

LEGO MINDSTORM 応用編

【目的】

レゴブロックマインドストームの組立、パソコンを使用したプログラミングを通して、機械装置の基本構造の理解と制御方法について学ぶ。

☆プログラミング編

多くの電子機器は、人からの命令を受けて実行しています。このマインドストームも例外ではありません。今まで動作させていたマインドストームも命令の下に動いていたのです。この命令のことをプログラムと言い、プログラムを組むことをプログラミングと言います。

この訓練では、マインドストームを使ってプログラミングを体験しましょう。

■使用方法

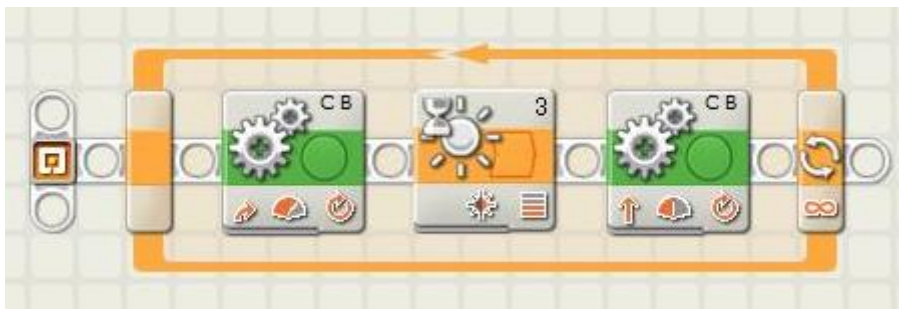
デスクトップ上にある「NXT 2.0 Programming」をダブルクリックすると、「LEGO MINDSTORMS Education NXT プログラミング」が起動します。初めに「始める前に」と「ソフトウェア概要」の説明を見ておくと理解しやすくなります。

メニューバーのファイル→新規で新しいプログラムが作成できます。

プログラミングの方法

LEGO MINDSTORMS Education NXT プログラミングでは、動作を命令する部品であるブロックを組み合わせてプログラムを作成します。

例として下の図のようなプログラムがあるとします。



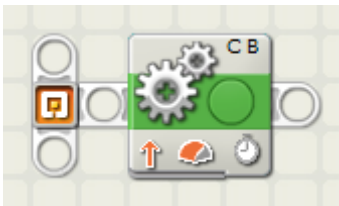
がスタートとなっており、ここを起点としてプログラムが動作します。ラインの最後まで処理を行うとプログラムは終了します。

■例題1 前進

●1-1. 5秒間だけ前進するプログラムを組んでみましょう。

まずはプログラムの基本として、5秒間車を前進させてみましょう。

1. 左にあるプログラミングパレットから移動ブロックをクリックします。
2. 設置した居場所をクリックするとブロックが配置されます。
3. 設置したブロックをドラッグすると移動させることができます。
4. スタート地点から線が繋がるようにブロックの位置を調整しましょう。
5. 間違えて違うブロックを選んでしまった場合は、マップ外をクリックするとキャンセルできます。



6. 配置したブロックをクリックすることで選択することができます。
7. ブロックを削除したい場合は、ブロックを選択した状態で Delete キーを押せば削除できます。
8. ブロックを選択すると下の図のような設定パネルが画面の下の方に表示されます。



9. 設定パネルにある持続時間の欄を「5 秒」に変更します。
10. メニューバーのファイル→別名で保存を選択します。
11. プログラムの名前を「reidai1-1」にして保存します。
12. PC と NXT（白い端末）を USB ケーブルで接続します。
13. 画面の右下に下の図のようなパネルがあります。これをコントローラーと言います。

