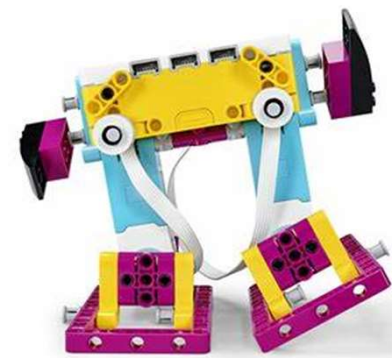


はじめての
レゴ ロボット
(SPIKE™ プライム)



Produce by Tetsuya FUKUDA, Robotics Educator

まずは、アプリを入れて環境設定しましょう！

- アプリをダウンロードして、プログラミングの準備をしましょう。

<https://education.lego.com/ja-jp/start/spike-prime/download>

※SPIKEアプリは、Windows 10、ChromeOS、MacOS、iOS、Androidと互換性あり。
インストールには、インターネット接続が必要です。

※インストールをしたら、インターネット環境がなくても、
プログラミングができます。



次に、ロボットをつくろう!

- まずは、基本ロボットをつくりましょう。
作り方は次のアドレスからダウンロード

<https://www.otemon-is.ed.jp/uploads/posts/pdf1/3163.pdf>



さあ、プログラミングに挑戦!

まずは、基本プログラムをつくりましょう。
テキストは次のアドレスからダウンロード

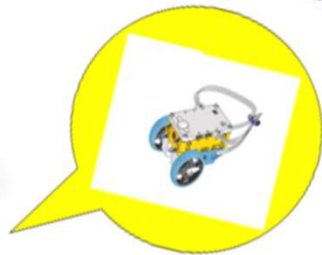
[https://www.otemon-
js.ed.jp/uploads/posts/pdf2/3163.pdf](https://www.otemon-js.ed.jp/uploads/posts/pdf2/3163.pdf)

- ※ 簡単なミッションを授業やセミナーで使用するときはiPadがおすすめ
- ※ ロボコンなどの難しいミッションをするときはPCがおすすめ

スパイク基礎プログラム



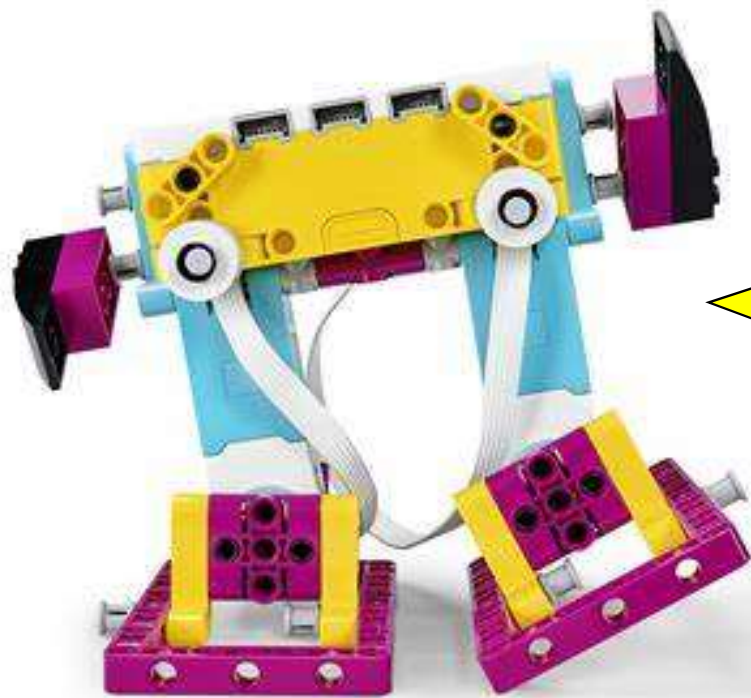
まずは、動かしてみよう



モーターは (EとF)

センサーは (A)

まずは、スパイクとiPadを接続しよう

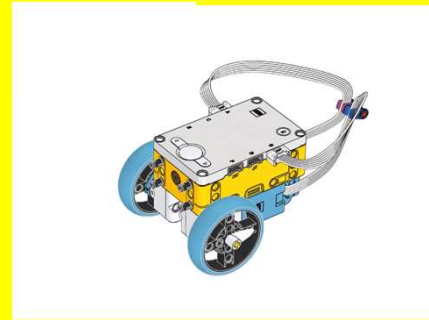


スパイク ^{きそ}基礎 プログラム



モーターは (EとF)

