

第15回全国高等学校情報教育研究会全国大会

# 情報Ⅰを見据えたプログラミング教育

～学習者の独自設計を可能にさせたミニチュア配膳ロボットの  
プログラミング教材の開発と実践と評価～

佼成学園中学校高等学校  
岡野 英樹

◆▶ 楽しく面白い授業を目指したいが・・・

学習指導要領

生徒の能力

共通テスト

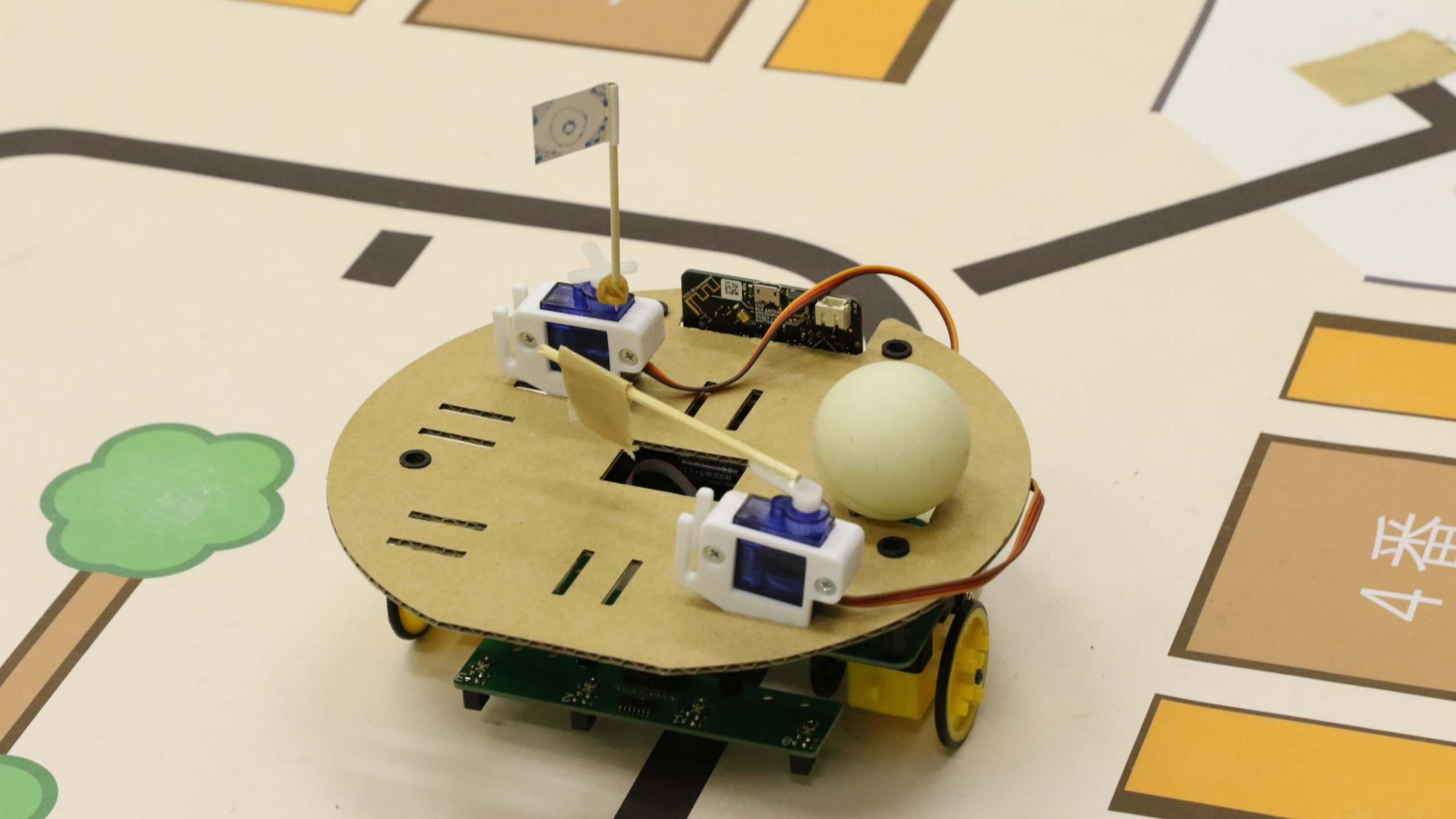
資金&設備

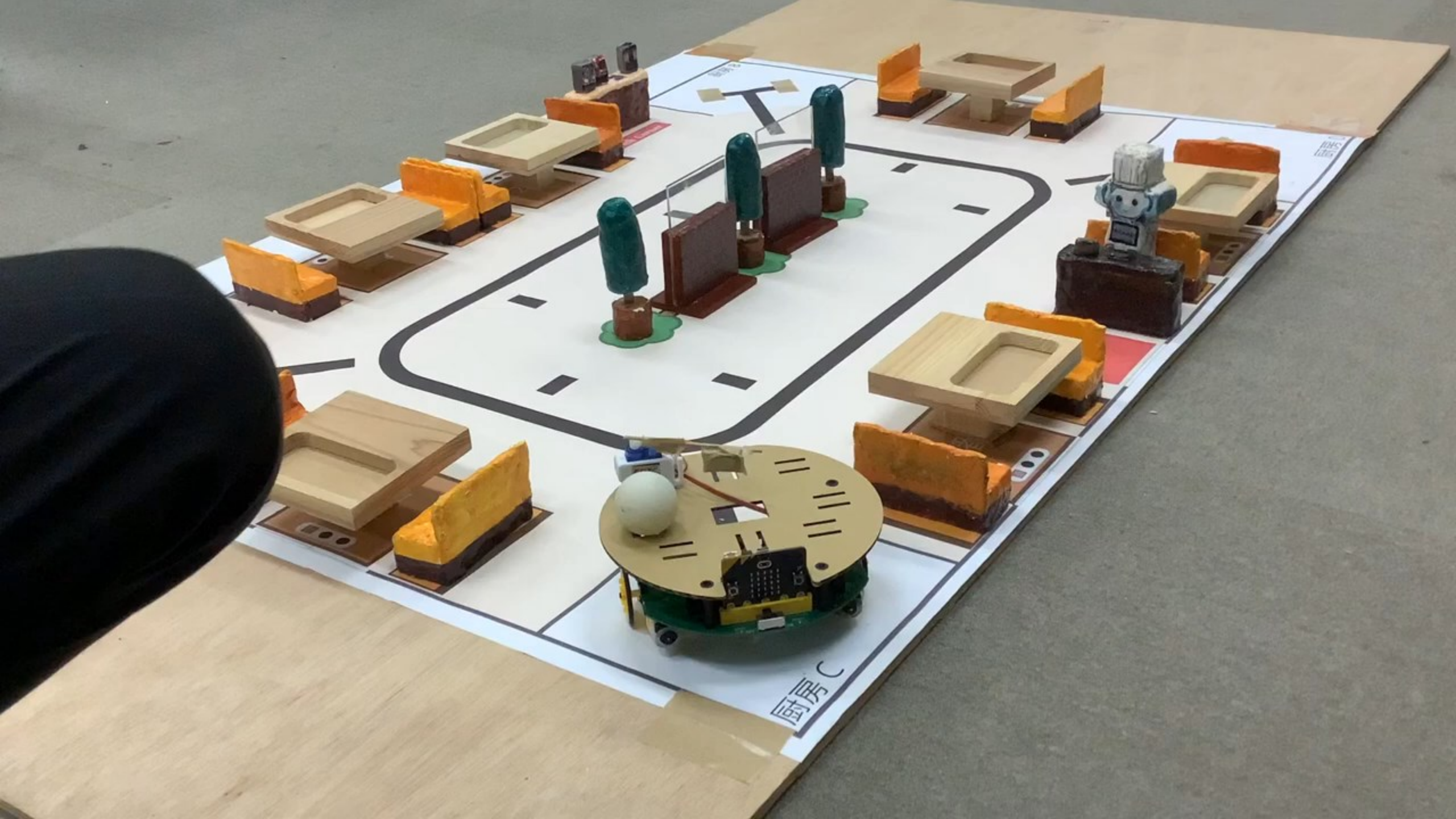


# モチベーションを支える三要素

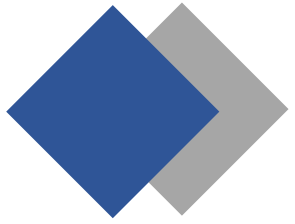


- 1 ハードウェアを動かすことができる。
- 2 実社会と結びつきがある。
