



# 情報リテラシ第二(3b)

担当: 増原英彦 TA: Jizhe Chenxin, Lubis Luthfan Anshar, 小串 智希

## 第4週

- プレゼンテーション (批評)
- 文書作成 (数式と図)

# LaTeXパッケージ

- パッケージ: 追加機能
  - LaTeXは基本機能だけでなく追加機能が沢山提供されている
  - 例: 「URLをきれいに表示する機能」
- プリアンブルに  
`\usepackage{パッケージ名}`  
と書くと
- 本文中でその機能を使える
- 参考: [TeX Wiki 目的別パッケージ簡易リファレンス](#)

```
\documentclass [a4j,11pt,autodetect-engine]
{jsarticle}

\usepackage{url}

\begin{document}

Abc.com社のページは
\url{http://abc.com/this_page?
a=1&b=2}
です。
```

Abc.com 社のページは `http://abc.com/this_page?a=1&b=2` です。

# 数式の例

```
\[
\frac{\pi}{2} =
\left( \int_0^{\infty} \frac{\sin x}{\sqrt{x}} dx \right)^2 =
\sum_{k=0}^{\infty} \frac{(2k)!}{2^{2k}(k!)^2} \frac{1}{2k+1} =
\prod_{k=1}^{\infty} \frac{4k^2}{4k^2 - 1}
\]
```

$$\frac{\pi}{2} = \left( \int_0^{\infty} \frac{\sin x}{\sqrt{x}} dx \right)^2 = \sum_{k=0}^{\infty} \frac{(2k)!}{2^{2k}(k!)^2} \frac{1}{2k+1} = \prod_{k=1}^{\infty} \frac{4k^2}{4k^2 - 1}$$

# 数式の注意点

\$1\$から\$N\$までの  
和  $\sum_{x=1}^N x$  を求めよ。  
 $\left[ \sum_{x=1}^N x \right]$

- LaTeXには「数式モード」と「通常文モード」がある  
→ 同じ文字が違う働きをする
  - 例: 数式モードでしか使えない文字がある (例えば  $_$ )
  - 例: アルファベットの並びの意味が変わる (変数の掛け算)
- 数式モードには「行内」(inline)と「別行立て」(display)がある  
→ 同じ数学記号が違うレイアウトになる
  - 例:

1 から  $N$  までの和  $\sum_{x=1}^N x$  を求めよ。

$$\sum_{x=1}^N x$$

# この数式をLaTeXで書くには?

- 数式には色々ある
  - 行列
  - ギリシャ文字
  - 様々な記号: 積分、微分、演算子、...
- 多過ぎて紹介できないので参考書から探して書き写す
  - 例えば: <https://en.wikibooks.org/wiki/LaTeX/Mathematics>

# 図の挿入

1. 図ファイルをOverleafにアップロードする  
(PDF, PNG, JPEG等)
2. プリアンブルに挿入  
`\usepackage[dvipdfmx]{graphicx}`
3. 図を挿入する場所に挿入  
`\includegraphics{アップロードしたファイル名}`

注意:

- 図ファイルが大きいと結果のPDFも大きくなる
- 拡大率、一部の切り取り、回転などもできる  
([Overleaf社の解説](#)や[マニュアル](#)を参照)

```
\documentclass [a4j,11pt,autodetect-engine]
{jsarticle}

\usepackage[dvipdfmx]{graphicx}

\begin{document}
こんな絵
\includegraphics{pig.pdf}
を描いてみました。
```

# 図表のレイアウト

- 図表を置く場所は様々
  - 本文の途中 (h)
  - ページの上下 (t, b)
  - 独立したページに (p)
- LaTeXでの図表の配置方法
  - 図や表と見出しを本文中の適当な所に書く
  - 配置したい場所の希望を指定する
  - LaTeXがよろしく配置してくれる