このコントローラーの「↓」をクリックすると NXT にプログラムがダウンロードされます。



14. NXT の USB ケーブルを抜き、NXT のオレンジ色のボタンを 3 回押します。

15. ダウンロードした「reidai1-1」のファイルを選択し、オレンジ色のボタンを押します。

16. 車が 5 秒間走り出したら完成です。

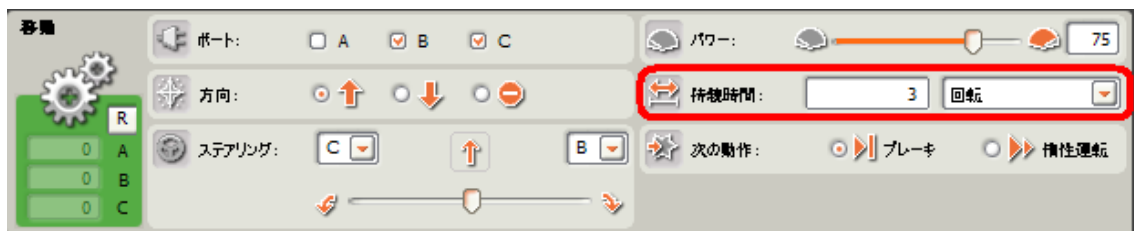
これで、車が前進するプログラムの作り方が分かりました。

●1-2. タイヤが3回転するまで前進させましょう。

1-1 では 5 秒間だけ車を前進させることが出来ました。次は、タイヤが 3 回転するまで前進させてみましょう。

※基本的なプログラミングの工程は 1-1 と同じなので、これからはブロックの配置や今までと違う部分だけを説明していきます。もし操作で分からないようであれば、1-1 を見直してみてください。

1. 移動ブロックを1つ配置します。
2. 配置したブロックを選択して、設定パネルにある持続時間の欄を「3 回転」に変更します。



3. プログラムの名前を「reidai1-2」としてNXT にダウンロードし、動作させてみましょう。
4. 車がタイヤ 3 回転だけ前進すれば完成です。

●1-3. タイヤが720度回転するまで後退させましょう。

移動ブロックでは、前進だけではなく後退も設定することが出来ます。今回は、タイヤが720度回転するまで後退するプログラムを作ってみましょう。

1. 移動ブロックを1つ配置します。
2. 配置したブロックを選択して、設定パネルにある持続時間の欄を「720度」に変更します。
3. また、方向の欄で「↓」を選択しましょう。



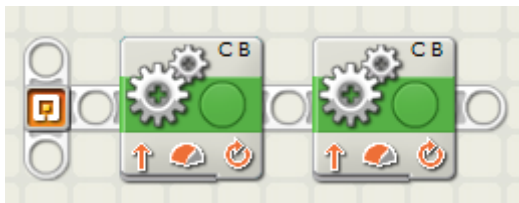
4. プログラムの名前を「reidai1-3」としてNXTにダウンロードし、動作させてみましょう。
5. 車が720度分後退すれば完成です。

待機時間の欄では、秒数、回転数、度数だけタイヤを動かすことが出来ることと、方向の欄で前進、後退を選べることがわかりました。

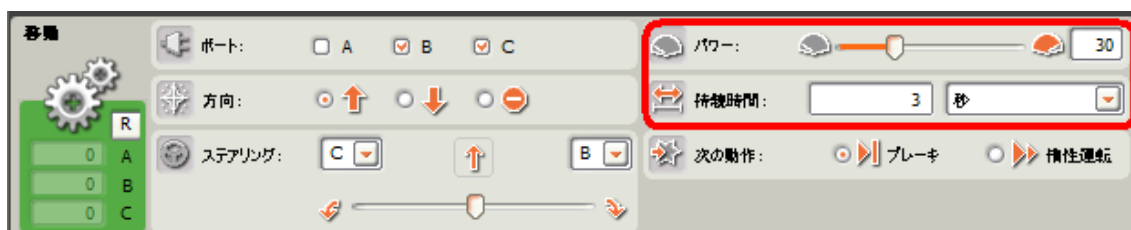
●1-4. ブロックを2つ繋げてみましょう。

今までは、1 つのブロックだけを配置していましたが、ブロックを複数配置することによって様々なプログラムを作成することが出来ます。今回は 3 秒間ゆっくり前進し、次の 5 秒間は速く前進するプログラムを作成しましょう。