- 3 自己表現が可能である。

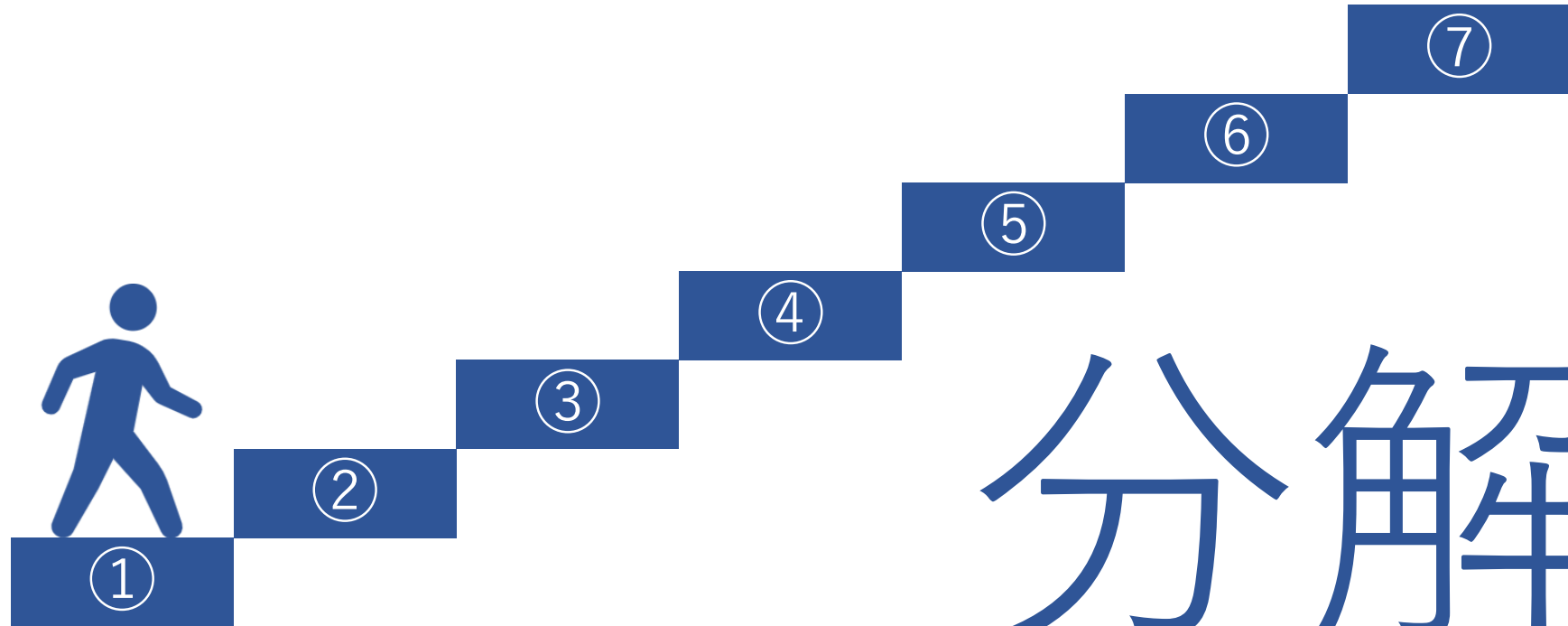




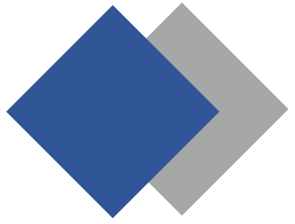
厨房 C



# プログラミング指導する上での考え方



# 分解



# プログラミング指導のポイント

1

生徒の課題の速度を上げる方法

2

質問やトラブル対応に時間をとられない方法

3

生徒間にスキルの差があり進捗がバラバラになる問題



# 昨年度の年間カリキュラム

1学期中間

(座) 教科書の内容 + (実) タイピング

スキル醸成

1学期期末

(座) 教科書の内容 + (実) HTML&CSS

コード記述  
の練習

