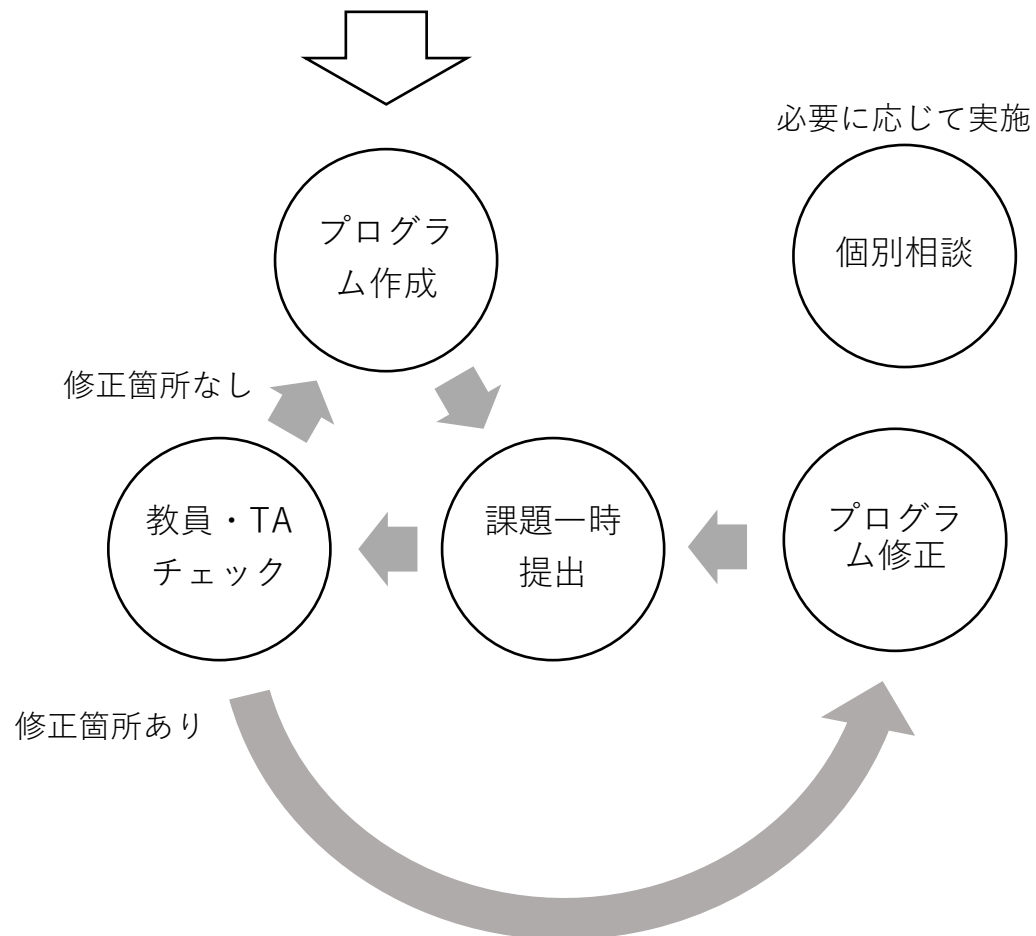


# 情報理工学実験 モバイルWebアプリケーション

情報基盤センター／飯田勝吉

## 講師による事前説明



# 事前の作業

---

1. ELMSの確認
2. 指導書のダウンロード (\*)
3. 指定ブラウザのインストール
  1. 標準Firefoxと共存するには特別な設定が必要
4. 演習用プログラムのダウンロード
5. Slackの加入

# 初回授業

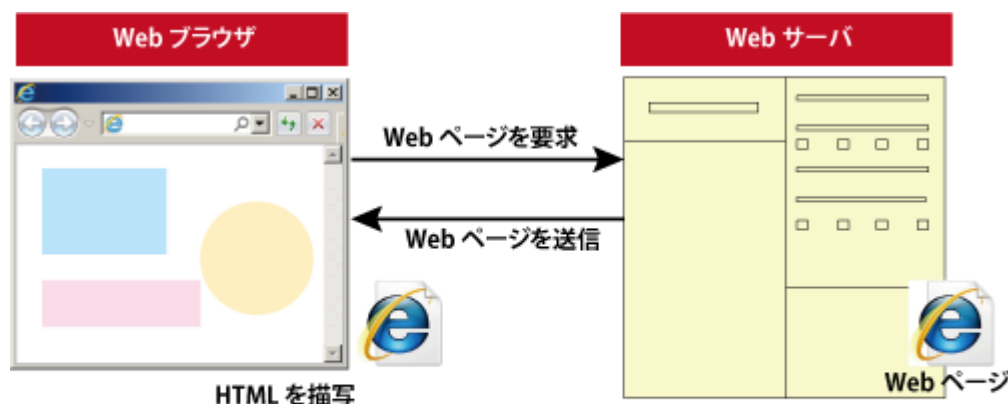
---

- ▶ 10/29(金)
  - ▶ 8:45 までにZoomに入ってください
  - ▶ 8:45～9:15：講師説明
  - ▶ 9:15～12:00：ミュート状態でプログラム開発を実施
    - ▶ 質問があればSlackで。
  - ▶ 状況により、全体アナウンスを実施
  - ▶ 1 限と 2 限の間の休憩時間はとります

# Webとは

## ▶ World Wide Web

- ▶ WebブラウザとWebサーバからなるシステム
- ▶ ハイパーリンクと呼ばれる情報のリンクが網目のようにつながっている「世界規模の蜘蛛の巣」



Web ブラウザは、HTTP で取得した HTML (など) を画面に描写するソフト

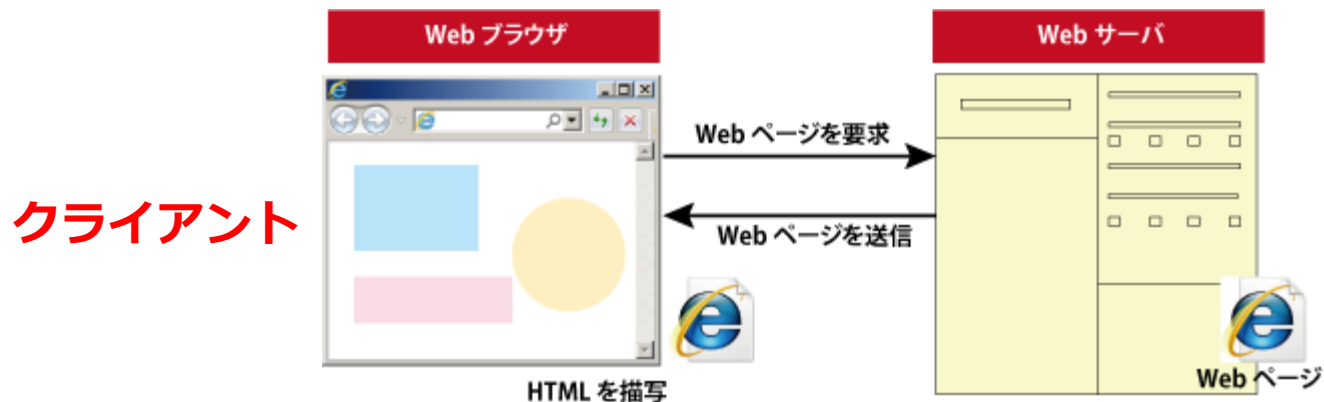