まずは、動かしてみよう



センサーは (A)

【基本1】 前に10cm進む

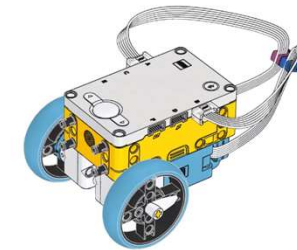


接続

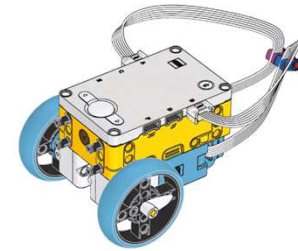
プログラムがスタートしたとき

移動に使うモーターを E+F にする

↑ の向きに 10 cm 移動する



【基本2】 前に20cm進む



▶ プログラムがスタートしたとき

移動に使うモーターを E+F ▼ にする

↑ ▼ の向きに 20 cm ▼ 移動する

【基本3】 前進20cmして、左に曲がる

接続



▶ プログラムがスタートしたとき

移動に使うモーターを E+F ▼ にする


↑ ▼ の向きに 20 cm ▼ 移動する

↶ ▼ の向きに 10 cm ▼ 移動する

↑ ↓ ↶ ↷

【基本4】 前進20cmして右に90度曲がる

16



The image shows a sequence of three programming blocks on a grid background. The first block is a yellow 'when program starts' block. The second block is a pink 'move' block with a motor icon, 'E+F' selected in a dropdown, an upward arrow icon, a '20' value field, and 'cm' selected in a dropdown. The third block is a pink 'turn' block with a motor icon, a right-turn arrow icon, a '6.8' value field, and 'cm' selected in a dropdown. A yellow dashed circle highlights the '6.8' value, with a callout bubble pointing to it.