2学期中間

(実) プログラミング基礎

2学期期末

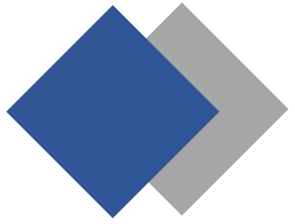
(実) 配膳ロボ (問題解決) + (実) プレゼンテーション

3学期学年末

(実) シミュレーション + (実) 相関分析, 回帰分析



# タイピングとHTML&CSS

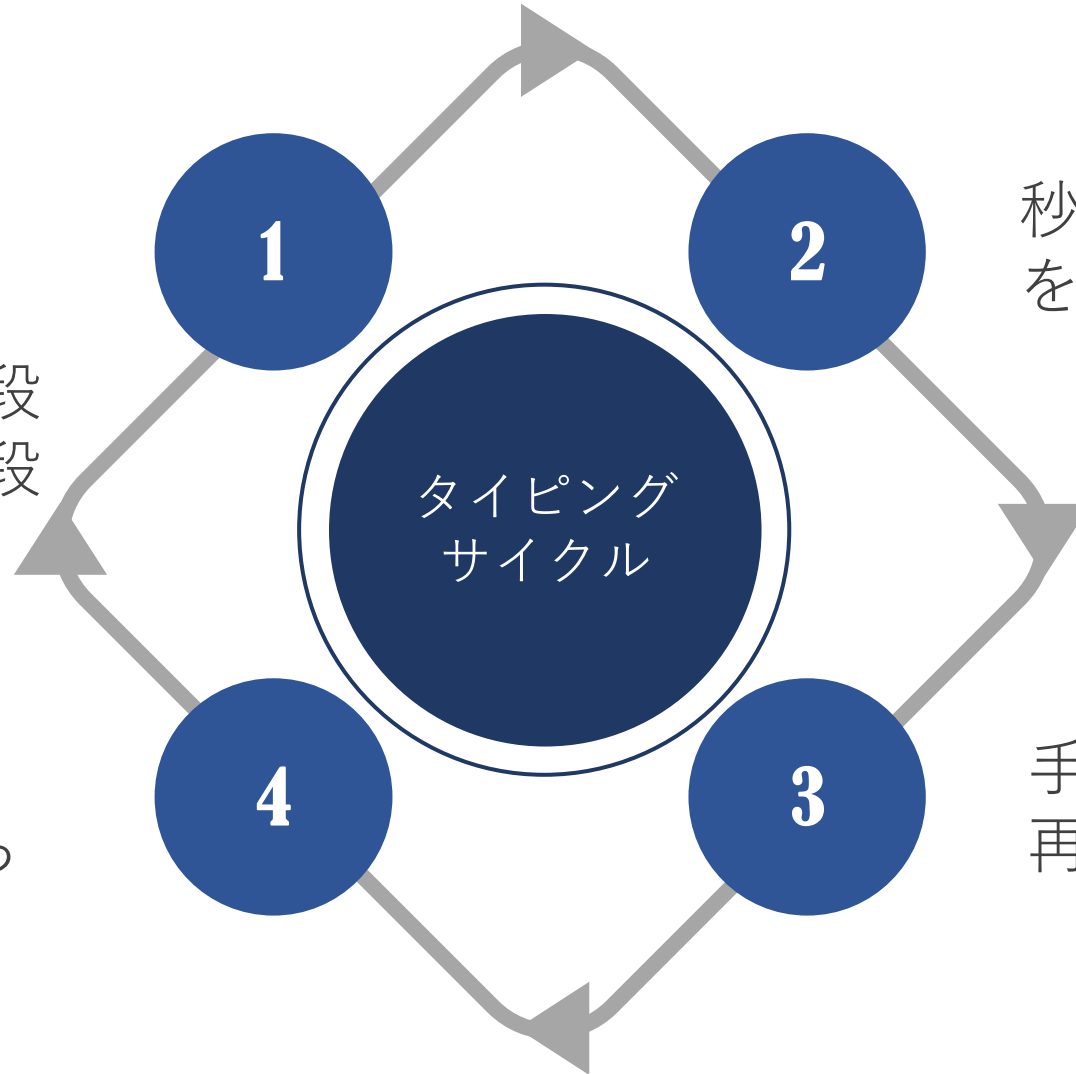


# タイピング

マイタイピング(web)で各自タイピングを行う。

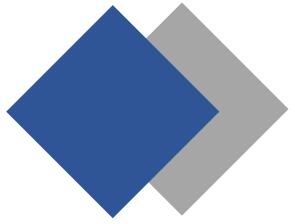
- 1周目 中段
- 2周目 中段 + 上段
- 3周目 中段 + 上段 + 下段
- 4周目 中段 + 上段 + 下段 + 最上段

