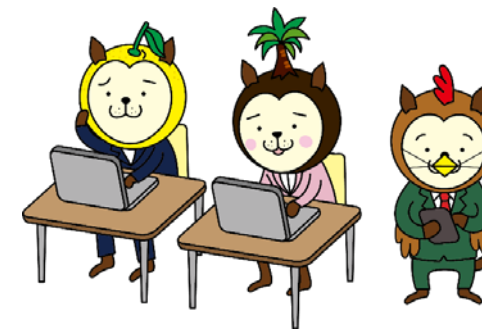


# D 情報の技術



## Scratch3.0を用いた 「お掃除ロボ」シミュレーション教材

---

宮崎県教育研修センター

# 取り扱い内容項目

(1) 生活や社会を支える情報の技術について調べる活動

## 学習指導のポイント

- ・ 生活や社会を支える情報技術に対する興味・関心の喚起
- ・ 生活や社会を支える情報技術の役割や影響に関する社会的な認識の形成
- ・ 生活や社会を支える情報技術の仕組み、原理・法則に関する科学的な理解
- ・ 情報に関する技術の見方・考え方への気づき

# 授業の進め方 (シミュレーション教材の利用分のみ)

---

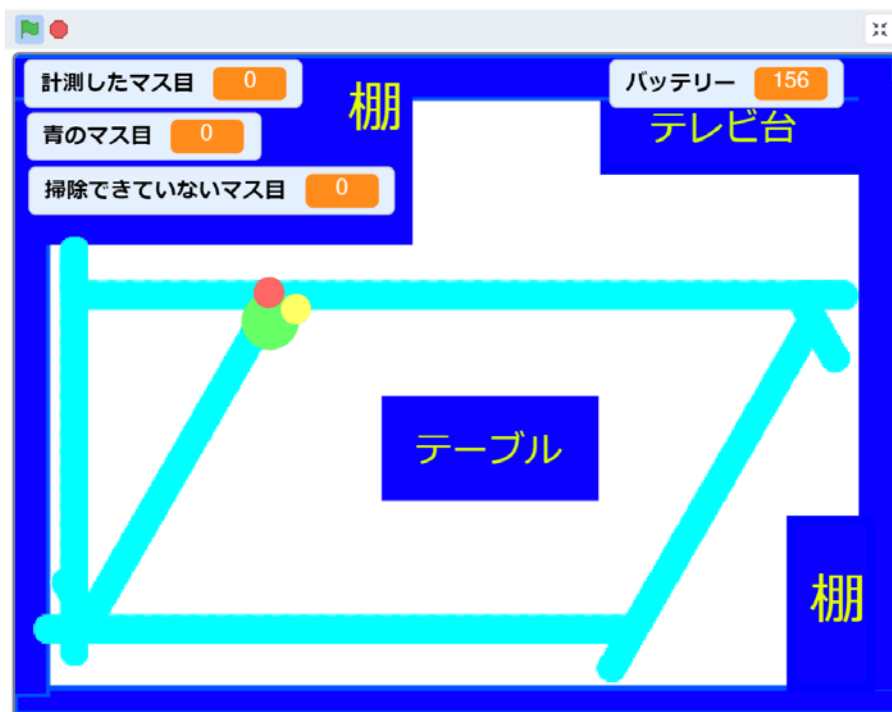
授業時数: 4時間程度

- ① 「お掃除ロボ」シミュレーション教材を体験する
- ② 市場で売られている「お掃除ロボ」の規格や特長を調べる
- ③ シミュレーション教材を活用して改善する  
～ 制約条件に基づいて改善を行う ～