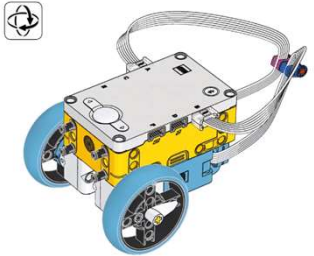
プログラムがスタートしたとき

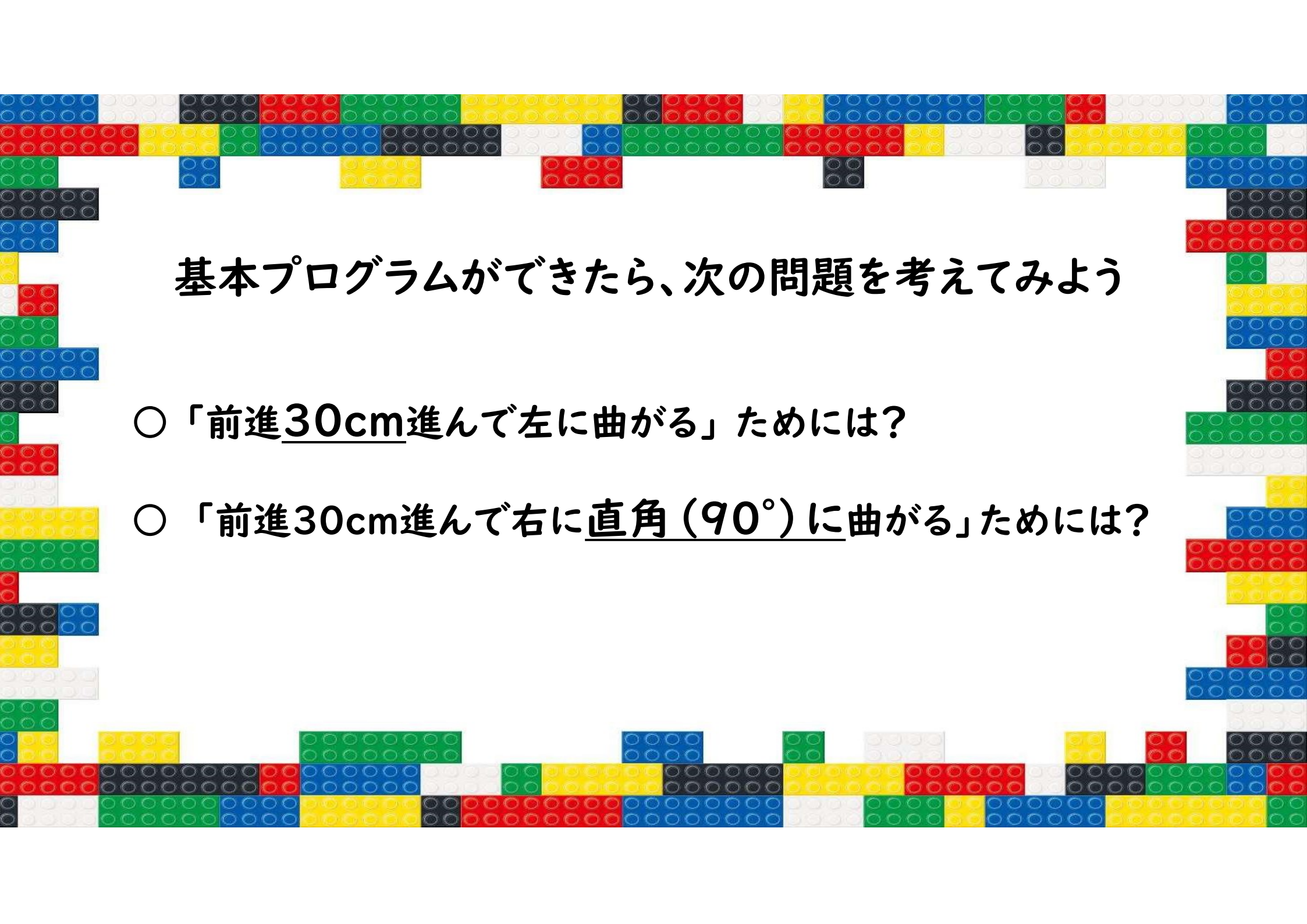
移動に使うモーターを E+F ▾ にする

↑ ▾ の向きに 20 cm ▾ 移動する

↻ ▾ の向きに 6.8 cm ▾ 移動する

自分のロボットの直角の数値を
はかろう！必ずメモしよう！



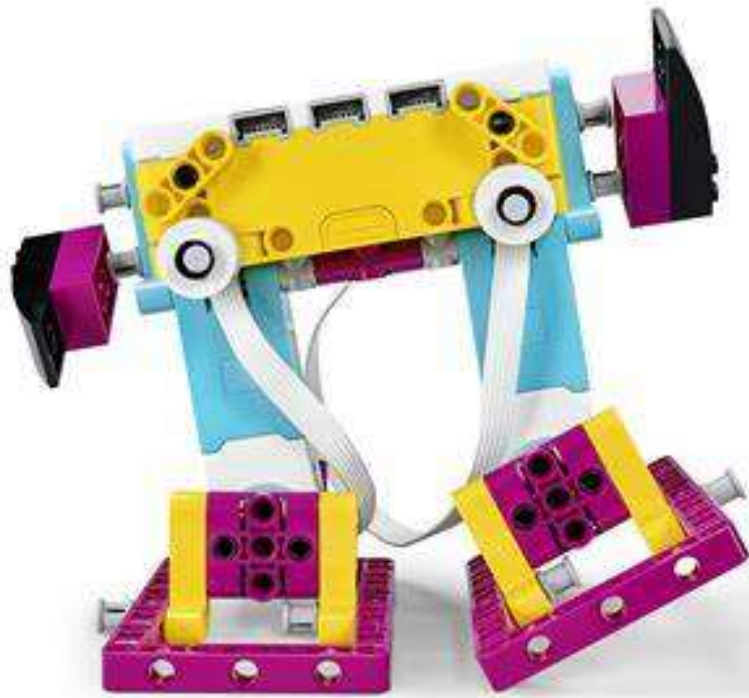


基本プログラムができたなら、次の問題を考えてみよう

- 「前進30cm進んで左に曲がる」ためには？
- 「前進30cm進んで右に直角 (90°)に曲がる」ためには？

おうよう

スパイク 応用 プログラム



モーターは (EとF)

センサーを使ってみよう!



