の二つの規則で作られたものだけが自然数である

- 木に関連する用語

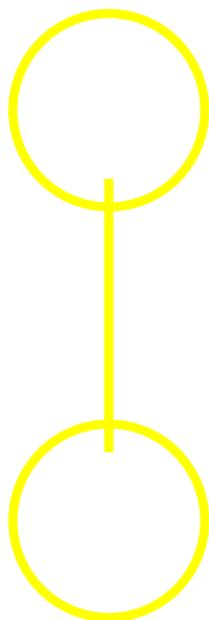
- ▶ 祖先/子孫：根は他の要素の先祖になる、その逆の関係が子孫
- ▶ 親/子：自分と直接接続している祖先は親、その逆の関係が子
- ▶ 上/下：親が上、子が下 ( 計算機の「木」は根を上を書く )
- ▶ 根/枝/葉：親を持たないノードが根、子を持たないノードが葉、その他が枝

# 木の例

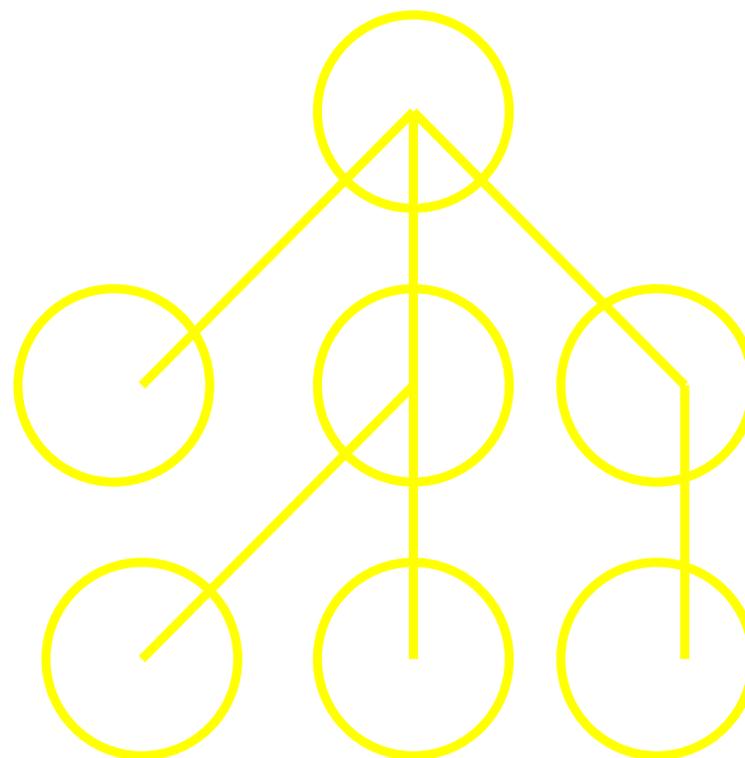
---



(a)



(b)

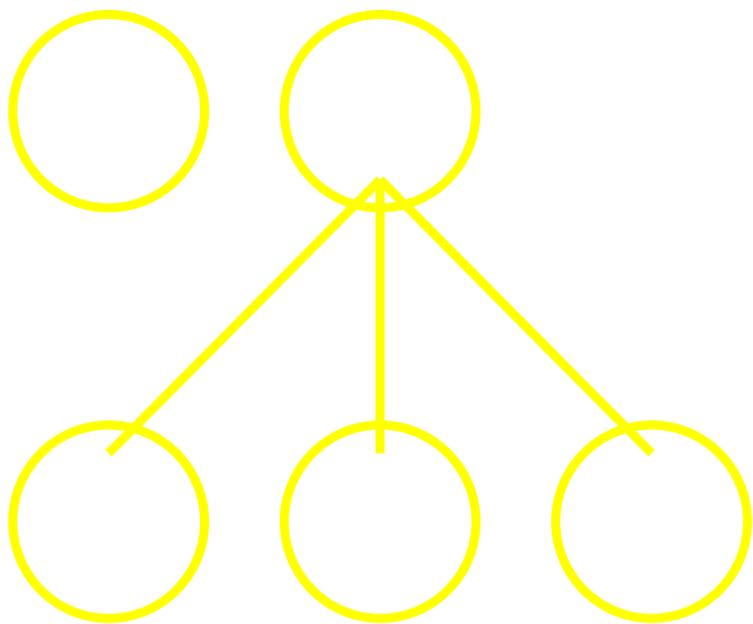


(c)