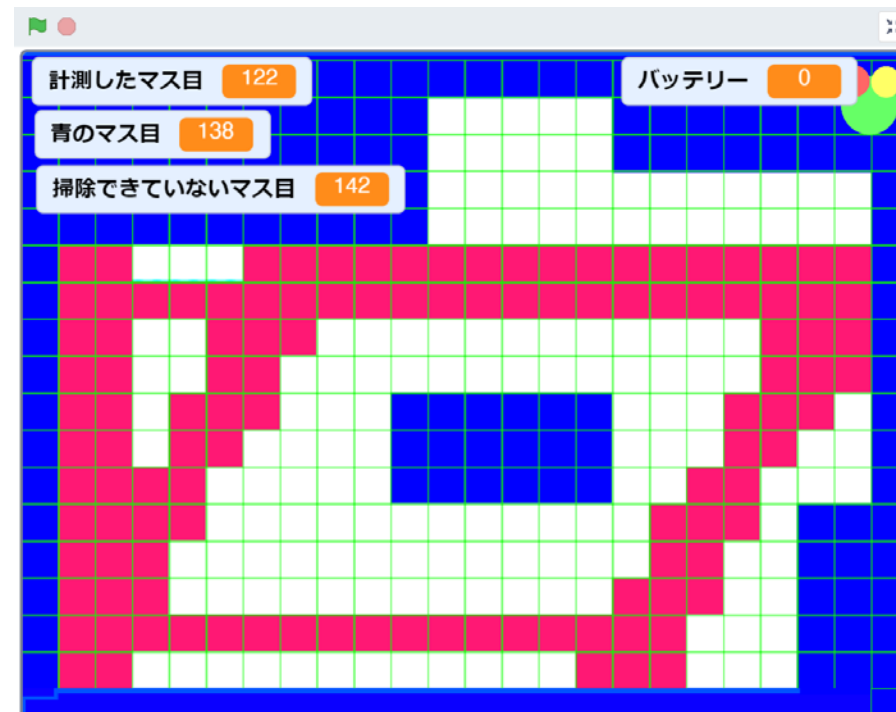
# 授業のねらい(シミュレーション教材の利用分のみ)

- ① 「お掃除ロボ」シミュレーション教材を体験する  
→ D(1)イ 問題解決の工夫について触れながら体感する
- ② 市場で売られている「お掃除ロボ」の規格や特長を調べる  
→ D(1)イ 問題解決の工夫について調べる中で、技術の見方・考え方に気付かせる
- ③ シミュレーション教材を活用して改善する  
～ 制約条件に基づいて改善を行う ～  
→ D(1)アイ 身近な計測・制御システムに込められた工夫に迫らせることで、技術の見方・考え方に気付かせる

- ① 「お掃除ロボ」シミュレーション教材を体験する  
→ D(1)イ 問題解決の工夫について触れながら体感する



お掃除ロボが走行する



掃除をしたところを、赤のマス目で塗る

# ① 「お掃除ロボ」シミュレーション教材を体験する



The screenshot displays the Scratch interface with a script for a vacuum robot simulation. The script is organized into three main sections:

- Initialization:** Sets the battery to 300, waits 3 seconds, and sets the battery to 0.
- Movement and Battery Management:** Moves the robot 10 steps, waits 0.1 seconds, and decreases the battery by 1 until it reaches 0.
- Sensor Detection and Action:** Uses color sensors to detect red, yellow, and blue. When red or yellow is detected, the robot moves -25 steps, turns 80 degrees, and draws a red line. When blue is detected, it moves -25 steps, turns 40 degrees, and draws a blue line. When green is detected, it turns 90 degrees. When the mouse pointer is over the robot, it moves the mouse pointer towards the robot.

The stage shows a grid with a red path and a blue obstacle. The sprite 'robo1' is positioned at x=215, y=160. The battery level is 0, and the number of blue squares is 142.

## プログラムの概要

緑の旗をクリックする

バッテリー設定300→0

ペンを下ろす

「robo1」の赤・黄のセンサーに触れたら、向きを変える

バッテリーが0になったら

マス判定する  
青→青、白→白  
水色→赤

それぞれのマスを数える

# robo1のプログラム



```

[がクリックされたとき]
  [バッテリーを 300 にする]
  [3 秒待つ]
  [バッテリー < 1 まで繰り返す]
  [10 歩動かす]
  [0.1 秒待つ]
  [バッテリーを -1 ずつ減らす]

もし [色 が 色に触れた] なら
  [-25 歩動かす]
  [60 度回す]

もし [色 が 色に触れた] なら
  [-25 歩動かす]
  [40 度回す]

もし [色 が 色に触れた] なら
  [30 度回す]

もし [色 が 色に触れた] なら
  [30 度回す]

[ペンを上げる]
[x座標を 215、y座標を 160 にする]
[0 度に向ける]
[ゴミ処理判定を送る]