Web サーバは、HTTP で要求された HTML (など) を配信するサーバ



蜘蛛の巣のように見えるから Web と呼ぶ

NPO法人ヒューマンICT推進委員会Webサイトより引用

# クライアントサイドおよびサーバサイドプログラム



Web ブラウザは、HTTP で取得した HTML (など) を画面に描写するソフト

Web サーバは、HTTP で要求された HTML (など) を配信するサーバ

## ▶ クライアントサイド

- ▶ クライアント側でのプログラムの実行
- ▶ 例 : Google map
- ▶ JavaScript

## ▶ サーバサイド

- ▶ サーバ側でのプログラム実行
  - ▶ 例 : ユーザ・商品データベースの探索
  - ▶ PHP, Ruby, Perl
  - ▶ Javaサーブレット、C言語

# JavaScriptとクライアントサイドプログラム

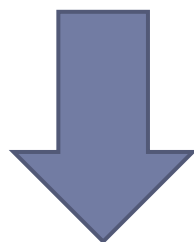
---

- ▶ Javascript
  - ▶ ブラウザNetscape Navigator 2.0に実装（1996年）
  - ▶ 標準化団体EcmaインターナショナルがEcmaScriptとして標準化（1997年）
  - ▶ 現在、ほぼすべてのブラウザに実装済み
  - ▶ C言語に似た手続き型言語
  