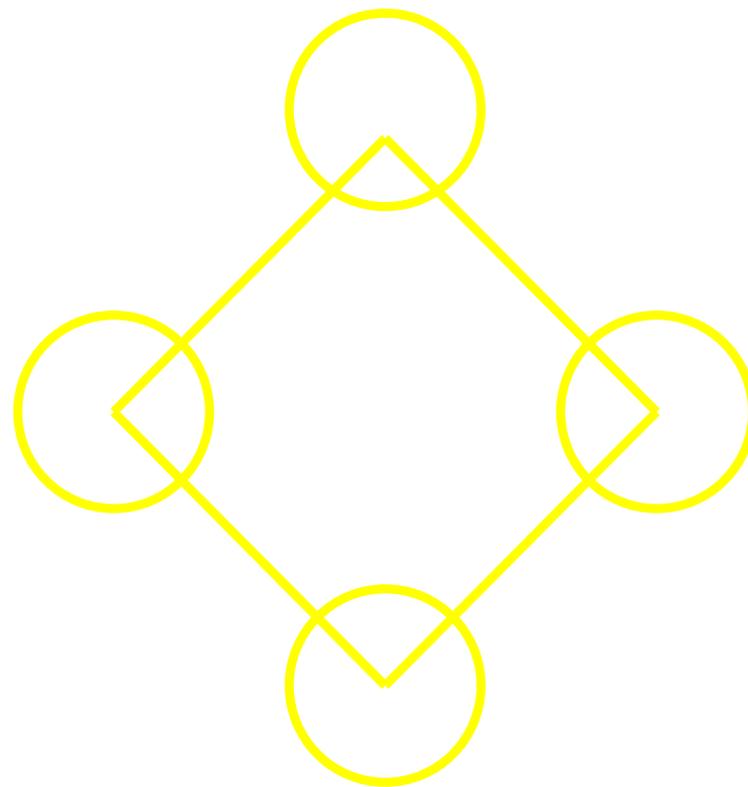
- (a) : 根が一つだけの木
- (b) : 根が一つ、葉が一つで、枝のない木
- (c) : 葉が四つ、枝が三の木

# 木でない例

---



(A)



(B)

- (A) : 繋がってない部分がある(森の例)
- (B) : 輪ができています(木には、輪ができない)

# ファイルシステム

---

## □ ファイルシステム

- 外部記憶上のデータの管理方法の事

  - ▶ ファイルをどのように指定するか的方式

- Windows 7 のファイルシステムは NTFS

- 情報の単位はファイル

## □ NTFS は階層型のファイル管理を行っている

- 外部記憶装置の一番大きな単位は、ドライブ(外部記憶装置)

  - ▶ SSD, HD, DVD-Drive, USB Memory, etc ..

- ドライブ内は木構造

  - ▶ ドライブは一文字 (ドライブ・レター) で表す

  - ▶ ハードディスクは C となっている (が、決まっているわけではない)