```

緑の旗をクリックする

バッテリー設定300→0

「robo1」の赤・黄色のセンサーに触れたら、向きを変える

「robo1」のボディやペンが壁に触れても、向きを変える(念のための処置)

「robo1」を移動させ、「ゴミ処理判定」を送る

```

[がクリックされたとき]
  [計測したマス目を 0 にする]
  [掃除できていないマス目を 0 にする]
  [青のマス目を 0 にする]

```

マス目の設定を「0」にする

```

[がクリックされたとき]
  [全部消す]
  [x座標を -207、y座標を -147 にする]
  [ペンを下ろす]
  [全部消す]
  [ペンの色を 青 にする]
  [ペンの太さを 15 にする]
  [0 度に向ける]

```

「robo1」をスタート位置にし、ペンを下ろす



# マス目のプログラム



```
ゴミ処理判定を受け取ったとき
  計測したマス目 を 0 にする
  排除できていないマス目 を 264 にする
  青のマス目 を 0 にする
```

「ゴミ処理判定」を受け取ったら

マス目を横に24回、縦に17回ずらしながら繰り返す  
水・青・白色に触れたら、それぞれカウントするとともに、マス目をそれぞれのコスチュームを変える

マス目をずらしながら、スタンプしていく

※左のプログラムの続きから

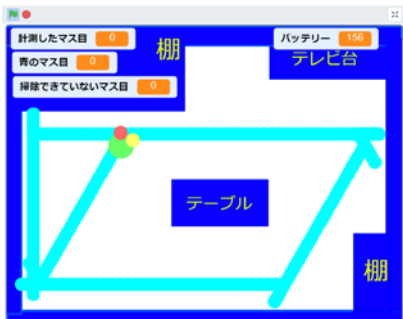
```
  声を アルト にする
  計測したマス目は をしゃべる
  計測したマス目 をしゃべる
  です をしゃべる
  もし 排除できていないマス目 = 0 なら
    声を テノール にする
    完了ですおめでとう! をしゃべる
    complete! Congratulations! と 2 秒待つ
  できなかったら
    声を テノール にする
    まだ、排除できていないマス目が をしゃべる
    264 - 計測したマス目 をしゃべる
    あります をしゃべる
```