- ▶ クライアントサイドプログラムでできること
  1. Webブラウザ上でのプログラムの実行
    - ▶ 四則演算、文字列操作、条件分岐、ループ、関数呼び出し等
  2. イベントの検出
    1. マウス操作など
  3. Webページ画面の書き換え
    - ▶ テキスト、画像の追加・修正・削除など
  4. 非同期通信
    - ▶ ページ表示中に動的に通信すること

# 本実験の目的

---

- ▶ ブラウザ上で動作するクライアントサイドスクリプト言語JavaScriptの基本を体験的に学ぶこと
  - ▶ クライアントサイドスクリプト言語の4つの機能の学習
  - ▶ jQuery, jQuery MobileというJavaScriptライブラリの利用
  - ▶ JavaScriptプログラミングで気を付けるべきことの学習



**クライアントサイドWebプログラミングやWeb APIに慣れる**



# 実験の予定

回数	日付	予定内容
1	10/29(金)	JavaScriptの基本 ( 1 )
2	11/01(月)	JavaScriptの基本 ( 2 )
3	11/02(火)	JavaScriptの応用
4	11/05(金)	ライブラリjQueryの基礎
5	11/08(月)	ライブラリjQuery Mobileの基礎
6	11/09(火)	JavaScriptのセキュリティ

# 担当者

---

## ▶ 教員

- ▶ 飯田 勝吉（いいだ かつよし）
  - ▶ 情報基盤センター／先端ネットワーク研

## ▶ T A

- ▶ 佐々木隆稀（ささき りゅうき）
  - ▶ 先端ネットワーク研M 2
- ▶ 漆原 航平（うるしばら こうへい）
  - ▶ 先端ネットワーク研M 2

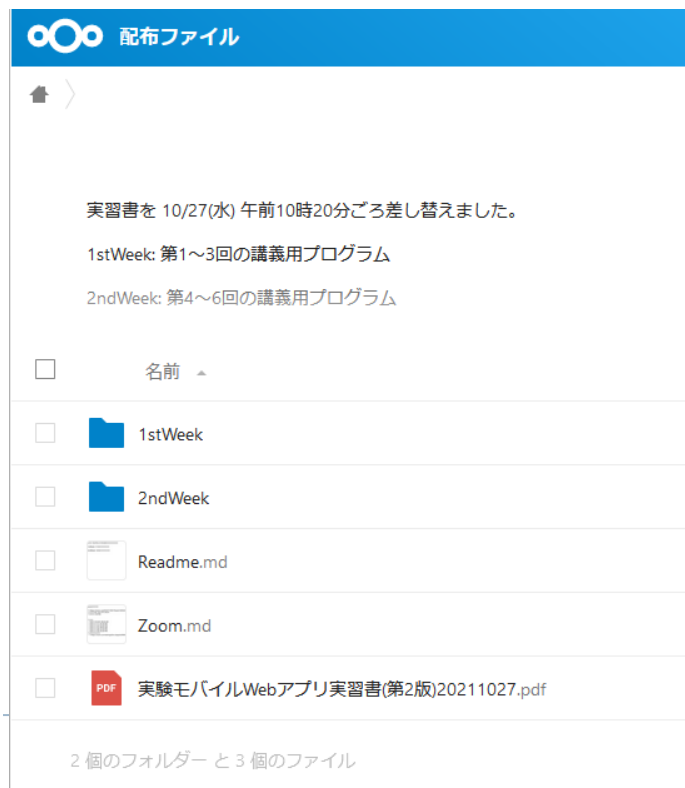
# 事前準備

---

1. 指定ブラウザのインストール
  2. 演習ファイルのダウンロード
  3. Slackの加入手続き
  4. Zoomの最新版へのアップデート (2021/10/27現在 5.8.3)
- ▶ 終わっていない人がいたら、ELMS/指導書をみて上記を行ってください。

# 資料等の配布場所など

- ▶ (\*) 演習ファイル提供URL：（実習書PDF版、Zoom接続情報、演習用サンプルプログラムを含む）
- <https://enpit01.iic.hokudai.ac.jp/nextcloud/s/G3kCZSTKXQbRiBp>
- ▶ ELMSにもリンクを書いています。



