

# ハッカソン 1 (Hackathon 1)

第1回

有本泰子，佐波孝彦，関根晃太

# 本日の内容

- ガイダンス
- プロジェクトの進め方
- プロジェクトテーマの紹介
  - 希望調査の説明
- チーム顔合わせ
  - 提出担当者決めと申告
- Arduino+Processing演習

# ガイドダンス

# この授業はハッカソン1です

- 時間：水曜日6-9限（14時～18時）
- 教室：7号館2階PC演習室2，7階実験室，7302教室
- 担当教員：有本泰子，佐波孝彦，関根晃太
- TA：21名
- 時間外の質問・相談
  - メールをください。
  - 有本：ar@mac-lab.org
  - 佐波：saba@cs.it-chiba.ac.jp
  - 関根：kouta.sekine@p.chibakoudai.jp



学  
習



計  
画



作  
業



報  
告

|      | 全体                                    | ハッカソン1<br>(個人事前課題)        | チーム課題<br>(事後課題) |
|------|---------------------------------------|---------------------------|-----------------|
| 0週目  | チーム分け, 教員割り当て                         |                           | 事前課題提出          |
| 1週目  | 顔合わせ, テーマの詳細・選定,<br>講義・演習 (テーマ共通, 復習) |                           | 演習課題            |
| 2週目  | 講義・演習 (テーマ別)                          | 予習                        | 演習課題            |
| 3週目  | 講義・演習 (テーマ別)                          | 予習                        | 演習課題            |
| 4週目  | 案の1本化. タスク分割と部門分け                     | 案出し                       | 目標設定<br>役割分担の議事 |
| 5週目  | 部門別に技術, 工程, 指標などを設定.<br>チーム全体の計画と照合.  | 部門内で必要な<br>技術, 工程, 指標の案出し | 計画書             |
| 6週目  | プロジェクト計画発表                            | プレゼン準備                    | 計画書の修正          |
| 7週目  | チーム・部門内の進捗報告/課題の共有                    | 個人の進捗報告                   |                 |
| 8週目  | チーム・部門内の進捗報告/課題の共有                    | 個人の進捗報告                   |                 |
| 9週目  | チーム・部門内の進捗報告/課題の共有                    | 個人の進捗報告                   |                 |
| 10週目 | チーム・部門内の進捗報告/課題の共有                    | 個人の進捗報告                   |                 |
| 11週目 | システム評価の会 (動作確認の日)                     | 個人の進捗報告                   |                 |
| 12週目 | 成果発表会                                 | プレゼン準備                    | 成果報告書 (個人)      |
| 13週目 | 報告書の修正                                | 個人成果報告書                   |                 |

# 本授業の概要

本講義では、ハッカソン2と連携した**10名程度**のチーム編成による課題解決型学習(**PBL**)を行う。特に、ハッカソン1では短期間では解決できないプロジェクトのメンバーとなり、

- 1) 情報工学の技術を駆使した課題解決能力
- 2) 割り当てられたサブゴールに対し、課題解決の方法・実行計画の立案・システムの性能評価などを実行する能力
- 3) チーム内の協議やコミュニケーションを円滑に進める能力などを養うことを目的とする。

# 到達目標

- 1) 社会や生活などの実課題に対し、解決方法を立案を作成できる
- 2) 課題解決に向けた計画が立てられる。
- 3) チーム内で連携し、システムの設計や実装ができる。
- 4) チームメンバーと協力し、計画的にプロジェクトの遂行ができる。

# 注意事項

- 履修制限
  - アイディアソンの単位を取得していること
- 注意事項・学習アドバイス
  - ノートPCおよびArduinoを持参すること



# 教科書・参考書

- 教科書：とくになし
- 参考書：適宜紹介する
  - 中嶋秀隆著 改訂7版 PMプロジェクトマネジメント 日本能率協会  
マネジメントセンター
  - ArduinoやM5Stack、Raspberry Piなどに関連する書籍全般。
  - Web開発やIoT、機械学習、統計解析などに関連する書籍全般

# チーム分けとテーマ

- 受講生数: 133名
- 32チーム（1チーム4~5名）
  - チームは既にmanabaで公開済み
- プロジェクトテーマは3つ（後ほど紹介）
  - 希望調査をします
  - 1テーマあたり10~11チーム

