

コンピュータ概論 A/B

-- TeX --

(TeX と Mathematica (2))

数学科 栗野 俊一 (TA: 佐藤 淳 [院生 1 年])

2015/11/10 コンピュータ概

論

伝言

私語は慎むように !!

□ 席は自由です (出席パスワード : 20151110)

○ できるだけ前に詰めよう

○ 教室にきたら直ぐにやる事

▶ PC の電源 On / ネットワーク接続 / Web を参照する / skype を起動する

□ 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>

□ ネットワークの認証が変りました

○ これまでは、ダイアログ表示だったのですが、先日から認証ページに代りました

LAN の認証方式の変更

□ LAN の認証方式の変更

○ 11 月から、LAN の認証方式が変更されました

▶ 従来：ダイアログボックスが出る (ID/PW を入力)

▶ 新規：認証ページが開く (ID/PW を入力)

□ 無線 LAN の変更で接続できなかつたら

○ Internet Explorer を開く

▶ ブラウザーのキャッシュをクリアする

▶ 一旦、全てのページを閉じる

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

▶ とにかく https でなく http の場所

○ ※ https のページを開くと、トラブルになるらしい

▶ 最初に http のページを開くと認証ページに飛ぶ

○ ※ google は、https に飛されるので、他のページが良い

▶ <http://www.yahoo.co.jp> もダメ？

○ ※ CST Portal も、同様

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

前回(2015/10/27)の内容

□ 講義

- タイプセットの仕方 : .tex ファイルから .pdf を作る方法
- TeX ファイルの作成方法 : サクラエディタで .tex ファイル作成する
 - ▶ 「おまじない」の三行+コンテンツ

```
\documentstyle{jsarticle}
```

```
\begin{document}
```