# 指定ブラウザ

---

- ▶ 指定ブラウザ : Mozilla Firefox 38.8.0 ESR
- ▶ Windows 版:  
<https://ftp.mozilla.org/pub/firefox/releases/38.8.0esr/win32/ja/>
- ▶ MacOS版:  
<https://ftp.mozilla.org/pub/firefox/releases/38.8.0esr/mac/ja-JP-mac/>
- ▶ Linux版:  
[https://ftp.mozilla.org/pub/firefox/releases/38.8.0esr/linux-x86\\_64/ja/](https://ftp.mozilla.org/pub/firefox/releases/38.8.0esr/linux-x86_64/ja/)
- ▶ ELMS/指導書にもリンクを書いています
- ▶ 普段、Firefoxを使っている人は、複数のFirefoxバージョンを共存する方法があります

# Slackの使い方

---

## ▶ 加入URL

- ▶ [https://join.slack.com/t/mwa-2021/shared\\_invite/zt-x8qqiub8-UQpuS07xkS97SzG91EWCnw](https://join.slack.com/t/mwa-2021/shared_invite/zt-x8qqiub8-UQpuS07xkS97SzG91EWCnw)

## ▶ 教員／TA → 学生

- ▶ リアルタイム情報提供
- ▶ 一時提出課題・確認後のコメントなど

## ▶ 学生 → 教員

- ▶ 質問など

## ▶ チャンネルについて

- ▶ #general：教員からの情報提供
- ▶ #質問：学生からの質問

# 課題一時提出について（1 / 3）

- ▶ 最終的なレポート提出のみの確認
  - ▶ 実はプログラムに重大な欠陥があるかもしれない
- ▶ そこで、小課題ごとにプログラムを提出
  - ▶ 授業 / T A が小課題ごとに確認を実施する
- ▶ <https://enpit01.iic.hokudai.ac.jp/nextcloud/s/ArfYZP9Jc52W9Hj>



課題一時提出



課題一時提出 にファイルをアップロード

📁 ファイルを選択するか、ドラッグ&ドロップしてください

# 課題一時提出について（2 / 3）

---

## ▶ 一時提出を求める課題一覧

- ▶ 課題1-1, 課題1-2, 課題2-1, 課題2-2, 課題3-1, 課題4-1, 課題4-2, 課題4-3, 課題5-1, 課題5-2

## ▶ ファイル名

- ▶ 学生番号-**氏名**-課題プログラム名-提出回数.拡張子
- ▶ 例 1 : 02123456-飯田勝吉-ex110-01.html
- ▶ 例 2 : 02123456-飯田勝吉-ex510-01.zip
- ▶ 注：一課題につき複数のファイルを提出する場合は、ZIPでファイル群を一つにまとめること。

## ▶ Zoom/slackの表示名

- ▶ 学生番号-**氏名**
- ▶ 学生番号に基づき一時提出課題を識別
- ▶ 当初、表示名は学生番号だけで十分と思っていたが、他の学生と混同するリスクがあると思い、氏名を追加することとした



# 課題一時提出について（3 / 3）

---

- ▶ 一時提出課題の確認結果について
  - ▶ 確認状況を学生本人と共有する
- ▶ (\*\*\*) 一時提出プログラムの動作確認状況確認サイト
  - <https://docs.google.com/spreadsheets/d/1e1kan3oHI9OhLZPcHdXQM2WwVp7ururDw8FXDLVubKs/edit#gid=0>
    - ▶ ELMSのメールアドレスでGoogle docsにログインすることで閲覧可能
- ▶ 要修正になっていた場合、いずれかの点でエラーがあるので、修正すること
  - ▶ どこが要修正箇所かわからなければ Slackの #質問 で聞くこと
  - ▶ 必要に応じて個別相談を実施