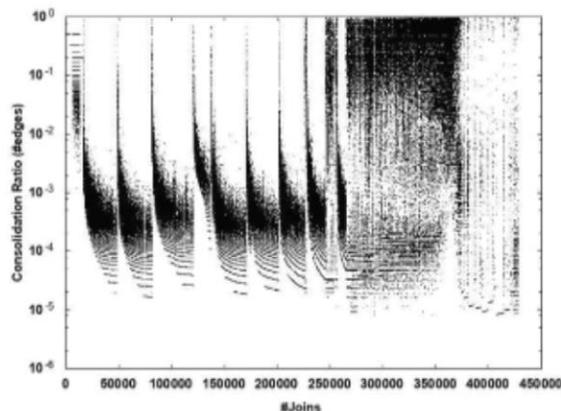


図3 CNM法におけるクラスターの合併比率の推移 (片対数)

図3は $n$ 回目の合併を、 $c_k := \text{Join}(c_i, c_j)$ とすると、 $(n, \text{ratio}(c_i, c_j))$ にプロットしている。 $|c|$ はクラスターのサイズである。この図より、解析の前半で、8つの巨大なクラスターの成長を観測できた。われわれはこの属って成長したクラスターがCNM法をmixiネットワークに適用したときの主な計算速度劣化の原因と考えた。

不均衡な合併はデンドログラムの高さを成長させ、 $d \approx \log n$ ではなく、 $d \approx n$ となり $O(n^2 \log n)$ となる。

## 4 提案手法

前節ではCNM法の問題点を指摘した。本節ではCNM法の問題点を改善し、速度を劇的に向上させる3つのヒューリスティックを提案する。

提案する手法を図4に掲げる。全体の構造はCNM法とほぼ同じである。異なるのは合併するクラスターの対を決める戦略である。CNM法では $\Delta Q_{c_i, c_j}^c$ を用いていた。本手法ではそれに加えて合併比率 $(\text{ratio}(c_i, c_j))$ を用いる。このヒューリスティックは、合併の均衡を保ち、平均したクラスターの成長を促すことにより、合併の不均衡による手法の性能の劣化を抑えるように設計した。

さて、本稿ではここまでクラスターサイズ $(|c_i|)$ を定義してこなかった。ここで、3種類のクラスターサイズの定義を行うことにより、3種類のヒューリスティックHE, HN, HE'を定義する。以下にそれぞれのヒューリスティックのクラスターサイズ $|c_i|$ の定義を示す。

# 図表のレイアウト

- 図表を置く場所は様々

- 本文の途中 (h)
- ページの上下 (t, b)
- 独立したページに (p)

- LaTeXでの図表の配置方法

- 図や表と見出しを本文中の適当な所に書く
- 配置したい場所の希望を指定する
- LaTeXがよろしく配置してくれる

```
\begin{figure}[tbp]
  \centering
  \includegraphics{pig.pdf}
  \caption{ブタです}
\end{figure}
```

# 数式と図のまとめ

図を挿入するための準備

```
\documentclass [a4j,11pt,autodot,ps-eps-engine] {jsarticle}
\usepackage[dvipdfmx]{graphicx}
\begin{document}
```

\$1\$から\$N\$までの

和  $\sum_{x=1}^N x$  を求めよ。

```
\[ \sum_{x=1}^N x \]
```

```
\begin{figure}[tbp]
```

```
\centering
```

```
\includegraphics{pig.pdf}
```

```
\caption{ブタです}
```

```
\end{figure}
```

```
\end{document}
```

図のレイアウト

行内数式

別行立て数式

希望の配置

中央揃え

図の挿入

図見出し



図1 ブタです

1 から  $N$  までの和  $\sum_{x=1}^N x$  を求めよ。

$$\sum_{x=1}^N x$$

# 課題: プレゼンテーション (スライド批評) 指定されたスライドを批評せよ

## 手順

1. 講義webページにある「第1次スライド」と「批評割当表」をダウンロード
  - 割当表「提出状況」のYES/NOは本人のスライドの提出状況
2. 自分の学籍番号の行の割当「あ」「い」「う」「え」が割り当てられた4つのスライド番号である。該当するスライドに対して批評を書け
  - アニメーションを使っている場合があるので、「スライドショー」で見るとよい
  - 次回に「この改善案は建設的でしたか?」という逆評価を行う
3. 改善案はオンラインのページに入力  
(成績評価の対象となる)
  - 期限: 7月18日(土)