センサーは (A)

センサーを使おう

○私たちは、目や耳などを使って情報をえます。
同じようにロボットは、センサーを使って情報をえます。

- ・ 距離センサー …… センサーと物体との距離をはかります。
- ・ タッチセンサー …… ボタンを押すと、感知します。
- ・ カラーセンサー …… 色や光の量をはかります。



距離センサー



タッチセンサー



カラーセンサー

【応用1】 距離センサー

前進して障害物が20cm以内であれば止まる



接続



```
graph TD; A[プログラムがスタートしたとき] --> B[移動に使うモーターを E+F にする]; B --> C[直進: 0 の向きに移動開始する]; C --> D[A 距離が 20 cm より近い まで待つ]; D --> E[移動をやめる]; E --> F[Hello を表示する];
```


【応用2】 タッチセンサー

前進して、タッチすると、止まる



接続

The diagram shows a sequence of programming blocks on a grid background. At the top left is a '接続' (Connect) icon. The main sequence consists of the following blocks:

- Yellow block: プログラムがスタートしたとき (When program starts)
- Pink block: 移動に使うモーターを E+F にする (Set motor for movement to E+F)
- Pink block: 直進: 0 の向きに移動開始する (Start moving forward: 0)
- Orange block: A が 押された まで待つ (Wait until button A is pressed)
- Pink block: 移動をやめる (Stop movement)
- Purple block: Hello を表示する (Display Hello)

【応用3】 カラーセンサー 前進して、赤になると止まる



接続

The diagram shows a sequence of blocks for a Scratch-style programming environment. It starts with a 'When program starts' block (yellow), followed by a 'Set motor to E+F' block (pink), a 'Move forward 0 units' block (pink), a 'Wait until color is red' block (orange), a 'Stop movement' block (pink), and finally a 'Say Hello' block (purple).

- ▶ プログラムがスタートしたとき
- 移動に使うモーターを E+F にする
- 直進: 0 の向きに移動開始する
- A 色が まで待つ
- 移動をやめる
- Hello を表示する

技術を高めよう（I）

3つのプログラム処理の基本を学ぼう

- 順次処理
- 繰り返し
- 条件分岐

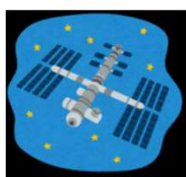


【ヒント1】 順番にならべる (順次)

20cm前進し、

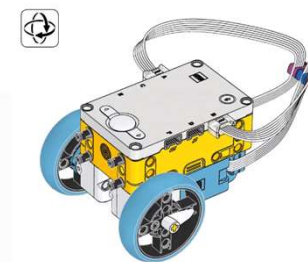


を表示



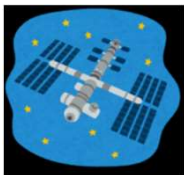
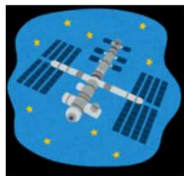
The diagram shows a sequence of three blocks for a robot program:

- Yellow block:** プログラムがスタートしたとき (When the program starts)
- Pink block:** 移動に使うモーターを E+F にする (Set motor to E+F)
- Pink block:** ↑ の向きに 20 cm 移動する (Move 20 cm forward)
- Purple block:** 2 秒間 LED をオンにする (Turn on LED for 2 seconds)



【ヒント2】 繰り返す (繰り返し)

ヒント 1 を2回繰り返す。



The diagram shows a Scratch-style block diagram for a robot program. It consists of the following blocks:

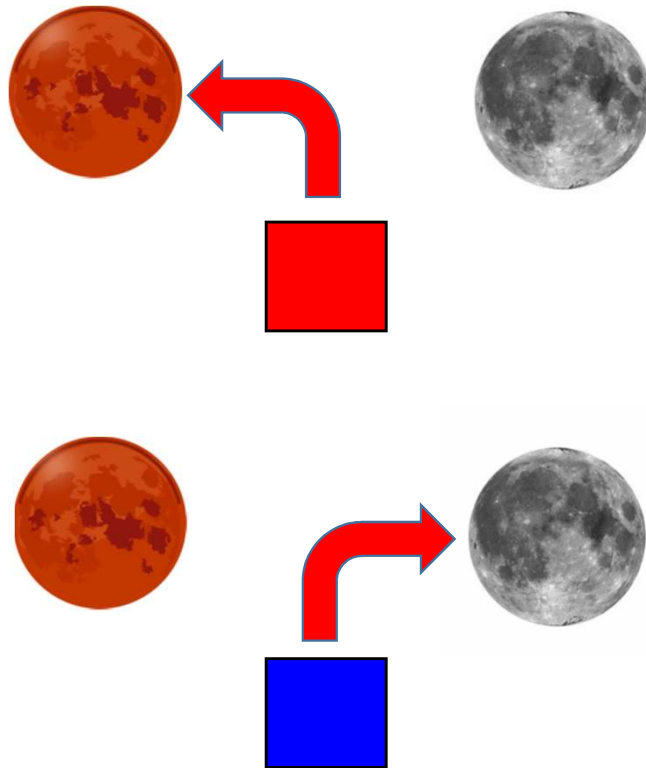
- Start Block:** A yellow block with a play button icon and the text "プログラムがスタートしたとき".
- Repeat Loop:** An orange block with a "2" in a circle and the text "回繰り返す".
- Motor Selection:** A pink block with a motor icon, the text "移動に使うモーターを", a dropdown menu showing "E+F", and the text "にする".
- Move:** A pink block with a motor icon, a dropdown menu showing an upward arrow, the text "の向きに", a circle containing "20", a dropdown menu showing "cm", and the text "移動する".
- LED Control:** A purple block with a keypad icon, a circle containing "2", the text "秒間", a dropdown menu showing a grid of LEDs, and the text "をオンにする".
- Loop Arrow:** A white arrow at the bottom right of the orange repeat block, indicating the loop repeats.



【ヒント3】 もし・・・（条件分岐）



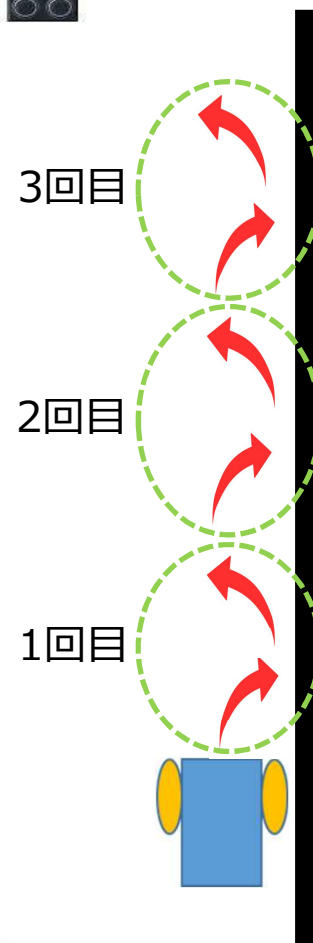
もし赤ならば、「火」を2秒表示して、左に10cm？
でなければ、「月」を2秒表示して、右に10cm？



```
graph TD; Start[プログラムがスタートしたとき] --> Motor[移動に使うモーターを E+F にする]; Motor --> If[もし A が 色が 赤 ならば]; If --> LED1[2 秒間 LED をオンにする]; LED1 --> Move1[の向きに 10 cm 移動する]; If --> Else[でなければ]; Else --> LED2[2 秒間 LED をオンにする]; LED2 --> Move2[の向きに 10 cm 移動する];
```

技術を高めよう(Ⅱ)

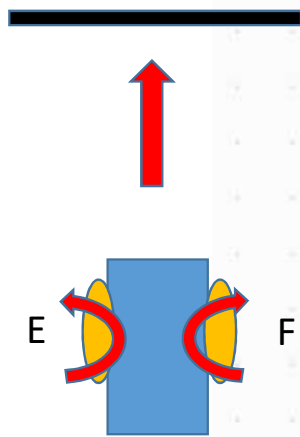
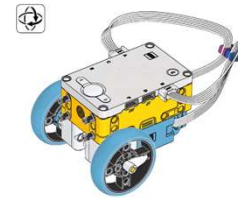
ライトレースに挑戦しよう



【ライントレース1】

前進して黒の線で止まる。

16



ブロックチェーン:

- ▶ プログラムがスタートしたとき
- 移動に使うモーターを E+F にする
- E を 方向に回す (反時計回り)
- F を 方向に回す (時計回り)
- A 反射光が < 70 % まで待つ
- 移動をやめる