- 要素(根以外は名前をもつ)は二種類ある

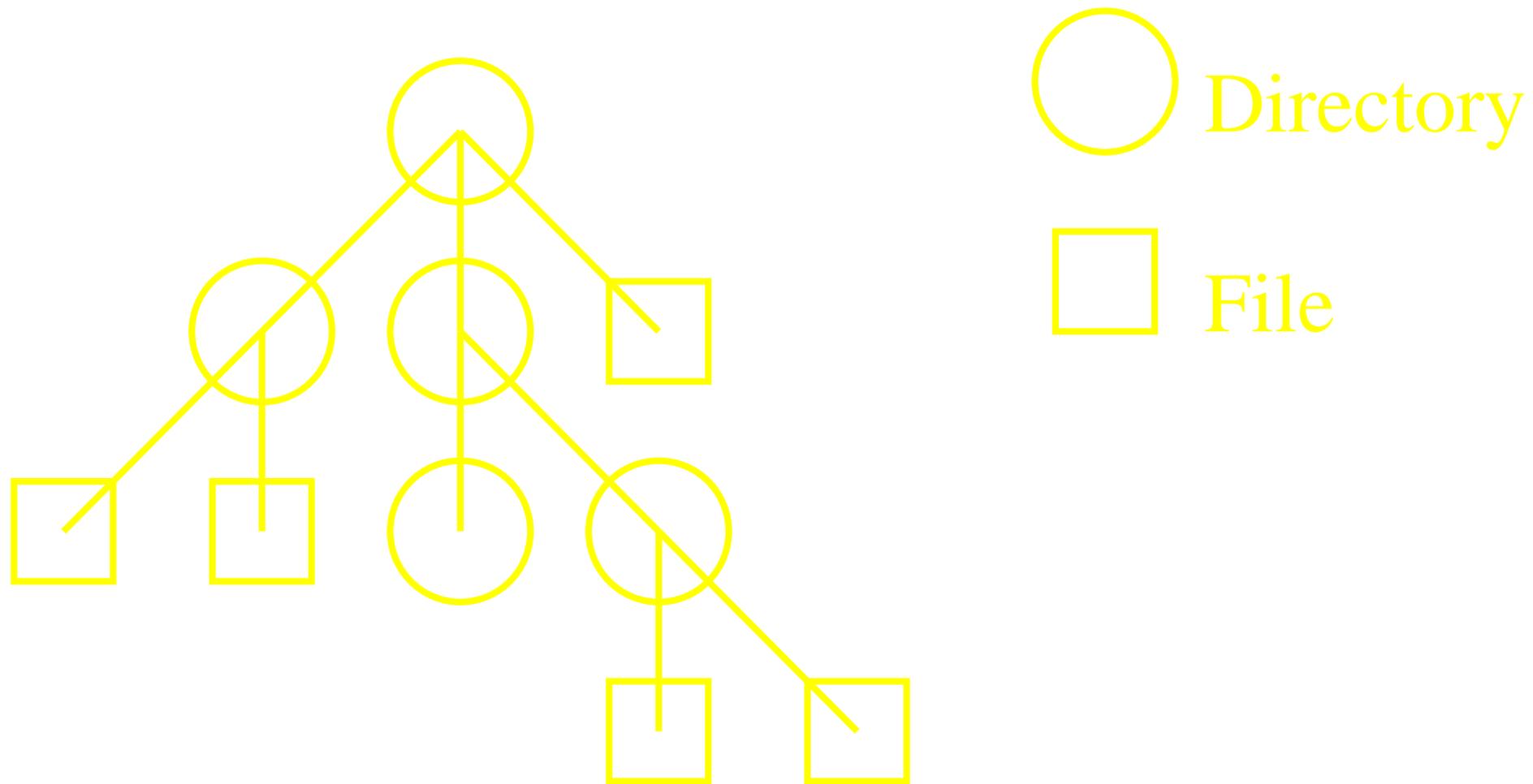
  - ▶ ディレクトリ(フォルダ) : 子を持つことができる

  - ▶ ファイル : 子は持てない(必ず葉になる)

- 根の事を「ルート(root)」と呼ぶ事が多い

# ファイルシステムの例

---



- ノードはディレクトリ(○)とファイル(□)の二種類
  - ファイルは葉にしかない (子ができない)
    - ▶ 空のディレクトリも作れ、葉になる。
  - 個々のノードには名前がついている

# パス(path)名

---

## □ パス(path)名

### ○ ファイルを指定する名前

- ▶ 絶対パス : どこにいても同じものを指すが長い
- ▶ 相対パス : 居る場所によって異なる物になるが短くできる

## □ 絶対パス(path)名

### ○ ディレクトリ/ファイルを一意に示す識別子

- ▶ 根からディレクトリ名を「¥」で継げたもの (かならず「¥」で始まる)
- ▶ 根からそのノードまでの経路(path)を示す
- ▶ 根自身のパス名は「¥」

### ○ 絶対パス名の例

- ▶ サンプル TeX ファイル: 「C:¥usr¥tex¥20130618¥sample-20130618.tex」

# 相対パス名

---

## □ カレントディレクトリとカレントドライブ

- 現在の注目位置を表現する(パスで表現される)

  - ▶ コマンドプロンプトで表示される文字列

  - ▶ アドレスバーに表示される文字列

- カレントディレクトリは、ドライブ毎に存在する

  - ▶ cd コマンドで変更(移動)可能 ( cd 移動先のフォルダのパス名)