### 建設的批評:

- どう直すべきか具体的に指摘
- 良い点を「良い」と指摘

# (例) 加計学園疑獄の概要

- ▶ 安倍晋三総理の知人が理事長を務める学校法人「加計学園」が、国の国家戦略特区に獣医学部の新設を申請し、2016年11月に認められ、開設は2018年4月の予定。
- ▶ 獣医学部の新設予定地は、広さ16.8ヘクタール、評価額36億7500万円もの土地なのに、今治市からタダで同学園に譲渡されている。同学園には愛媛県と今治市が96億円もの補助金を支払っている。
- ▶ 国家戦略特区を担当する内閣府と、実際に学部の新設を認可する文部科学省が協議している。報道によれば、審査のスケジュールをめぐる難色を示した文科省に対し、内閣府が「官邸の最高レベルが言っている」「総理ご意向」などと開設を急がせた。

# 文章の量

- 注意点

- 聴衆はスライドに書いた文章を読んでしまう
  - 話を聞いてもらえない
- 小さい文字は遠くからは読めない

- とるべき方針

- 重要なキーワードのみにする (1項目はスライド1行程度)
- 口頭で説明することだけ書く
- 最低サイズ: 24ポイント

# (例) 加計学園疑獄の概要

- ▶ 学校法人「加計学園」の申請

- ▶ 申請内容: 獣医学部の新設

- ▶ 設置先: 国家戦略特区

- ▶ 学園への補助

- ▶ 今治市から土地を譲渡 (広さ16.8ha、評価額36億円)

- ▶ 愛媛県と今治市から補助金 (96億円)