Eのモーターは
時計と反対回りで前
Fのモーターは
時計回りで前

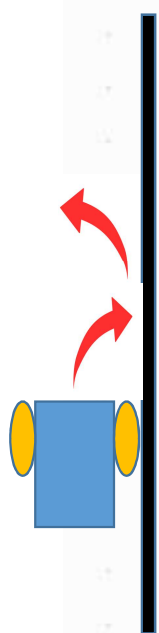
意外と忘れがちな
止めるプログラム

設定は白と黒の反射光の
平均の数字を入れよう



【ライントレース2】

左のモーター（E）だけ動かして黒の線で反応したら、
右のモーター（F）だけを、0.3秒動かして止まる。



The diagram shows a blue sensor unit with two yellow lenses on the left, positioned to detect a vertical black line. Two red curved arrows indicate the sensor's field of view.

プログラムがスタートしたとき

- 移動に使うモーターを E+F にする
- E を 方向に回す
- A 反射光が < 70 % まで待つ
- 移動をやめる
- F を の方向に 0.3 秒 回す

Eのモーターだけを動かして黒で止まる

Fのモーターだけを0.3秒動かして止まる

【ライントレース3】

ライントレース2を10回繰り返す

The diagram illustrates a line following robot (a blue and yellow mobile robot) moving along a vertical black line. Three dashed green circles, labeled '1回目' (1st), '2回目' (2nd), and '3回目' (3rd), show the robot's path as it moves up the line and turns back to the start, indicated by red curved arrows.

The code block is a vertical stack of blocks:

- Start block: プログラムがスタートしたとき (When program starts)
- Loop block: 10 回繰り返す (Repeat 10 times) - circled in red with a callout box.
- Motor block: 移動に使うモーターを E+F にする (Set motor E+F for movement)
- Turn block: E を 方向に回す (Turn motor E in direction)
- Wait block: A 反射光が < 70 % まで待つ (Wait until sensor A reflects light < 70%)
- Stop block: 移動をやめる (Stop movement)
- Turn block: F を の方向に 0.3 秒 回す (Turn motor F in direction for 0.3 seconds)

A callout box points to the '10 回繰り返す' block with the text: 目的を達成するためには何回繰り返せばよいのだろうか? (How many times should it be repeated to achieve the goal?)

16 

応用プログラムができたなら、あなたはロボットマスター！

World Robot Olympiad や
FIRST LEGO Leagueなどのロボコンに挑戦しよう！



コンピュータ技術が進む中、
世界中の子どもたちが、プログラミングに挑戦しています。

(ロボコン世界大会の映像: World Robot Olympiad 2018)
<https://vimeo.com/351091648/32c0cc8a52?fbclid=IwARlkhQ0Yk9jIqqJU%20VtViAPQI-fUM6riXQI4ckIdQqRsMKUflx4rLjgFCP9M>

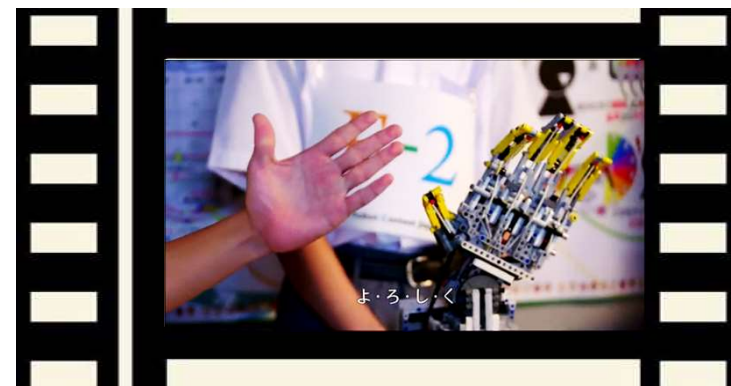


みなさんも、もっともっとプログラミングに挑戦しよう！
そして、すばらしい未来をつくってください。

Tetsuya FUKUDA

おまけ

私のロボット教育に対する思いを
10分のショートムービーに映像化
していただきました。



<https://www.youtube.com/watch?v=4qAdFCZGRaU&t=493s>

フリップの文字は、
私がみなさんに伝えたいメッセージです。