## □ 相対パス

- カレントディレクトリからみた経路

  - ▶ かならず「¥」以外で始まる

- 基本：カレントディレクトリ + ¥ + 相対パス = 絶対パス

  - ▶ カレントディレクトリが異れば同じ相対パスでも異なるファイルを指す

- 応用

  - ▶ 「.」はカレントディレクトリを表す

  - ▶ 「..」は親ディレクトリを表す(根の親は根になっている)

  - ▶ 相対パスを利用して、従兄弟ファイルを指定することもできる

# パス名の例

---

## □ 状況 : TeX の利用中

### ○ 例 1 (子/子孫)

- ▶ カレントディレクトリ: C:¥usr¥tex
- ▶ 絶対パス: C:¥usr¥tex¥20130618¥sample-20130618.tex
- ▶ 相対パス: 20130618¥sample-20130618.tex

### ○ 例 2 (親)

- ▶ カレントディレクトリ: C:¥usr¥tex¥20130611
- ▶ 絶対パス: C:¥usr¥tex¥hello.tex
- ▶ 相対パス: ..¥hello.tex

### ○ 例 3 (兄弟/従兄弟)

- ▶ カレントディレクトリ: C:¥usr¥tex¥20130618
- ▶ 絶対パス: C:¥usr¥tex¥20130611¥sample-20130611.tex
- ▶ 相対パス: ..¥20130611¥sample-20130611.tex

## □ パス名の利用

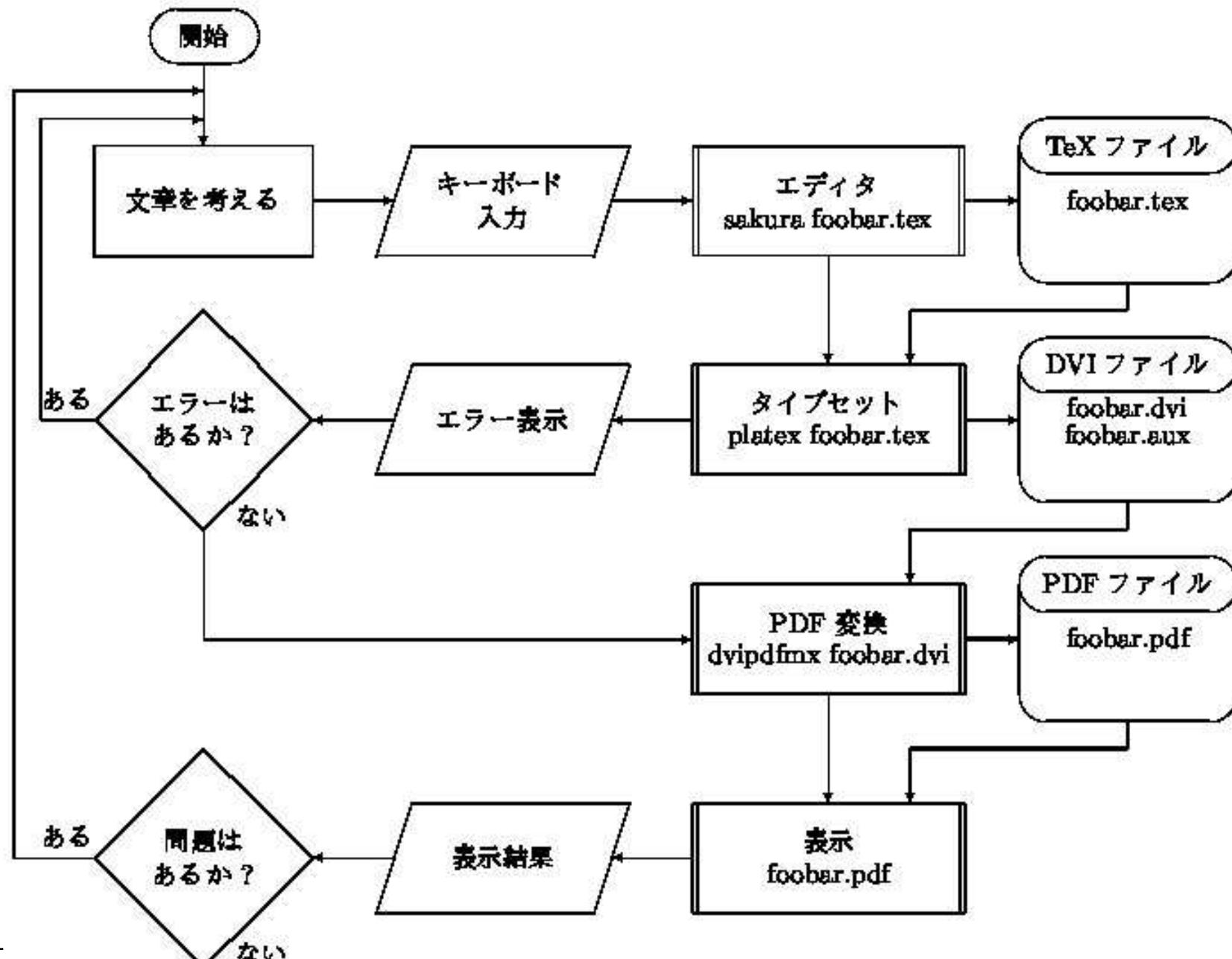
- 異なるファイルはパス名が異なるのでファイル名が同一でも区別できる
- パス名はファイルの位置も表現しているため、そこに行ける