- ▶ 政府の対応

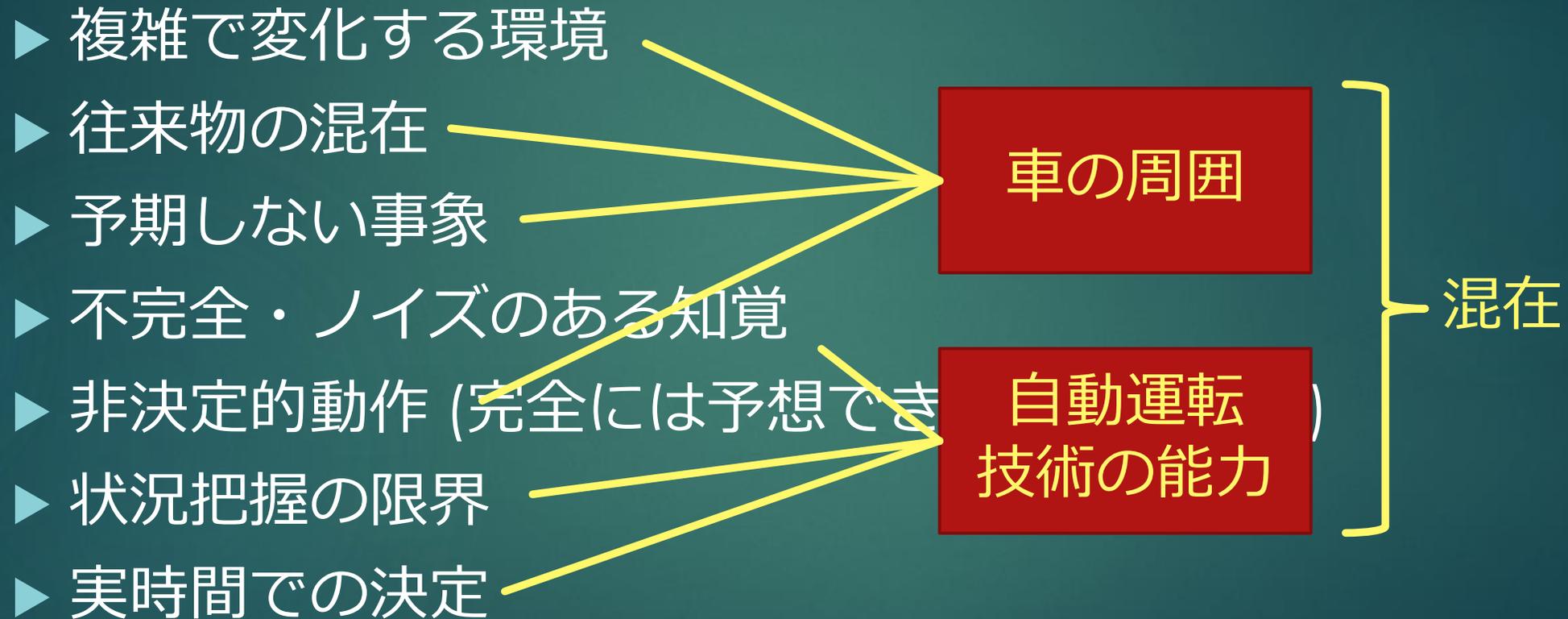
- ▶ 内閣府: 国家戦略特区を担当

- ▶ 文部科学省: 学部新設認可を担当

スケジュールをめぐる対立  
(報道による)



# (例) 自動運転: 安全性の向上?



# 箇条書き

- 原則

- 並列・直列する事柄を並べる

- 注意点

- 「同じレベルで並列・直列しているか？」

- とるべき方針

- 並んでいるものの関係を常に考える

# (例) 自動運転: 安全性の向上?

- ▶ 複雑で変化する環境
  - ▶ 往来物の混在
  - ▶ 予期しない事象
  - ▶ 非決定的動作 (完全には予想できない動作結果)
- ▶ 自動運転技術の不足
  - ▶ 不完全・ノイズのある知覚
  - ▶ 状況把握の限界
  - ▶ 実時間での決定

# *motivating example:* **SPARK**

```
class MyCoolRddApp {  
  val param = ...  
  val log = new Log(...)  
  ...  
  def work(rdd: RDD[Int]) {  
    rdd.map(x => x + param)  
        .reduce(...)  
  }  
}
```

## **PROBLEM:**

(x => x + param)  
not serializable  
because it captures  
this of type  
MyCoolRddApp  
which is itself not  
serializable

# motivating example: SPARK

```
class MyCoolRddApp {  
  val param = ...  
  val log = new Log(...)  
  ...  
  def work(rdd: RDD[Int]) {  
    rdd.map(x => x + param)  
      .reduce(...)  
  }  
}
```

## PROBLEM:

(x => x + param)

*not serializable*

because it captures  
this of type

MyCoolRddApp

which is itself not  
serializable

## motivating example: SPARK

```
class MyCoolRddApp {  
  val param = ...  
  val log = new Log(...)  
  
  def work(rdd: RDD[Int]) {  
    rdd.map(x => x + param)  
    .reduce(...)  
  }  
}
```

### PROBLEM:

(x => x + param)  
not serializable  
because it captures  
this of type  
MyCoolRddApp  
which is itself not  
serializable

<http://docs.scala-lang.org/sips/pending/spores.html>

# 色使い

- 注意点

- 多すぎになりやすい
- 作成環境と発表環境の違い (機材・知覚)

- とるべき方針

- 重要なキーワード数個に色を付ける
- 色の種類は必要最小限
- 色が無くても分かるように
- コントラストに注意する

# 色の使い方: コントラストに気を付けよ

- 良い色使い

- 明るい背景 + 暗い文字
- 暗い背景 + 明る文字

プレゼン  
テーション

プレゼン  
テーション

- 悪い色使い

- 明るい背景 + 明るい文字
- 暗い背景 + 暗い文字
- (明暗と暗明の混在)