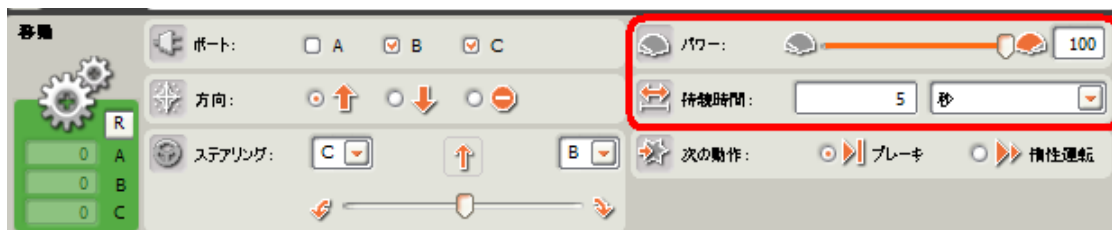
1. 移動ブロックを2つ繋げて配置します。



2. 左の移動ブロックを選択します。
3. 設定パネルでパワーを 30、待機時間を 3 秒に設定します。



4. 右の移動ブロックを選択します。
5. 設定パネルでパワーを 100、待機時間を 5 秒に設定します。



6. プログラムの名前を「reidai1-4」として NXT にダウンロードし、動作させてみましょう。
7. 車が 3 秒間ゆっくり動き、次に 5 秒間速く動けば完成です。

移動ブロックでは、パワーの欄で速度を調整できることと、ブロックを複数活用することで色々な動作が出来ることが分かりました。

●1-5. 慣性運転について学びましょう。

前回は、2つのブロックを用いて3秒と5秒の間に車の速度を変えるプログラムを作成しました。しかし、3秒と5秒の切れ目で車の動きが一時停止してしまっていることが分かります。この一時停止を改善するための設定について説明します。

1. 「reidai1-4」を開きます。
2. 左の移動ブロックを選択します。
3. 設定パネルの次の操作の欄で「慣性運転」を選択します。

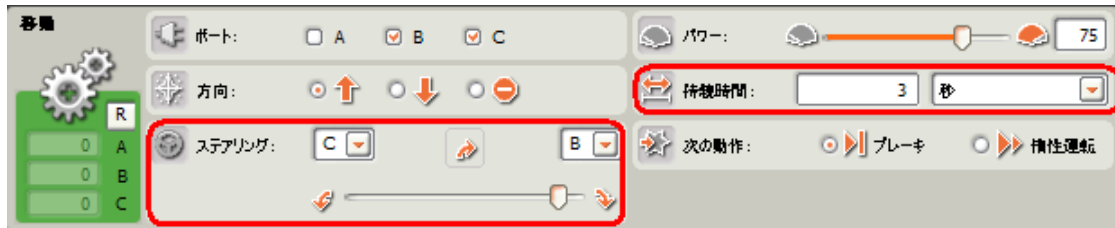


4. プログラムの名前を「reidai1-5」としてNXTにダウンロードし、動作させてみましょう。
5. 1-4 と 1-5 のプログラムを用いて交互に走らせ、違いを比べてみましょう。