判定結果を音声で伝える

※右のプログラムに続く



# ① 「お掃除ロボ」シミュレーション教材を体験 ⇒その後の生徒の感想(予想)



同じところを動くので、センサーを増やしたら、いっぱい塗られるのでは？

バッテリーの寿命（計測時間）を伸ばせば、いっぱい塗られるのだろうか？

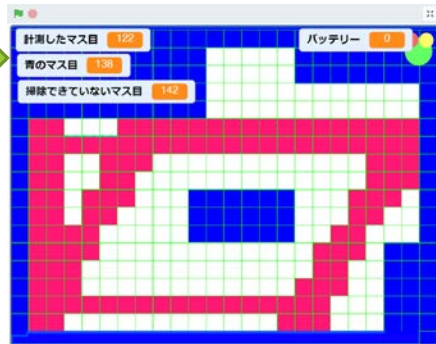
水色がもっと塗られるようにするには、どうすればよいだろうか？



世の中に出回っているお掃除ロボットは、どんな動きをするのだろうか？

どんな形のお掃除ロボットがあるのだろうか？

どんな性能を備えているのだろうか？



- ② 市場で売られている「お掃除ロボ」の規格や特長を調べる  
→ D(1)イ 問題解決の工夫について調べる中で、技術の見方・  
考え方に気付かせる

世の中に出回っているお掃除ロボットは、どんな動きをするのだろうか？

どんな形のお掃除ロボットがあるのだろうか？

どんな性能を備えているのだろうか？

様々な疑問に対し、個人  
またはグループで調べ、開  
発者の意図や思いに触れる  
機会とする



### ③ シミュレーション教材を活用して改善する ～ 制約条件に基づいて改善を行う ～

→ D(1)アイ 身近な計測・制御システムに込められた工夫に迫らせることで、技術の見方・考え方に気付かせる

#### 制約条件

##### ○絶対条件

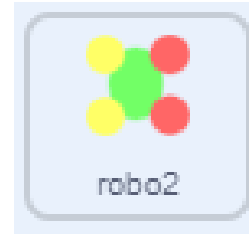
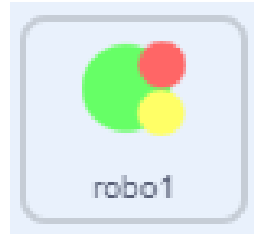
- ・バッテリーは300
- ・ペンの太さは1.5で統一

##### ○緩和条件

- ・ロボットの形は変えてよい
- ・センサー数は4つまで使用可
- ・ロボットの動きは自由に変えられる

##### ○禁止条件

- ・コースは変えてはならない
- ・スタート位置は変えてはならない



##### ○ robo2

- ・センサーは4つ
- ・センサーを増やすことで動きを複雑にする

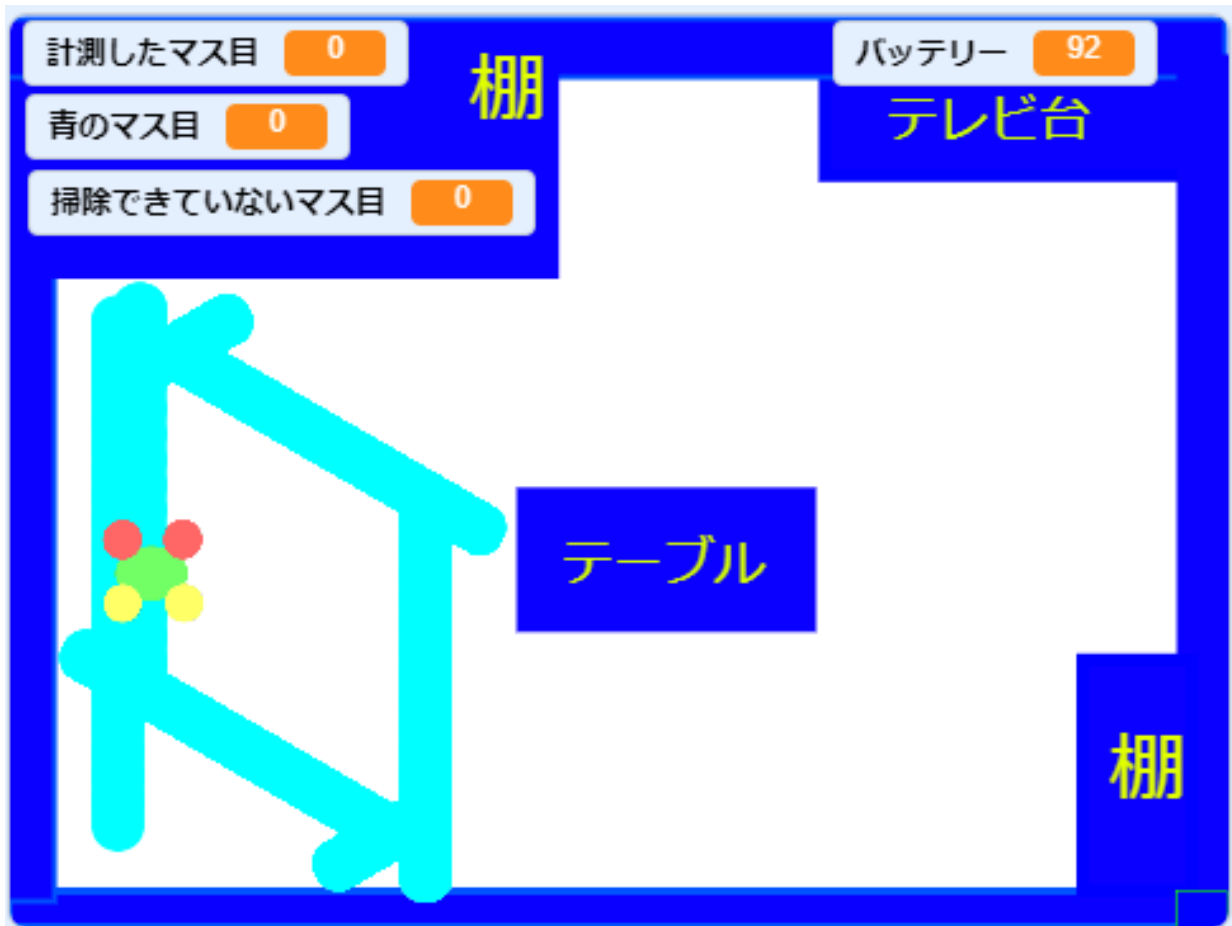
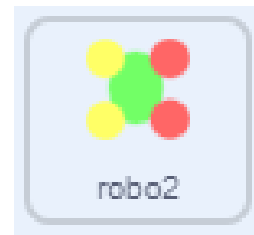