プレゼン  
テーション

プレゼン  
テーション

プレゼン  
テーション

# motivating example: SFARK

```
class HelloWorldApp {
  val param = "SFARK"
  def top = "SFARK"

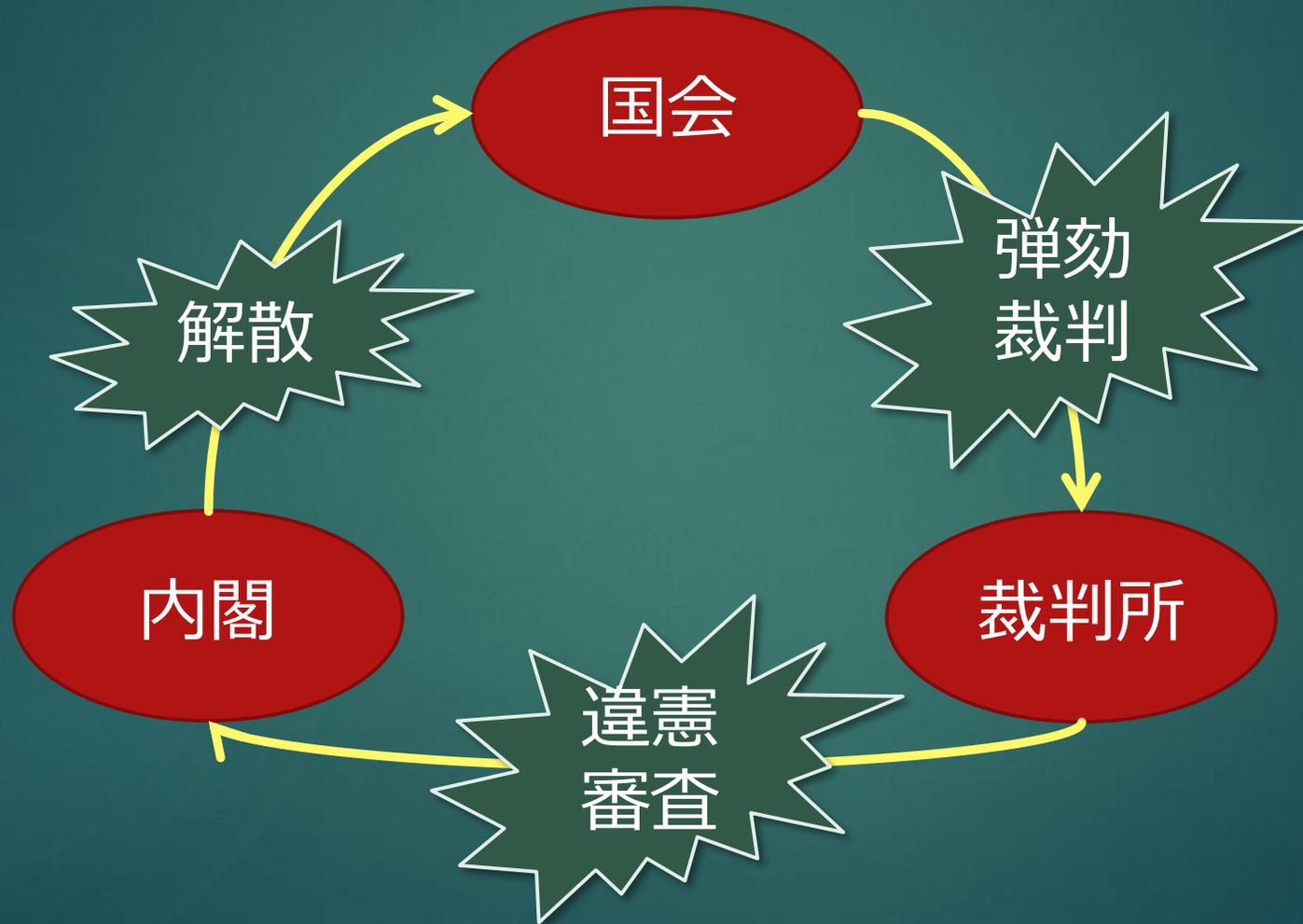
  def work() {
    println("Hello World!")
    println("SFARK")
  }
}
```