ここに、文章を入力

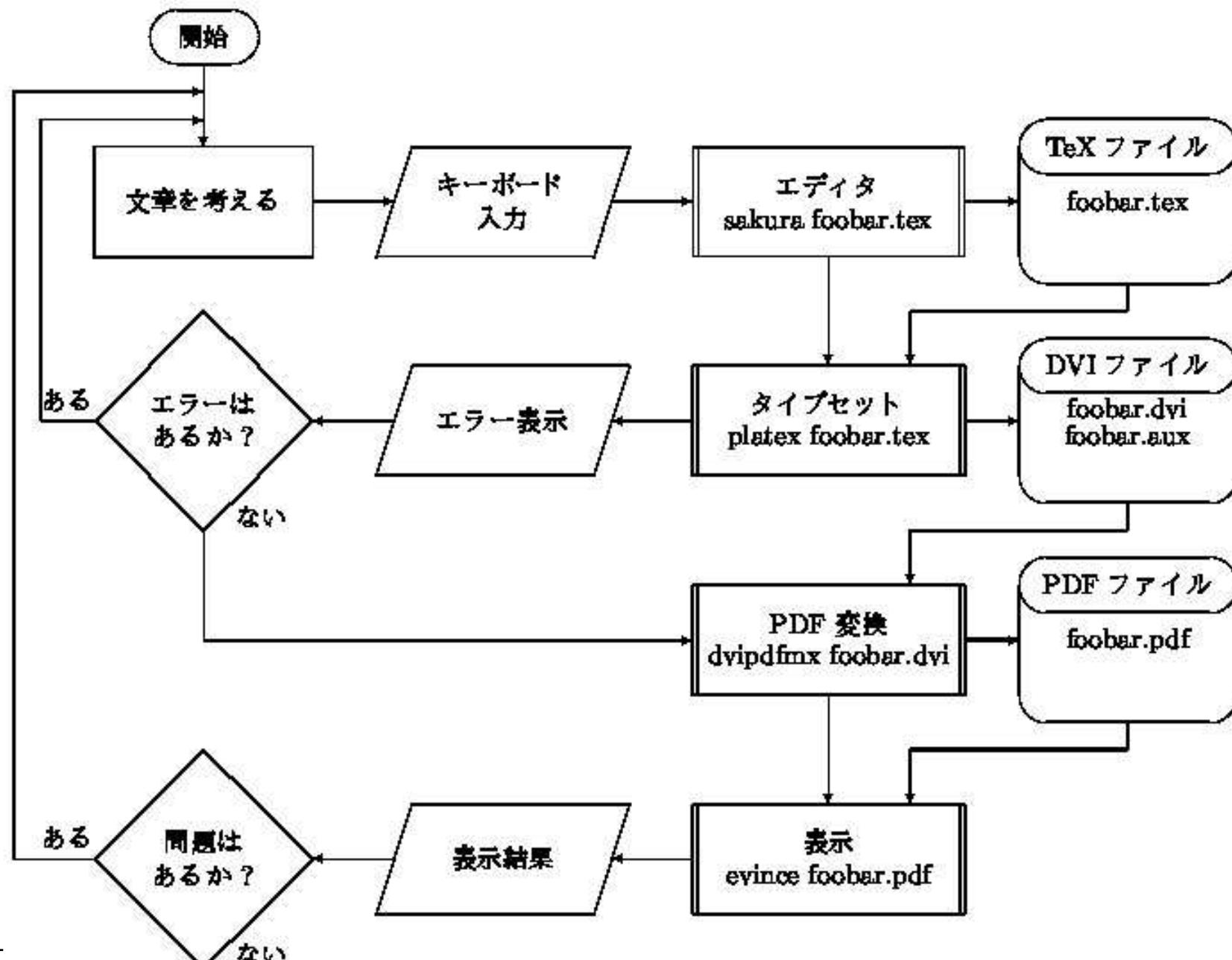
```
\end{document}
```

- (pLa)TeX(2e) による様々な表現
 - ▶ タイトルの付け方
 - ▶ 章立て / 目次の作成 / etc.. / 後は「検索」

□ 演習

- TeX でのタイプセット (Ubuntu 上で、コマンドで指定)
 - ▶ platex(ピー・ラテック), dvipdfmx(ディブイアイピーデーエフエムエックス), evince(エヴィンス)

TeX 文章の作成フロー(再)



本日(2015/11/10)の予定

- 本日(2015/11/10)の予定
 - TeX と Mathematica の連携 (前回[2015/10/27]の積み残し)
 - 「メタシステム」とは？ (前々々回[2015/10/13]の積み残し)
- 本日(2015/11/10)の目標
 - Mathematica の結果を TeX で利用する
- 演習
 - [演習 1] Mathematica の数式を TeX で利用する (前回[2015/10/27]の積み残し)

本日の課題 (2015/11/10)

□ 前回 (2015/10/27) の課題 (急遽変更した / Portal は無変更)

○ CST Portal に以下のファイルを提出しなさい

- ▶ ファイル名 : 20151110-QQQQ.tex (QQQQ は学生番号)
- ▶ 表題 : 自分で作成した TeX ファイル
- ▶ 内容 : サクラエディタで、最初から TeX ファイルを作成する
- ▶ 条件 : 本文中に自分の学籍番号と名前を変更
- ▶ 形式 : テキストファイル (TeX ファイル形式)

□ 今回 (2015/11/10) の課題 (前回の課題)

○ CST Portal に以下のファイルを提出しなさい

- ▶ ファイル名 : 20151110-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-20151027.tex 参照)

ファイルの取込と Mathematica (再)

□ TeX 文章の分割

- 「TeX の文章」は「複数のファイルに分割」できる
 - ▶ `\input / \includegraphics` を利用して、「他のファイル」を取り込める
 - ▶ 文章の「再利用」が可能 (「コピペ」とは違う形)
 - ▶ 「別のファイル」なので「別に作る」事ができる

□ Mathematica による「式」の出力

- `TeXFrom` : Mathematica の「式」を、TeX 形式に変換する関数
- `Put[TeXForm[「式」], "c:/usr/tex/20151110/expr.tex"]` とする
- 「式」の内容が、TeX 形式に変換されて `expr.tex` に保存される
 - ▶ Mathematica の式を TeX 形式でファイルに保存して `\input` で取り込む