手を隠して秒間2タイプ  
(2000点) 以上打てたら合格



秒間2タイプ (2000点)  
を超えたら教員に報告

手を隠す板を上に置き  
再度テスト



# HTML&CSS

```
index.html
1 <!DOCTYPE html>
2 <html>
3 <head>
4   <title></title>
5   <style type="text/css">
6     |
7   </style>
8 </head>
9 <body>
10
11 </body>
12 </html>
```

1

タイピング実践

2

正確に記述しなくても動く

3

エディタの基本操作を学習

<html enter enter <style enter

14タイプ

# プログラミングと問題解決

# 2学期 プログラミングと問題解決

## Python基礎学習(10コマ)

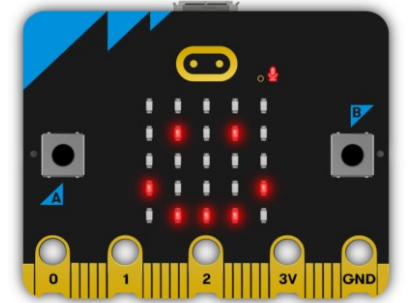
プログラミングの基礎を学習

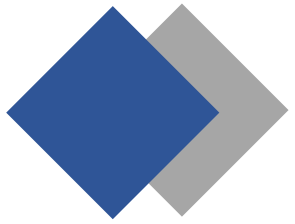
## 配膳ロボット基本動作(5コマ)

大きなプログラムを書く際の順序立てなどを学習

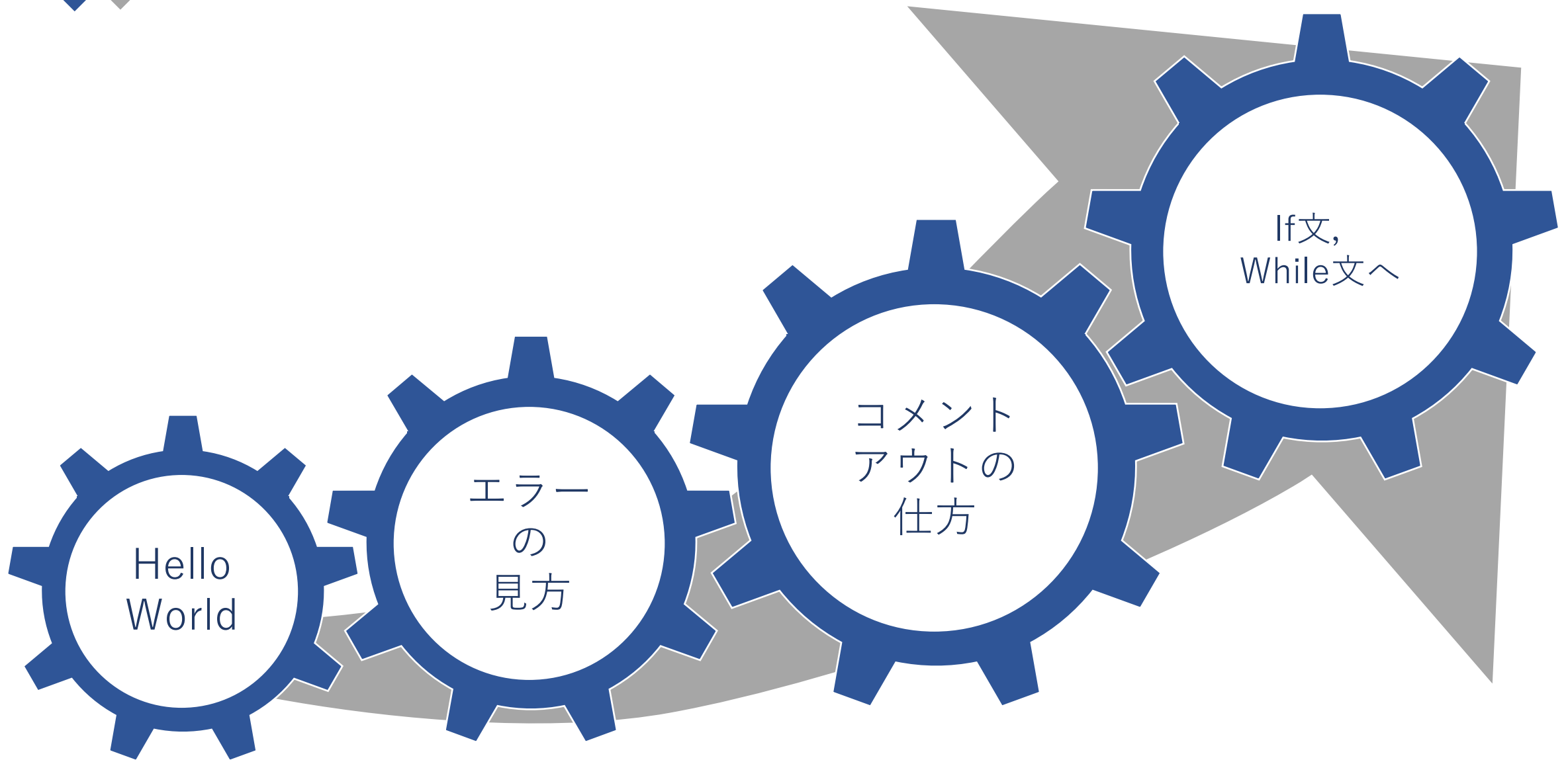
## 配膳ロボットカスタマイズ(4コマ)