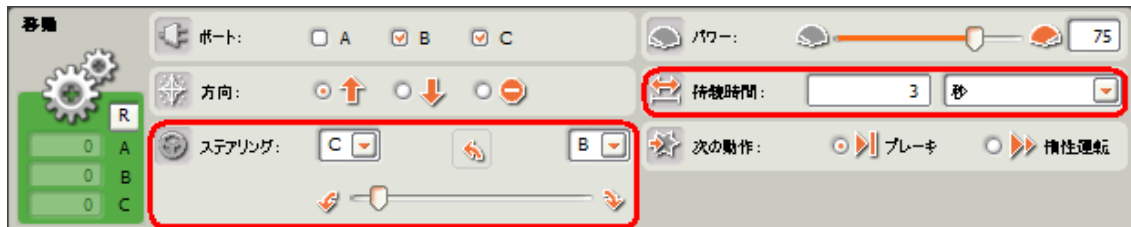
●1-6. 車をカーブさせてみましょう。

今までのプログラミングでは前進と後退のみを行ってきました。今回は車がカーブするプログラムを作成しましょう。

1. 移動ブロックを2つ繋げて配置します。
2. 左の移動ブロックを選択します。
3. 設定パネルでステアリングと待機時間を下の図のように設定します。



4. 右の移動ブロックを選択します。
5. 設定パネルでステアリングと待機時間を下の図のように設定します。

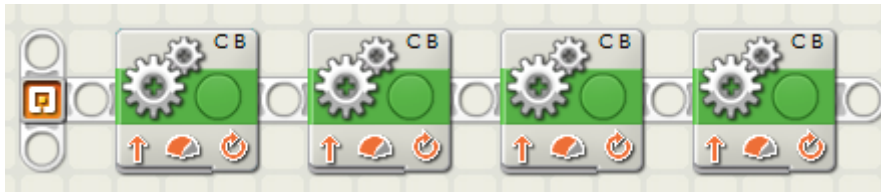


6. プログラムの名前を「reidai1-6」として NXT にダウンロードし、動作させてみましょう。

◆1-7. 車に切り返しをさせましょう。

前回のプログラミングで車をカーブさせることを学びました。今回は今まで学んだプログラムを活用して車に切り返しをさせてみましょう。

1. 移動ブロックを4つ繋げて配置します。



2. 左の移動ブロックを選択します。
3. 設定パネルで待機時間を 3 秒に設定します。
4. 左から 2 番目の移動ブロックを選択します。
5. 設定パネルでステアリングを下の図のように、方向を「↓」に、待機時間を 2.5 回転に設定します。



6. 右から 2 番目の移動ブロックを選択します。
7. 設定パネルでステアリングを下の図のように、待機時間を 2.5 回転に設定します。

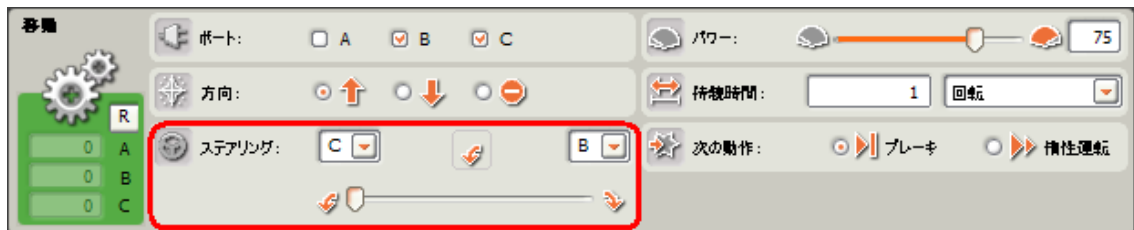


8. 右の移動ブロックを選択します。
9. 設定パネルで待機時間を 3 秒に設定します。
10. プログラムの名前を「reidai1-7」として NXT にダウンロードし、動作させてみましょう。
11. 車が 3 秒間進んだ後に切り替えして戻ってくれば完成です。

●1-8. その場で旋回（超信地旋回）させてみましょう。

前回は車に切り返しをさせるプログラムを作成しました。今回は、車は進まずにその場で旋回するようなプログラムを作成しましょう。

1. 移動ブロックを3つ繋げて配置します。
2. 左の移動ブロックを選択します。
3. 設定パネルで待機時間を3秒に設定します。
4. 真ん中の移動ブロックを選択します。
5. 設定パネルでステアリングを下の図のように設定します。



6. 右の移動ブロックを選択します。
7. 設定パネルで待機時間を3秒に設定します。
8. プログラムの名前を「reidai1-8」としてNXTにダウンロードし、動作させてみましょう。
9. 車がある程度進んだ時に旋回をして戻ってくれば完成です。