# TeX を利用した文章ファイルの作成手順

---

- テキストエディタで QQQQ.tex ファイルを作成する
  - サクラエディタで、QQQQ.tex ファイルを新規作成し、文章を入力する
- タイプセット
  - latex コマンドで QQQQ.tex をタイプセットする
    - ▶ 「latex QQQQ.tex」を実行
    - ▶ QQQQ.dvi が作成されていれば OK
- pdf ファイルの作成
  - dvi2pdf コマンドで QQQQ.dvi から QQQQ.pdf を作成する
    - ▶ 「dvi2pdf QQQQ.dvi」を実行
    - ▶ QQQQ.pdf が作成されていれば OK
- 最終結果の確認
  - QQQQ.pdf をアクロバットリーダーで表示して内容を確認
    - ▶ 「QQQQ.pdf」を実行
    - ▶ QQQQ.pdf の内容が表示され、問題なければ OK

# TeX 文章の作成フロー



# TeX 文章 (1) : Hello World

---

## □ TeX 文章の内容 : 二種類の情報が含まれる

### ○ 内容 : 文章の内容そのもの ( 普通に日本語で入力する )

- ▶ 日本語の文章は適切に整形されるので、体裁を気にする必要はない
- ▶ 「体裁をどうするか」も細かく指定できるが、最初は「御仕着せ」で十分

### ○ 構成情報 : その部分の文章内での位置付け ( 「\」で始まる「マクロ」で指定 )

- ▶ 文章に関する情報を記述しており、「体裁」に影響する

## □ pLaTeX で「Hello, World」

### ○ 「Hello, World」という内容(?)の TeX 文章を作成してみる (tex-000.tex)

- ▶ pLaTeX では、取り敢えず、三行の「お呪い」を入れる

```
\documentclass[a4paper]{jsarticle}
\begin{document}
\end{document}
```

- ▶ 「文章の内容」は「\begin{document}」と「\end{document}」の間に入れる

### ○ tex-001.tex の内容

```
\documentclass[a4paper]{jsarticle}
\begin{document}
Hello, World
\end{document}
```

# typeset (1) : Hello World

---

- **tex** ファイルから pdf ファイルを作成する
  - pdf ファイル : とりあえず、表示や印刷に都合が良い形と思え
    - ▶ 「pdf ファイルとは何か？」 → ググれ
- **tex** ファイルを作成する
  - 「サクラエディタ」を開く
    - ▶ [スタート] → [すべてのプログラム] → [サクラエディタ] → [サクラエディタ]
  - 内容を入力
    - ▶ キーボードから 4 行の内容を入力
  - **tex** ファイルへの内容の保存
    - ▶ [ファイル(F)] → [名前を付けて保存(A)] → [c://usr//tex//20130618//tex-000.tex] に保存
- **pdf** ファイルの作成手順
  - 「コマンドプロンプト」を開く
    - ▶ [スタート] → [すべてのプログラム] → [アクセサリ] → [コマンドプロンプト]
  - **tex** ファイルのある場所へ移動「`cd \usr\tex\20130618`」
  - **platex** でタイプセット「`platex tex-000.tex`」
  - **dvipdfmx** で pdf ファイルを作成「`dvipdfmx tex-000.dvi`」