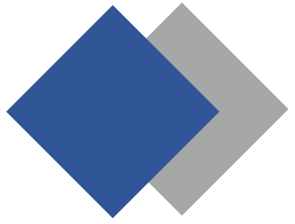
自分達で飲食業界の問題を発見し解決してプレゼンテーション





# Python基礎学習の戦略





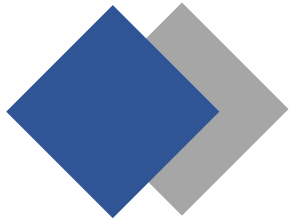
# エラーの見方

```
1 # Add your Python code here. E.g.  
2 from microbit import *  
3  
4  
5 while True:  
6     display.scroll('Hello, World!')  
7     display.show(Image.HEART)  
8     sleep(2000)  
9  
10
```

正しい

```
1 # Add your Python code here. E.g.  
2 from microbit import *  
3  
4  
5 while True:  
6     display.croll('Hello, World!')  
7     display.show(Image.HEART)  
8     sleep(2000)  
9  
10
```

間違っている



# エラーの見方（プリント）

## 2. 間違い探し

課題 2-1★ 配布されたファイルをテキストエディタに張り付けて、どこが違うかを探し、正しい答えを書いてみよう。（このプリントの写真提出）

**AttributeError（属性エラー）…ドット（.）の後の単語が間違っているところなる可能性大**

---

6

行目の

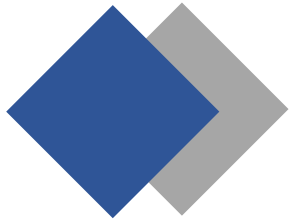
croll

が間違っていて

scroll

が正しい





# エラーの種類

Attribute Error

ドット (.) の後の単語が間違っている可能性がある

Syntax Error

特殊記号が間違っている可能性がある

Name Error

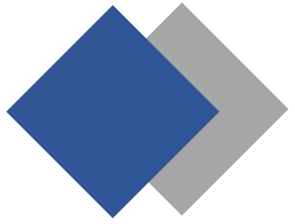
単語やドット (.) の前の単語が間違っている可能性がある

Indentation Error

タブやスペースが間違っている可能性がある

Import Error

モジュールの指定が間違っている可能性がある



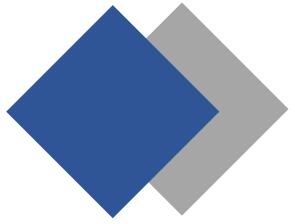
## エラー対応

「何行目の何エラーが出たか」を  
報告したら質問対応するというルール

エラーの内容把握

エラーの解決

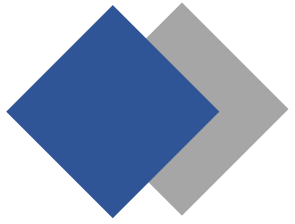
対応する時間を極力削減



## コメントアウト

初心者は間違ったコードを書いた際に、  
書いたコードを消そうとする。

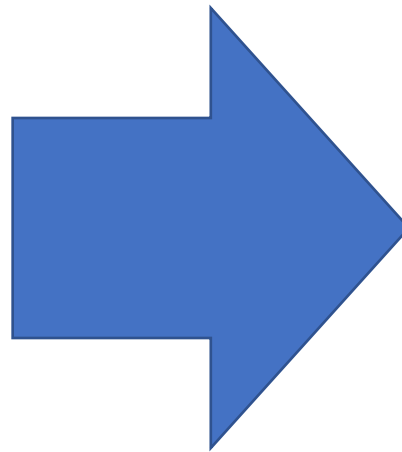
コメントを利用することで、無効化させ、  
先に進んだり前に戻ったりしやすくさせる。



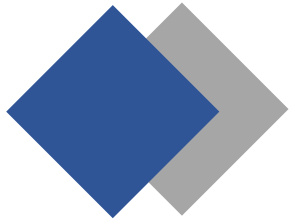
# コメントアウト

```
1 # Add your Python code here. E.g.
2 from microbit import *
3
4
5 a = 53
6 b = 59
7 c = 67
8 d = 4661
9 e = "13"
10 f = "40"
11 g = "0"
12
13 ans = a + b
14 ans = a - c
15 ans = b * c
16 ans = d / b
17 ans = a + e
18 ans = e + f
19 ans = f - g
20
21 while True:
22     display.scroll(ans)
23
```

Ctrl + /



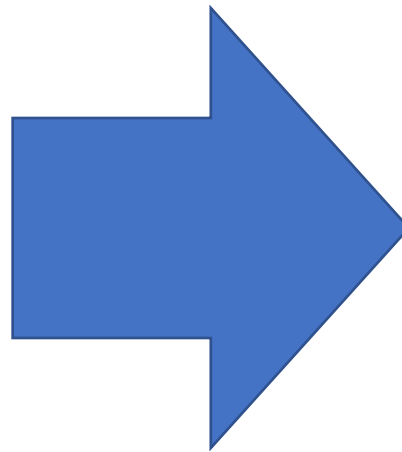
```
1 # Add your Python code here. E.g.
2 from microbit import *
3
4
5 a = 53
6 b = 59
7 c = 67
8 d = 4661
9 e = "13"
10 f = "40"
11 g = "0"
12
13 # ans = a + b
14 # ans = a - c
15 # ans = b * c
16 # ans = d / b
17 # ans = a + e
18 # ans = e + f
19 # ans = f - g
20
21 while True:
22     display.scroll(ans)
23
```