◆1-9. 180 度回転する設定を探しましょう。

この先は、1-8 のようにターンをして帰ってくるプログラムを組むことがよくあります。

そこで、90 度、180 度回転する設定値を調べてみましょう。

90 度回転の設定値 : _____

180 度回転の設定値 : _____

■例題2 待機

●2-1. 3秒前進→2秒待機→3秒後退

例題1では、移動ブロックのみを使用してプログラミングを行っていました。

例題2では、待機ブロックを使用して様々なプログラムを作成します。待機ブロックは「～するまで待機」という処理を行います。

今回は、3秒進み、2秒経つまで待機し、3秒後退するプログラムを作成しましょう。

1. 移動ブロックを2つと待機ブロックを図のように繋げて配置します。



2. 左の移動ブロックを選択します。

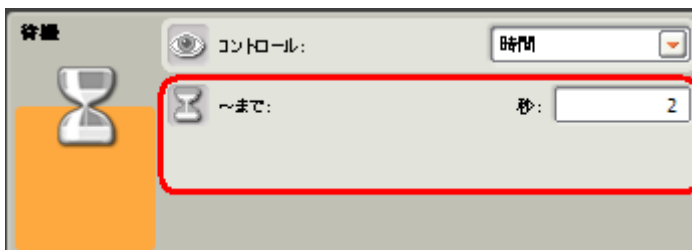
3. 設定パネルで待機時間を3秒に設定します。

4. 右の移動ブロックを選択します。

5. 設定パネルで方向を「↓」、待機時間を3秒に設定します。

6. 真ん中の待機ブロックを選択します。

7. 設定パネルで「～まで」を2秒に設定します。



8. プログラムの名前を「reidai2-1」としてNXTにダウンロードし、動作させてみましょう。

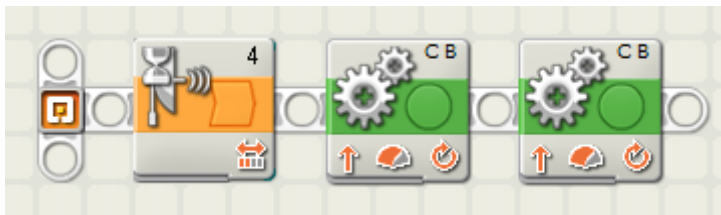
●2-2. 超音波センサー反応まで待機→3秒前進→3秒後退

前は、時間での待機でしたが、今回は超音波センサーを用いた待機について学びます。

超音波センサー：超音波を用いることによって、物体が超音波センサーの前にあると反応をします。また、距離を測ることもできます。

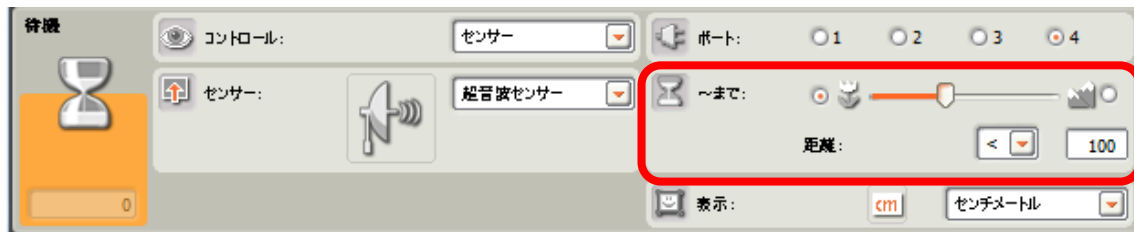
目の前に物体が通り過ぎたら車が3秒前進し、次に3秒後退するようなプログラムを作成しましょう。