# TeX 文章 (2) : 日本語の文章

---

## □ TeX での日本語の文章の扱い (tex-001.tex)

- 基本は、入力された内容がそのまま pdf に反映される

- ▶ 文章は、改行が削除され、一行に追い込まれ、右揃えされる

- 空行(改行を二つ連続させる)をいれると、段落の区切とみなされる

- ▶ 段落の先頭には、空白が一文字自動的に挿入される

- 文章作成の段階では、「最終的な結果」を意識しなくてよい

- ▶ 「文章の内容」と「論理構造(どの部分がまとまっているか)」だけを意識する

- ▶ 「文章の体裁」を考えるのは「TeX の仕事」と割り切ってよい

- ▶ tex ファイルは、「文章作成(操作)」に「都合よく \*して\* よい」

## □ 「体裁」は、「まとめて」、「統一的」に変更可能

- twocolumn を追加するだけで、二段組にできる (tex-002.tex)

# TeX 文章 (3) : 表紙、章立て、目次

---

## □ 表紙

- 文章には、文章の内容とは別に表紙をつける事ができる
  - ▶ 表紙を作るには、`\maketitle` を冒頭に記述すればよい
- 表紙の内容は、次の形で指定する ( `tex-003.tex` )
  - ▶ 表題 : `\title{表題}`
  - ▶ 作者 : `\author{作者名\thanks{所属}}`
  - ▶ 日時 : `\date{作成日時}`
- 表紙を独立にするには、
  - ▶ `titlepage` も追加する (そうしないと、一緒に表示される)

## □ 章立て

- 文章の章立てを行いたい場合は、章題を追加する ( `tex-004.tex` )
  - ▶ 章題は `\section`, `\subsection ..` などで指定する
  - ▶ 章題は自動的に番号が振られ、ポイントが大きくなり目立つようになる

## □ 目次

- 目次を入れるには、`\tableofcontents` を挿入するだけです ( `tex-005.tex` )

# TeX 文章 (4) : 式

---

## □ TeX 文章の中での、式の扱い

○ TeX 文章の中では、式は文章自身とは別に扱われます

▶ 式は、文章の一部(一種の単語)として扱う事も、独立した物とも扱えます

## □ TeX の式 (tex-006.tex)

○ 文中の式は「\$」～「\$」で挟みます

○ 独立した物として扱う場合は、「\」～「\」で挟みます

## □ 式特有のマクロ

○ TeX では、様々な式を表現するための命令がある (他にも色々:ググれ)

▶ 指数 / 添字 :  $x^2$  ( $x$  の二乗) ,  $x_i$  ( $i$  番目の  $x$ )

▶ 分数 :  $\frac{\text{分子}}{\text{分母}}$

▶ 総和 :  $\sum_{i=1}^n i^2$  ( $i$  が 1 から  $n$  までの  $i^2$  の和)

▶ 積分 :  $\int_a^b f(x) dx$  ( $f(x)$  を  $a$  から  $b$  まで定積分)

○ 注意 : 「{」～「}」は範囲指定に利用されている

▶  $\frac{1}{2}$  は  $\frac{1}{2}$  と同じ

# ファイルの取込と Mathematica

---

## □ TeX 文章の分割

- TeX の文章は複数のファイルに分割できる (tex-007.tex)

- ▶ `\input` を利用して、他のファイルを取り込める

## □ Mathematica による式の出力 (tex-008.tex)

- `Put[expr,"file.tex"]` とすれば、`expr` の式を `file.tex` に保存できる

- ▶ Mathematica の式を TeX 形式でファイルに保存して `\input` で取り込む

## □ Mathematica による図の出力 (tex-009.tex)

- `Export["file.eps",Plot[..]]` とすれば、`Plot[..]` の内容を EPS 形式で保存できる

- ▶ Mathematica の図を EPS 形式でファイルに保存して `\includegraphics` で取り込む

# [演習] Mathematica の結果の TeX での利用方法

---

## □ Mathematica で三次関数のグラフ描画

- 三次関数  $f(x) = x^3 - x + QQQQ$  (QQQQ は学生番号) を描画

## □ [演習 1]

- Mathematica で三次関数のグラフを作成する

▷ sample-20130618.nb を参照

## □ [演習 2]

- Mathematica の結果をファイルに出力

▷ expr.tex / graph.eps の作成

## □ [演習 3]

- Mathematica の結果取り込んだ TeX の typeset

▷ sample-20130618.tex を参照

## □ [演習 4]

- pdf ファイルの作成と、提出

▷ sample-20130618.pdf を参照