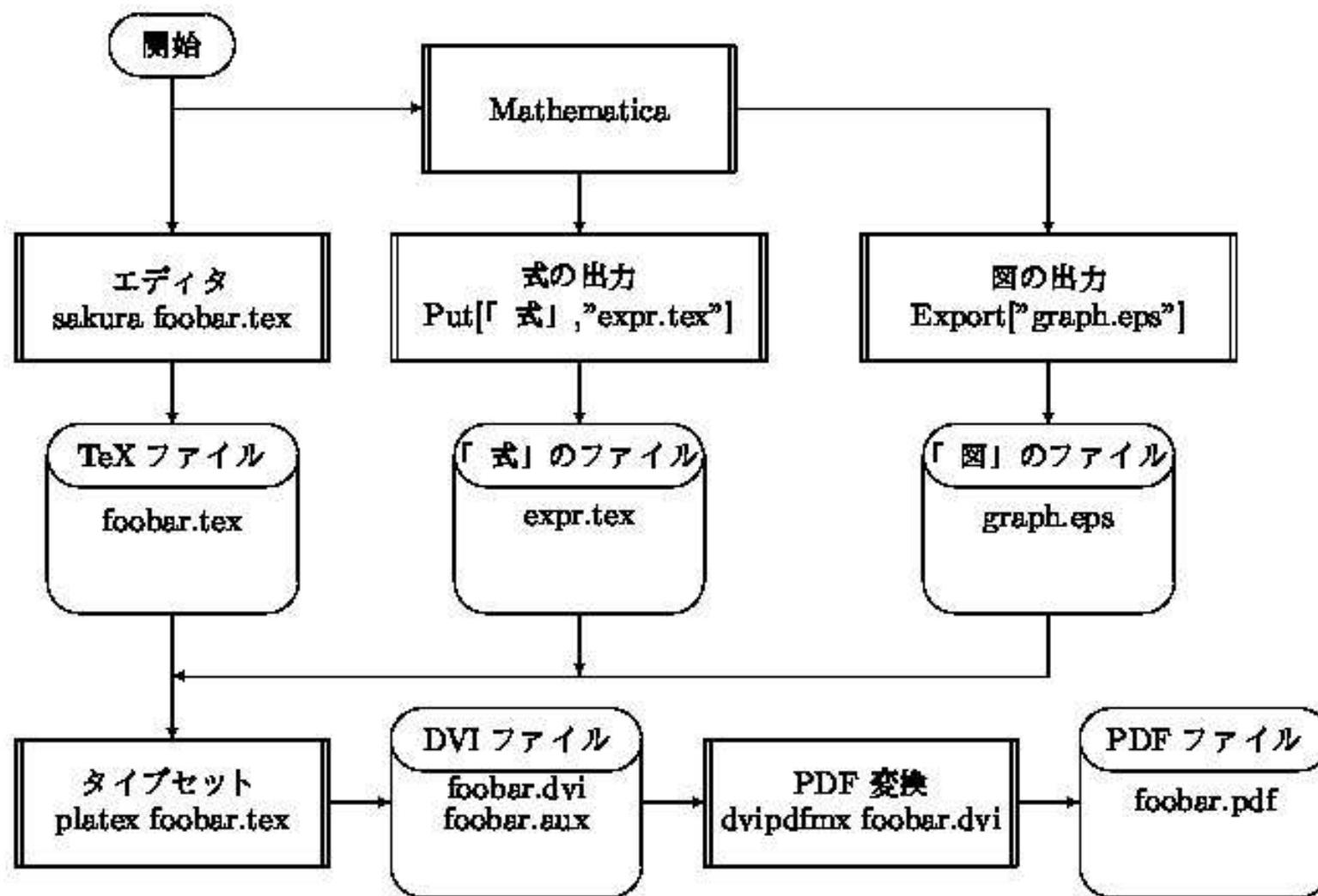
□ Mathematica による「図」の出力

- `Export["c:/usr/tex/20151110/graph.eps", Plot[...]]` とすれば、`Plot[...]` の内容を EPS 形式で `graph.eps` に保存できる
 - ▶ Mathematica の図を EPS 形式でファイルに保存して `\includegraphics` で取り込む