# Zoomの使い方について

---

## ▶ 全体説明

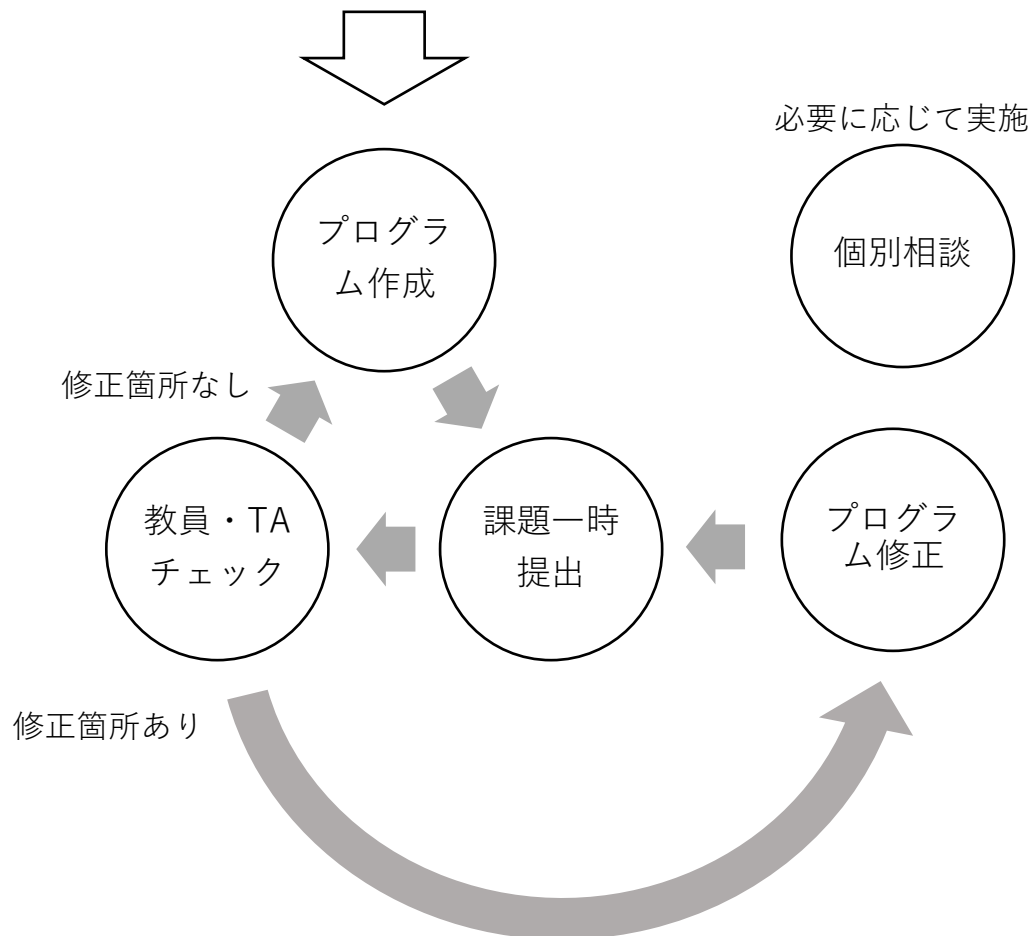
- ▶ 「大部屋」で全員がいる状態で説明

## ▶ 演習時間

- ▶ 通常時：全員、大部屋にいる
- ▶ 教員、TA：
  - ▶ 学生からの質問待機
  - ▶ 一時提出課題がある場合
    - 課題をチェックし、問題がある場合
      - Slackのダイレクトメッセージで指摘
      - 必要があれば、ブレイクアウトルームで学生と議論
- ▶ 学生
  - ▶ 質問があればslackの#質問、に書き込み
  - ▶ 状況により、ブレイクアウトルームに移動しそこで議論