# 授業の進め方（第1週～第3週）

- テーマごとに分かれて演習
  - 演習課題の提出
- 授業
  - 前回演習課題解説（15～20分）
  - 新しい課題に関する説明（30～50分）
  - 演習課題を解く（30～50分）
    - 学習した内容をもとに課題を解いたり，プログラムを作成する
- 事後学習
  - 演習課題に取り組み，提出する

# 授業の進め方

## (第4週～第5週, 第7週～第11週)

- 事前学習
  - manabaのプロジェクト内チームスレッドでの意見交換
  - 個人作業の進捗状況を報告書にまとめて提出
  - 代表者がチームメンバーの報告書をチームとしての報告書にまとめて提出
- 授業
  - TAによる進捗状況の確認 (10~15分)
  - 各自に割り当てられた作業を進める
- 事後学習
  - 授業当日までの作業内容を進める
  - manabaのプロジェクト内チームスレッドでの意見交換

# manabaのチームスレッドによる議論

📄 テストプロジェクト1 [課題に関する説明]

Bチーム : チームスレッド

課題に関する説明

テストプロジェクト1です。

受付期間 2025-02-28 19:05:00～2025-03-31 23:55:00

提出方法 チームで提出

閲覧設定 自分が所属しているチームのみ閲覧できる

提出物 課題を提出したチーム同士で閲覧・コメント可

学生による再提出の許可 再提出を許可する

🗨️ コメントを書く

コメントの表示

🔄 更新順

👤 ツリー

1

Bチームしか見えないはずのコメントです。

👤

テストB

佐波 孝彦

🗑️

2025-02-28 19:19:36

🔄 レス

2

テスト

👤

テストです。

木澤 妃名子

🗑️

2025-02-28 19:36:42

🔄 レス

表示モード

👤 教員

👥 学生

このチームは既に課題を提出しています。

➤ 提出物を確認

🔄 チームリスト

▶ Aチーム

▶ Bチーム

▶ Cチーム

▶ Dチーム

▶ グループA

👥 Bチームのメンバー ➤ アクセス状況

▶ 坂井 祐太

▶

▶

意義のあるコメントの数を点数化  
積極的に議論をしてください

ハッカソン1

13

# 進捗報告書の提出（個人・チーム）

- 進捗報告書（個人）の提出
  - 満点：3点
  - 提出締切：授業前日午前9時
  - A4用紙1枚程度
- 進捗報告書（チーム）の提出
  - 満点：3点
  - 提出締切：授業当日午前9時
  - 提出係りが個人の進捗報告書をまとめて作成
- 提出内容により減点あり
- 締切厳守
  - 締切り時間以降の提出は減点
  - やむを得ない事情がある場合は、担当教員に別途相談すること

# 出席

- 出席システムで出席登録を行うこと
- 教員が呈示するQRコードで出席登録
- すべての授業に出席することが前提
- 病気や怪我などでやむを得ず欠席した場合は、欠席届とともに欠席理由を証明する書類を教員に提出すること

# 成績評価

- チーム内の議事録/進捗報告/議事録などの報告書、最終報告書、プレゼンテーションによって評価する
- 合格者について以下の割合で評価
  - チーム・個人活動
    - 演習課題（6%，3回）
    - 各回の進捗報告書・議論（個人・チーム）（14%，7回）
  - プレゼンテーション
    - プロジェクト計画発表（15%）
    - 成果報告発表（25%）
  - 提出物
    - プロジェクト計画書(15%，1回)
    - 成果報告書(25%，1回)
- 評価合計が60%以上を合格とする