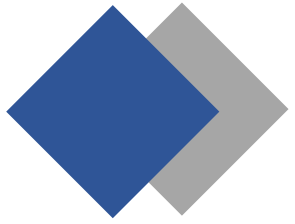
# アンコメント

```
1 # Add your Python code here. E.g.
2 from microbit import *
3
4
5 a = 53
6 b = 59
7 c = 67
8 d = 4661
9 e = "13"
10 f = "40"
11 g = "0"
12
13 # ans = a + b
14 # ans = a - c
15 # ans = b * c
16 # ans = d / b
17 # ans = a + e
18 # ans = e + f
19 # ans = f - g
20
21 while True:
22     display.scroll(ans)
23
```

Ctrl + /



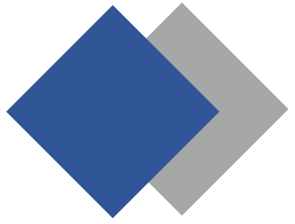
```
1 # Add your Python code here. E.g.
2 from microbit import *
3
4
5 a = 53
6 b = 59
7 c = 67
8 d = 4661
9 e = "13"
10 f = "40"
11 g = "0"
12
13 ans = a + b
14 # ans = a - c
15 # ans = b * c
16 # ans = d / b
17 # ans = a + e
18 # ans = e + f
19 # ans = f - g
20
21 while True:
22     display.scroll(ans)
23
```



# その後の学習内容

|            | No1 | No2 | No3 | No4 | No5 | No6 | No7 | No8 | No9 | No10 |
|------------|-----|-----|-----|-----|-----|-----|-----|-----|-----|------|
| 実行         | ■   | ■   | ■   | ■   | ■   | ■   | ■   | ■   | ■   | ■    |
| エラー内容      |     | ■   | ■   | ■   | ■   | ■   | ■   | ■   | ■   | ■    |
| コメントアウト    |     |     | ■   | ■   | ■   | ■   |     | ■   | ■   |      |
| 変数         |     |     |     | ■   | ■   | ■   | ■   | ■   | ■   | ■    |
| 疑似乱数       |     |     |     |     | ■   | ■   |     | ■   |     |      |
| IF文        |     |     |     |     | ■   | ■   | ■   | ■   | ■   | ■    |
| 初期化とループ    |     |     |     |     |     |     | ■   | ■   | ■   | ■    |
| センサーの利用    |     |     |     |     |     |     |     | ■   | ■   | ■    |
| アクチュエータの利用 |     |     |     |     |     |     |     |     | ■   | ■    |
| 関数         |     |     |     |     |     |     |     |     |     | ■    |