- ▶ 注意：ブレイクアウトルームを行き来するので、Zoomを最新版にアップデートしておくこと

## 講師による事前説明



# ローカルウェブサーバ（1 / 3）

## ▶ 「ローカルウェブサーバ」が必要な課題

### ▶ 課題4-1, 6-1, 6-2



## ▶ それ以外の課題

### ▶ ブラウザからファイルに直接アクセスすればよい



# ローカルウェブサーバ（2 / 3）

## ▶ Windows向けローカルウェブサーバ

### ▶ 04WebServer

### ▶ <http://soft3304.net/04WebServer/>



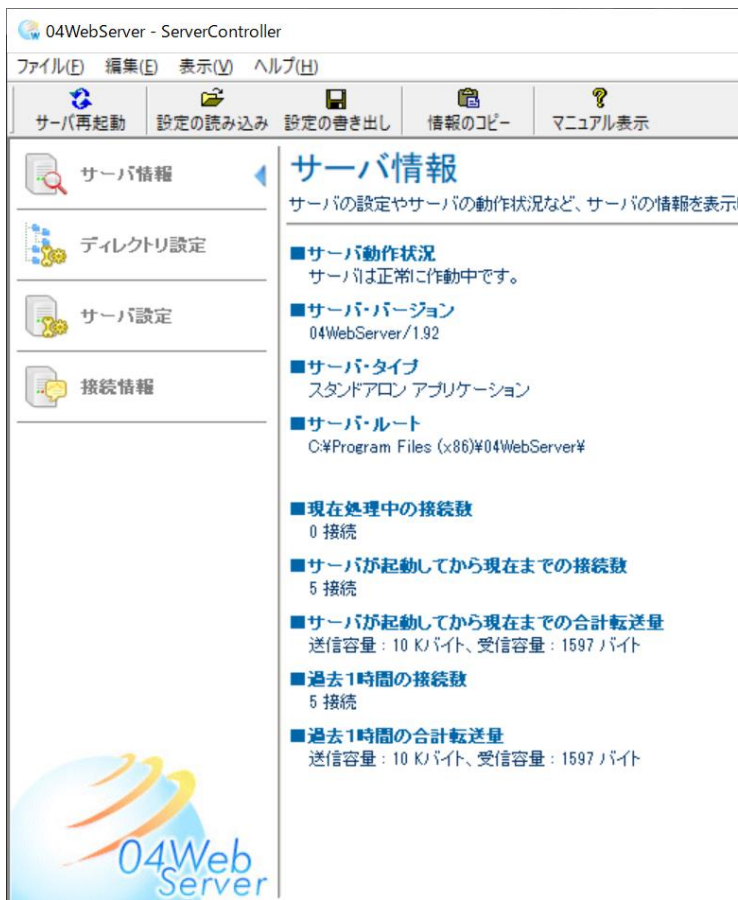
## ▶ MacOSの場合

### ▶ `python -m SimpleHTTPServer 8080`

# ローカルウェブサーバ（3 / 3）

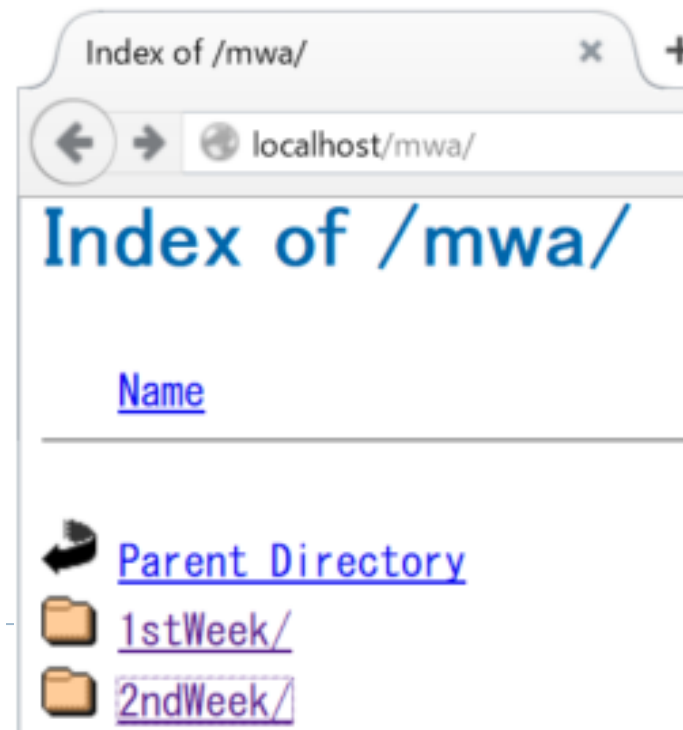
## ▶ 04WebServer

- ▶ 標準インストールでよい



## ▶ ウェブコンテンツ

- ▶ HTML, JavaScriptの配置場所
- ▶ 例: C:\Program Files (x86)\04WebServer\DocumentRoot\mwa



# 課題の補足

---

## ▶ 課題6-3

- ▶ ex420をex421に変更したことにより、課題6-3は継続できなくなった
  - ▶ 課題6-3を廃止する
- ▶ つまり、第6回課題は課題6-1, 6-2 そして課題6-4（発展課題）を実施すること

# レポート

---

- ▶ 全 6 回分をまとめて 1 通のレポートとすること
- ▶ 〆切：11月16日（火） 23：59（JST）
- ▶ 提出方法：ELMS
- ▶ 合格の条件
  - ▶ 指定したすべてのプログラムが課題一時提出で合格すること
  - ▶ レポートを提出し合格すること



# レポート・提出物に関する補足事項

---

- ▶ レポートにキャプチャ画面を含むこと
  - ▶ キャプチャ画面が必要な課題
    - ▶ ex110, ex120, ex220, ex421, ex430, ex510, ex520, ex610, ex620
  - ▶ キャプチャ画面が不要な課題
    - ▶ ex210, ex310, ex410, ex640
- ▶ 意図
  - ▶ 画面装飾が必要な課題はキャプチャ画面をつけてもらう
  - ▶ 画面キャプチャをしてもプログラムができたかどうか、確認できないものはキャプチャ不要