**PROBLEM:**  
( $\lambda$   $\Rightarrow$   $\lambda$  = param)  
not serializable  
because it captures  
this of type  
HelloWorldApp  
which is itself not  
serializable

# (例) 三権分立

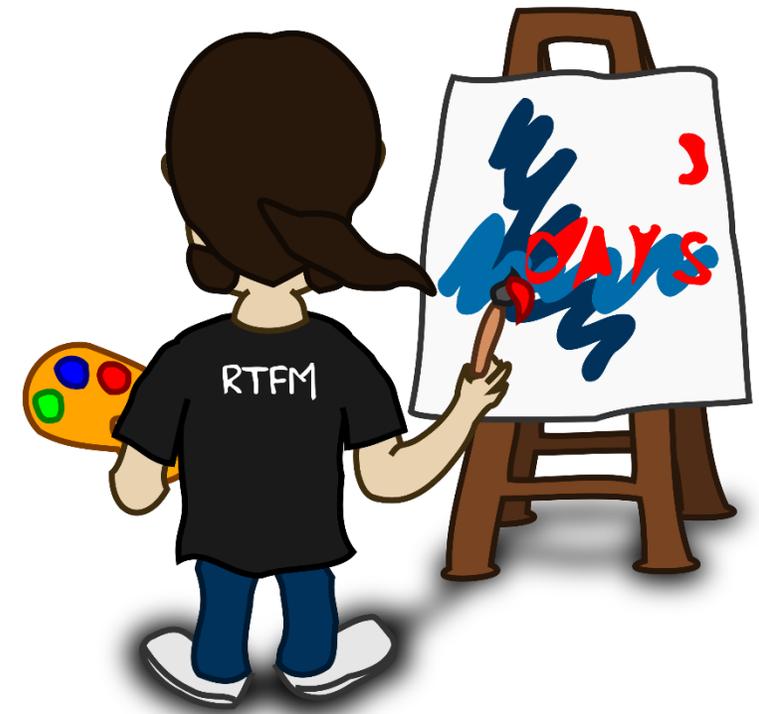
- ▶ 国会
  - ▶ 弾劾裁判を行う
  - ▶ 弾劾裁判によって裁判官は罷免されうる
- ▶ 裁判所
  - ▶ 法律の違憲審査をする
  - ▶ 審査される法律は内閣が制定
- ▶ 内閣
  - ▶ 国会を解散できる

# (例) 三権分立



# 図の使い方

- 原則
  - 関係を示すのに有効
  - 数量を示すのに有効 (グラフ vs 表)
- 注意点
  - 図形の意味は伝わりにくい
  - 一貫性がないと聴衆は混乱する
  - 理解できる要素数には限りがある
  - 挿絵は注意を疎外しないように



# アニメーション

- 注意点
  - 使い過ぎになりやすい
  - 見逃される
- とるべき方針
  - 順に見せる・隠す
    - 聴衆の注意を集中させたい場合
    - 図の位置関係を変えたくない場合
  - 動きをつける: 付加情報に限る

# 課題: プレゼンテーション (スライド批評) 指定されたスライドを批評せよ

## 手順

1. 講義webページにある「第1次スライド」と「批評割当表」をダウンロード
  - 割当表「提出状況」のYES/NOは本人のスライドの提出状況
2. 自分の学籍番号の行の割当「あ」「い」「う」「え」が割り当てられた4つのスライド番号である。該当するスライドに対して批評を書け
  - アニメーションを使っている場合があるので、「スライドショー」で見るとよい
  - 次回に「この改善案は建設的でしたか?」という逆評価を行う
3. 改善案はオンラインのページに入力  
(成績評価の対象となる)
  - 期限: 7月18日(土)

### 建設的批評:

- どう直すべきか具体的に指摘
- 良い点を「良い」と指摘