TeX と Mathematica の連携

2015/11/10

TeX と Mathematica の連携フロー



[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-20151027.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-20151027.tex を参照

□ [演習 4]

- pdf ファイルの作成と、提出

▷ sample-20151027.pdf を参照

UI : ユーザ・インターフェース

□ 「UI : ユーザ・インターフェース」とは (What)

○ 人間と計算機の仲立ち(の部分の形式や、それを担うプログラム)事

▶ インターフェース : 境界面 / ユーザ : (計算機にとっての) 人間

▶ cf. マン・マシンインターフェース

□ UI が必要な理由 (Why)

○ 「異なるもの(人間と機械)の間」には、お互いの「情報交換を行う仕組」が必要

▶ cf. 日本人とフランス人が会話をするために英語(インターフェース)を使う

○ 計算機の「『生』の情報」は、人間には分り難い

▶ 人間に分かり易い形にした「表現」が必要になる

▶ cf. 「デスクトップ(アイコン)」は、「ファイルシステム(ファイル)」の表現

▶ ※ 「ファイル」には、「名前」(という属性)はあっても「アイコン」はない(Explorer が勝手に付けている)

□ UI を担うプログラム (How)

○ UI を決めるのは、「UI プログラム」

▶ 「UI プログラム」が変れば「UI も変る」

▶ cf. 同じ日本語でも「京都弁」だと... ?

○ Windows では、基本 UI が固定 (設定変更は可能だが..)

▶ Ubuntu(Linux) では、UI を変更できる(自分で選べる)

CLI : コマンド・ライン・インターフェース

□ CLI : コマンド・ライン・インターフェース

○ CLI とは (What)

- ▶ (キーボードから)コマンドを入力し、その結果が(画面に)表示される仕組み
- ▶ 正例(CUI) : Ubuntu での TeX ファイルのタイプセットの時に利用している
- ▶ 反例(GUI) : Windows での操作 (マウスを利用する)

○ 「コマンド」の例 (Ubuntu での操作)

- ▶ cd, ls, platex, dvipdfmx, evince

○ 「表示結果」は、単なるログ

- ▶ その「表示」内容は「一時的なもの」で記録されない
- ▶ 「記録される」のは、「コマンドの効果」

□ CLI の特徴

○ 「コマンド」で、計算機の「操作」ができる

- ▶ 「対象」によって「コマンド(名)」は異なるが、対象に拘わらず「使い方」は共通
- ▶ 例: 「コマンドプロンプト」は、Windows 98/7/8/10 で共通
- ▶ 例: 「ターミナル」は、Ubuntu(Linux), MacOS で共通 (実はスマホにもある)
- ▶ 反例: GUI は、全部異なる

○ 「操作結果」は、「明示的に命令」しない限り表示されない

- ▶ 正例: platex した後に、ls して dvi ファイルの作成を確認
- ▶ 反例: 作成すると、直にアイコンが増える

CLI(CUI) vs GUI

□ GUI : グラフィカル・ユーザ・インターフェース

- 「結果/対象」を見ながら、メニュー形式で、「操作内容」を指定して操作

- ▶「結果」は、直に表示され、反映される

- ▶「操作」も「表示される」ので、「選ぶ」だけ (その場で選ぶ必要がある)

- GUI の特徴

- ▶「見ながら」、「その場で探しながら」操作できる

- ▶直感的で、学習コストが小さい

□ CUI : キャラクタ・ユーザ・インターフェース

- 「結果」は、その場では表示されない

- ▶「操作」は、「コマンド(名とその機能)」を学習しないとできない

- CUI の特徴

- ▶「頭の中で操作」する (必要がある/できる)

- ▶「見ずにできる」ので、「まとめて処理 (マクロ化)」できる

□ CUI(間接) vs GUI(直接)

- GUI は、「直接」操作するので、「分り易い」が、「もどかしい」

- CUI は、「間接(メタ)」操作なので、「難しい」が、「強力」