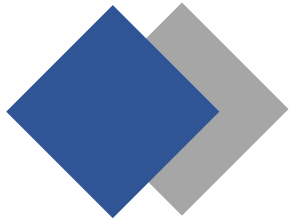
前の時間の内容を復習してから次の内容へ



# スキルの差の吸収

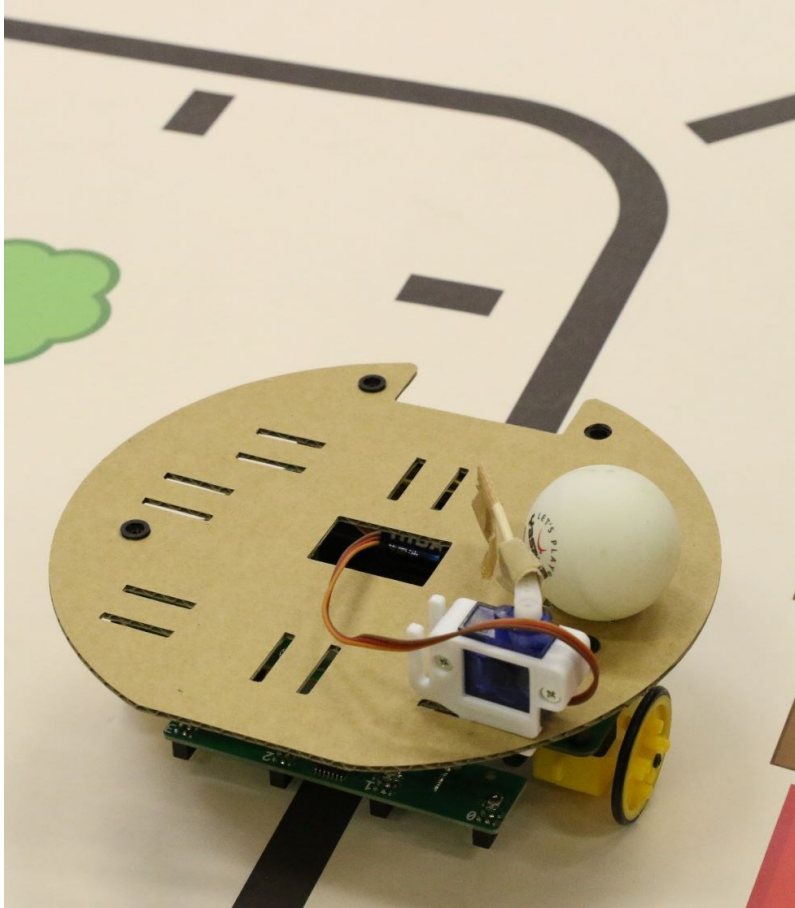
生徒間のスキルの差を吸収する 3つのレベル別問題

- ★ まるうつし（写経レベル）の問題
- ★★ 自分で少し考える問題
- ★★★ 深く考えたり，Webで調べたりしながら解く問題



# 配膳ロボ

## 基本編



1

より実践的な教材で近未来をイメージ

2

プログラミング的思考力の分解

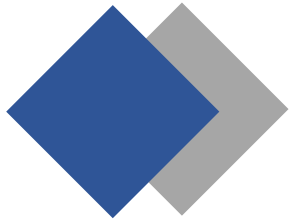
3

目標を小さく分割し，小目標のクリアを繰り返していく

4

クリアしていくことでモチベーションをアップさせる





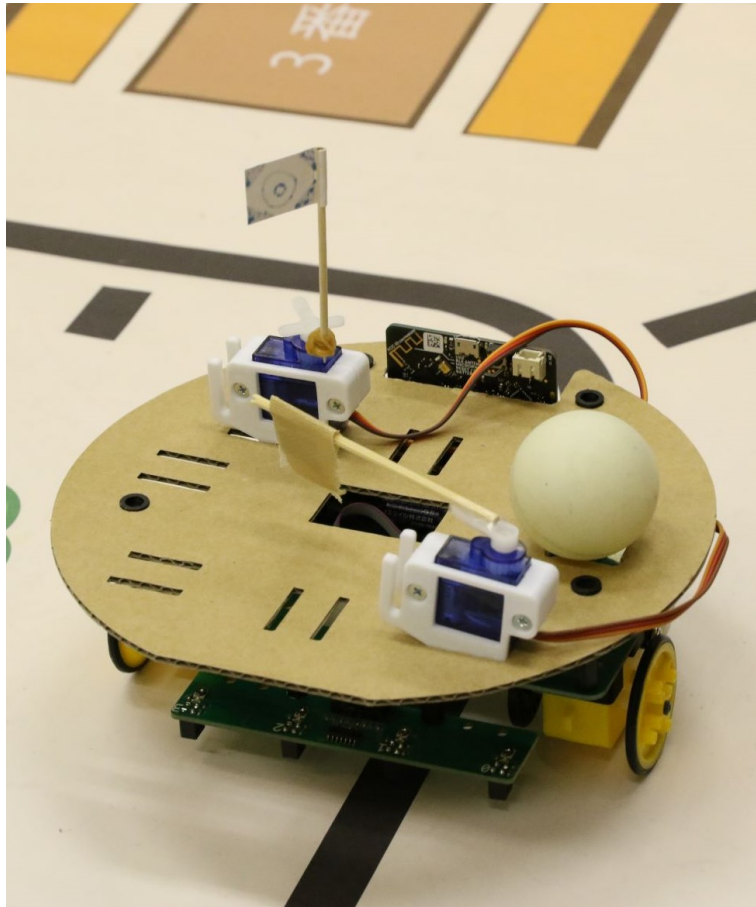
# 配膳ロボ 基本編 11のSTEP

- STEP1. コードはコメントアウトを使い行為を分解しておく。
- STEP2. モジュールを読み込み、まっすぐ走る。
- STEP3. 環状線をライントレースする。
- STEP4. テーブル番号のセットをする。
- STEP5. テーブルセットのプログラムをコメントアウトし、配膳テーブル番号を4（番目）にしておく。
- STEP6. テーブルをカウントするプログラムを書く。
- STEP7. 指定したテーブルで停まり、配膳する。
- STEP8. キッチンをカウントするプログラムと定数の宣言。
- STEP9. 環状線からスタートし、自分のキッチンに戻る。
- STEP10. 自分のキッチンからスタートし、T字路で左折し、4番目のテーブルで配膳し、自分のキッチンに戻る。
- STEP11. テーブル番号のプログラムをアンコメントし、T字路にもどって半回転させるプログラムを書いて完成。

目標に準拠した評価

# 配膳ロボ

## カスタマイズ編



1

180度サーボモータ，低速モータ，液晶ディスプレイ，LEDのうちどれか一つを利用

2

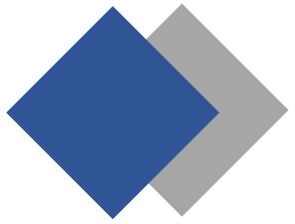
自分達で表現方法を工夫

3

配膳ロボのベンチャー企業の社長，副社長になったつもりで配膳ロボを製作

4

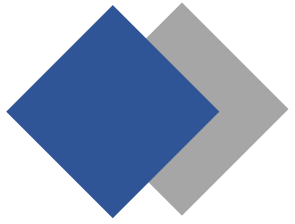
できた配膳ロボをプレゼンテーションし，iPadで動画をとって提出



# 配膳ロボ カスタマイズ編の評価

|                     | 8                                | 6                                    | 4                                   | 2                      |
|---------------------|----------------------------------|--------------------------------------|-------------------------------------|------------------------|
| プレゼン声<br>(バトス, エトス) | 聞こえやすい声で、抑揚をつけながら話している           | 聞こえやすい声で話している。                       | 声が小さく、聞き取りにくい。                      | 声が小さく、聞こえない。           |
| 説明のわかりやすさ<br>(ロゴス)  | 説明が最初から最後まで首尾一貫している。             | 説明が最初から最後まで首尾一貫しているが、一部論理性にかけるものがある。 | 説明が曖昧でよくわからない部分が残る。                 | 説明が支離滅裂でよくわからない。       |
| データ, グラフなど<br>(ロゴス) | 自分で持ってきたグラフをわかりやすく編集し、効果的に使っている。 | 与えられたグラフを効果的に使っている。                  | 自分で持ってきたグラフや与えられたグラフを使っているが効果的ではない。 | グラフを使っていない。            |
| 問題発見・意識<br>(ロゴス)    | 様々な社会課題を発見し、問題意識を明確にしている。        | 様々な社会課題を発見しているが、問題意識が少し曖昧である。        | 社会課題を発見できていないが、問題意識はある。             | 社会課題を発見できておらず、問題意識もない。 |
| デモ動画                | デモ動画があり、適切                       | デモ動画があるが適                            | デモ動画があるが動                           | デモ動画がない。               |