##### ○ robo3

- ・センサーは3つ
- ・センサー接触後、それぞれ違う動きをさせることで塗られる面積を増やす

## robo2のプログラム

- ・センサーは4つ
- ・センサーを増やすことで動きを複雑にする



センサー数を増やしたにもかかわらず、同じところを繰り返すだけになってしまった...



## robo3のプログラム

- ・センサーは3つ
- ・センサー接触後、それぞれ違う動きをさせることで塗られる面積を増やす



違う動きをさせるようプログラムすることで、塗られる面積が増えたぞ！反射角度や回転方向を変えたら、どんな動きになるだろう？



### ③ シミュレーション教材を活用して改善する ～ 制約条件に基づいて改善を行う ～



ここでは、生徒にプログラムの構造を理解させ上で、制約条件の中で自由な発想させ、プログラムの改善・実行・修正を行います。

### ③ シミュレーション教材を活用して改善する ～ 制約条件に基づいて改善を行う ～



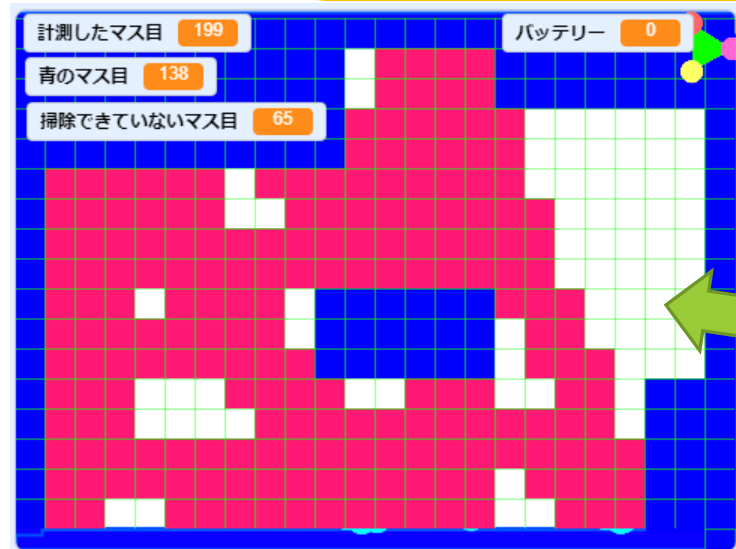
そうすることで、現在利用されている「お掃除ロボット」のプログラムの理解につながり、かつ、試行錯誤しプログラミングすることで開発者の思いや意図を体感することができます。





今回、マス目を用いた「ゴミ処理判定」で、AI処理のようなことをさせました。この処理では、残念ながら判定としての正確性を欠きます。つまり、プログラムの改善の余地が、まだまだあります。

しかしながら、このような体験をさせることで、現在の技術の進展を実感することができ、更なる改善への意欲につながるものと思われます。



掃除できていないマス目 **65**

塗られていないマス目を数えると  
**「72」**になります。  
計測で示した**「65」**とは異なります。