# ハッカソン1の三つの目的

1. 能動的に「自ら学ぶ」自分なりの方法を身につける
2. チームワークとコミュニケーションを体験する
3. 報告書を書き，プレゼンテーションをする経験を積む

詳しくはテーマ共通にある「ハッカソンのテキスト」を参照のこと

# プロジェクトの進め方

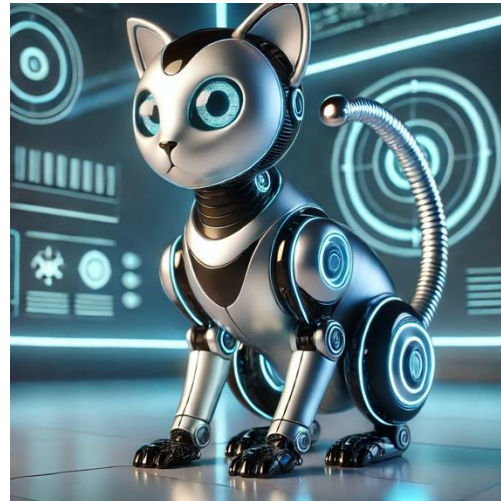
# 課題に対する解決案を考える

どんな猫型ロボットを作りたいかを決めて、  
必要な作業を決定する

ネコ型ロボットを作りたい

移動する？

触覚をつける？

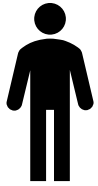


視覚をつける？

鳴く？しゃべる？

# 作業を分担する

## 作業担当者を決める



歩かせる

ネコ型ロボットを作rinaさい



カメラをつける



センサーをつける



しゃべらせる

# 作業スケジュールを決める

割り当てられた作業の課題の細分化し、  
いつまでに終わればよいかスケジュール化

## 猫型ロボットを作る

チーム99

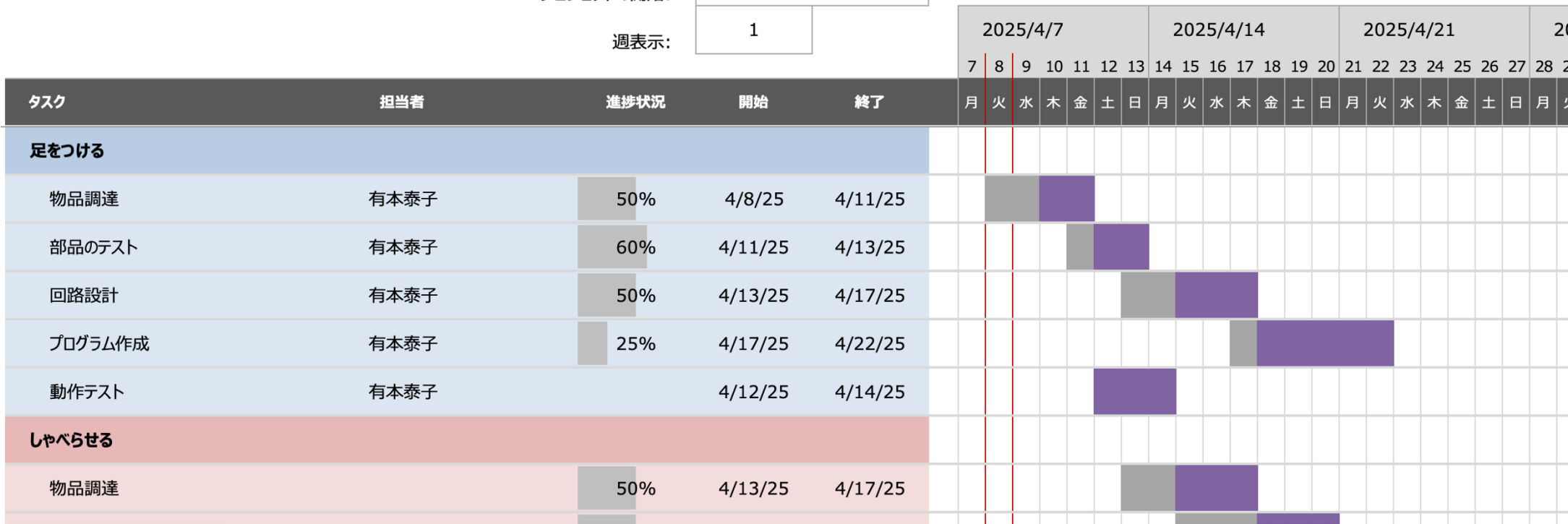
プロジェクト代表者：有本

プロジェクトの開始:

火, 2025/4/8

週表示:

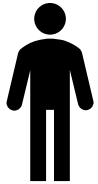
1



# 必要な機材の調達

各作業に必要な機材を調達する

サーボモータ



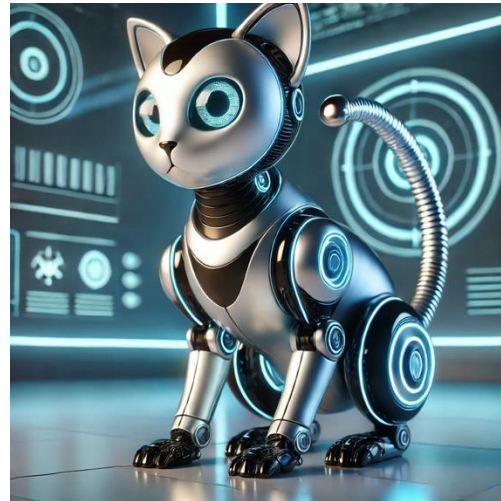
歩かせる

超音波センサー



センサーをつける

ネコ型ロボットを作rinaさい



カメラモジュール



カメラをつける

スピーカー

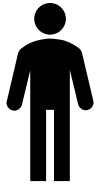


しゃべらせる

# 実装・テスト

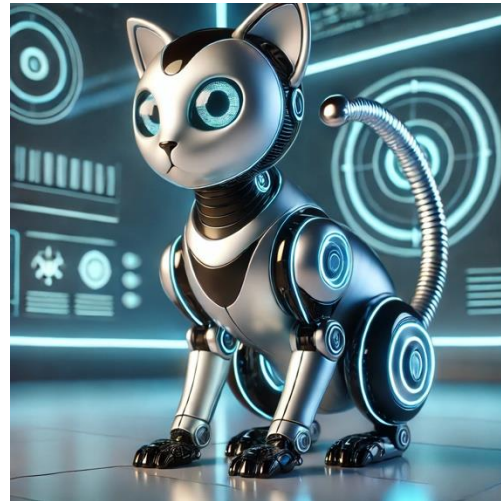
## 各担当箇所の実装とテスト

サーボモータ



歩かせる

ネコ型ロボットを作rinaさい



カメラモジュール



カメラをつける

超音波センサー



センサーをつける

スピーカー



しゃべらせる

# システム結合・テスト

作成した各パーツを結合してテスト





# プロジェクトテーマの紹介

# 3つのテーマから選択

- シンセサイザー（楽器）を作成する
  - デジタル伝言ゲーム
  - IoTスイッチを作ろう
- 
- arduino利用に関する制限
    - 最低でも人数分-1台を利用
    - arduinoの中で処理が完結するように指示する
      - PC側で処理しない

# 希望調査

- プロジェクトでどのテーマにするかディスカッション
- 希望通りのテーマになるとは限らない
- チーム単位でmanabaのプロジェクトから提出
  - 希望順位をつけてください
- 締め切り：4月15日午前9時

# チーム顔合わせ

# 自己紹介

- 名前
- 得意なこと
- 苦手なこと
- チームの提出物を提出する係を  
選定
  - フォームから提出係を申告
  - 提出物：毎週の進捗報告書，計画書，  
成果報告書など