ルールブック評価



# プログラミング指導のポイントまとめ

1

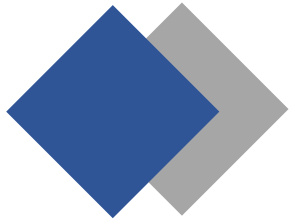
生徒の課題の速度を上げる方法  
→タイピングを行いスキル上達

2

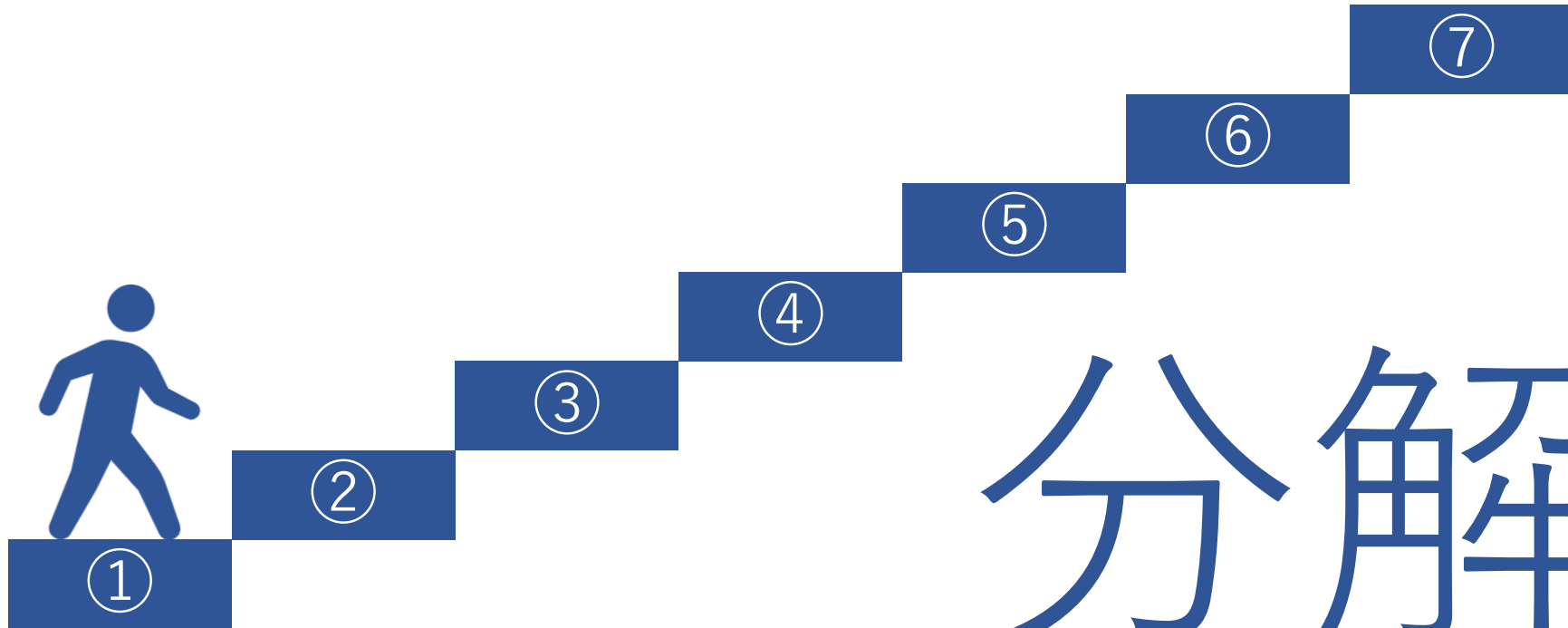
質問やトラブル対応に時間をとられない方法  
→エラーの種類を教え，授業支援ソフトやルールを作り対

3

応。生徒間にスキルの差があり進捗がバラバラになる問題  
→レベル別問題を作り，学習者の能力に合わせていく。



# プログラミング指導する上での考え方



# 分解



ご清聴ありがとうございました。

こうせい

# 佼成 プログラミング教育

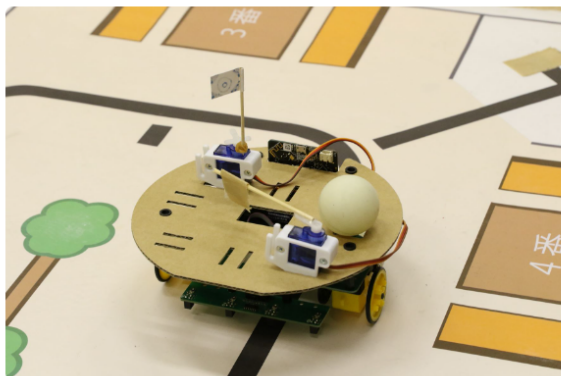
検索

佼成学園中学校・高等学校 教員研究ポートフォリオ

佼成学園ホームページへ

## 情報Ⅰを見据えたプログラミング教育～学習者の独自設計を可能にさせたミニチュア配膳ロボットのプログラミング教材の開発と実践と評価～

2021年度 東京都私学財団助成  
公益法人東京都私学財団賞受賞  
情報科 岡野英樹  
共同研究者 情報科 萩原知明



令和4年度から新課程「情報Ⅰ」が必修で始まる。その中で必修化となったプログラミングの内容の研究授業。扱う題材は、近未来に世の中に普及する可能性が高い配膳ロボット（以下、配膳ロボ）。配膳ロボのミニチュア版を製作し、取り組みを実践しその内容を分析、考察していく。タイピングを学習し、Pythonの基礎知識を教えた後、チームで配膳ロボのプログラミングに挑み、一通り配膳ロボが機能したら、最後はチームでその配膳ロボをカスタマイズ（再構成）させ、自分達のオリジナルの配膳ロボを作りプレゼンテーションを行うというものである。

報告書を読む

スライドを見る