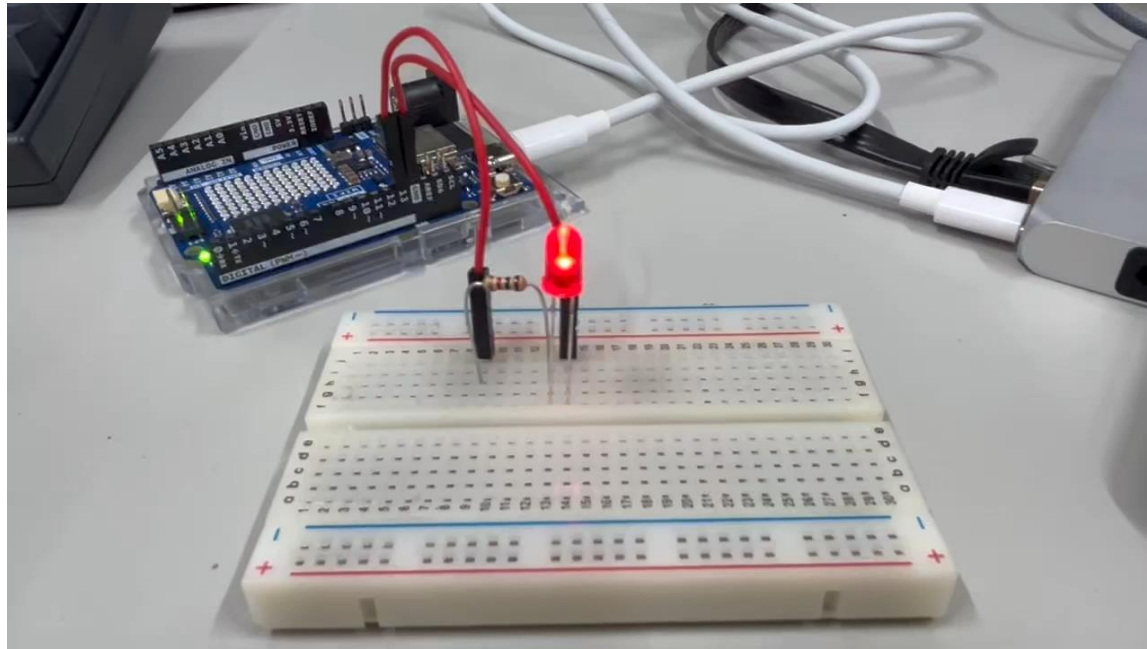


<https://forms.gle/4VganVbvXauw42xs5>

# Arduino+Processing

# 復習問題：簡単なLED点滅

- ブレッドボードにLEDをひとつ配置し，以下の設定で点滅させなさい
  - 点灯：1秒
  - 消灯：0.5秒



# ArduinoからProcessingへデータ転送

- Arduino側：データをSerial portへ出力するプログラムを書く

```
void setup() {  
  // シリアルポートを開く。115200pbsでデータを送信する。  
  Serial.begin(115200);  
}  
  
void loop() {  
  Serial.write(100); // シリアルポートに値を出力  
  delay(1000); // 1秒待つ  
  Serial.write(200); // シリアルポートに値を出力  
  delay(1000); // 1秒待つ  
}
```



# ArduinoからProcessingへデータ転送

- Processing側：データをポートから取得するプログラム

```
import processing.serial.*;

Serial port;// シリアルポート
int x;      // x座標用

void setup(){
  // ウィンドウサイズ
  size(256, 256);
  // ポートを初期化
  port = new Serial(this, "COM3",115200);
  // シリアルポートの初期化
  port.clear();
}
```

(となりに続く)

接続するポート名  
人によって異なる

```
(つづき)
void draw(){
  // 背景色は白
  background(255);
  // 座標xに50*50の円を描く
  ellipse(x,100,50,50);
}

// データが送信されてきたら呼び出される
void serialEvent(Serial p){
  // ポートからデータを取得
  x = p.read();
  // 書き出して値を確認
  println(x);
}
```

# Arduino側で使用する関数

- 一つの値を送信
  - `Serial.write(val)`
  - `val`: 1バイト分のデータ
- 複数の値を送信
  - `Serial.write(str)`
  - `Serial.write(buf, len)`
  - `str`: 文字列
  - `buf`: `byte`型配列
  - `len`: 配列長

いずれもバイト単位でバイナリとして送信される

<https://docs.arduino.cc/language-reference/en/functions/communication/serial/write/>

# Processing側で使用するクラス

- **Serial**

- シリアル通信を実現するクラス
- コンストラクタ
  - Serial(parent)
  - Serial(parent, baudRate)
  - Serial(parent, portName)
  - Serial(parent, portName, baudRate)
- parent (PApplet): "this"と書く
- baudRate (int): 通信速度 (9600がデフォルト)
- portName (String): ポート名 (デフォルトはCOM1)

# Processing側で使用するメソッド

- データを受信するメソッド
  - **read()**
    - バッファ内にある次の値を0~255の値で返す
    - 何も値がない時は-1を返す
    - 戻り値の型はint
- データを受信した時に呼び出される関数
  - **serialEvent(Serial p)**
    - この関数内でread()メソッドを使ってデータを取得する
    - この関数を使うと、何も値がない時はread()関数を呼び出さないなので-1を取得することがない

# 提出課題

# 演習課題提出

- 作成したスケッチファイルを提出
  - 提出課題1：HCK01\_01.ino
  - 提出課題2：HCK01\_02.ino, HCK01\_02.pde
  - 発展課題：HCK01\_03.ino
- 提出課題 1 及び2の提出は必須
- 提出期限：4月15日午前9時