1. 待機ブロックと移動ブロックを2つ、図のように繋げて配置します。



2. 左の待機ブロックを選択します。

3. 設定パネルで「～まで」の部分を下の図のように設定します。



4. 真ん中の移動ブロックを選択します。

5. 設定パネルで待機時間を3秒に設定します。

6. 右の移動ブロックを選択します。

7. 設定パネルで方向を「↓」、待機時間を3秒に設定します。

8. プログラムの名前を「reidai2-2」としてNXTにダウンロードし、動作させてみましょう。

●2-3. プッシュセンサー反応まで待機→3 秒前進→ターン→3 秒前進

今回はプッシュセンサーを用いた待機について学びましょう。

プッシュセンサー：センサーが押されることによって条件を判断します。

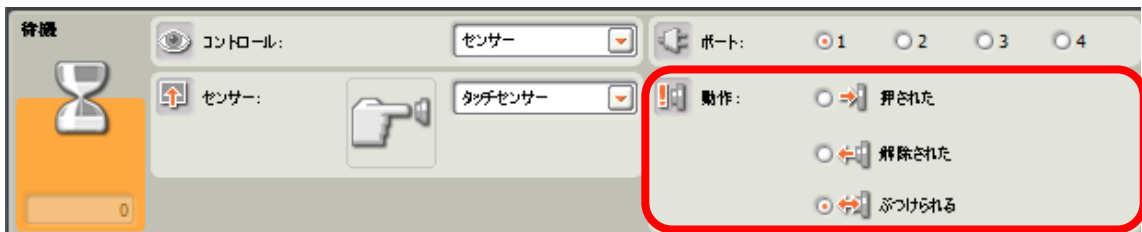
目の前に物体が通り過ぎたら車が 3 秒前進し、次に 3 秒後退するようなプログラムを作成しましょう。

1. 待機ブロックと移動ブロックを2つ、図のように繋げて配置します。



2. 左の待機ブロックを選択します。

3. 設定パネルで「操作」の部分で「ぶつけられる」を選択します。



4. 真ん中の移動ブロックを選択します。

5. 設定パネルで待機時間を 3 秒に設定します。

6. 右の移動ブロックを選択します。

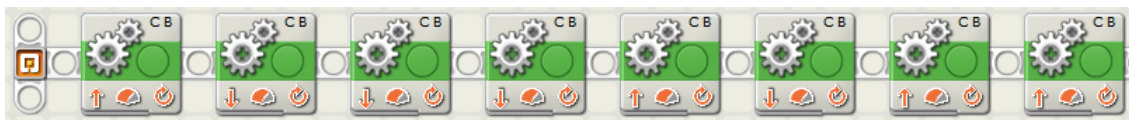
7. 設定パネルで方向を「↓」、待機時間を 3 秒に設定します。

8. プログラムの名前を「reidai2-3」として NXT にダウンロードし、動作させてみましょう。

■例題3 ループ

今までのプログラムはスタート地点からラインを右にたどって処理を行っていました。

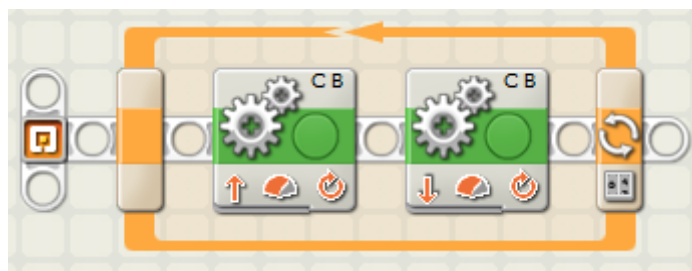
では、同じ処理を何度もしたいと思った時はどうしたらいいでしょうか。例えば、前後に動き続けるプログラムを作りたいとすると、



このようにずっとラインを伸ばして同じブロックを置き続けなければなりません。

ここで使うのがループブロックです。

ループブロックは、囲んだブロックの処理を指定した回数だけ繰り返すという役割があります。このループブロックを使って今回のプログラムを作るとすると、



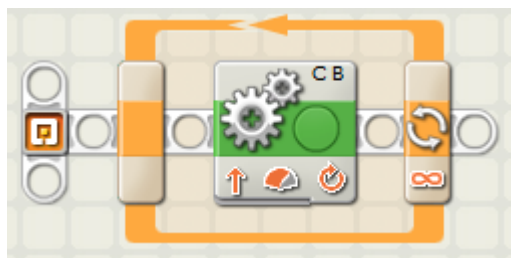
このように短く簡単に同じ処理を繰り返すことができます。

●3-1. 無限前進→障害物にぶつかる前に止まる …超音波センサー反応で停止

1. 移動ブロックとループブロックを図のように繋げて配置します。



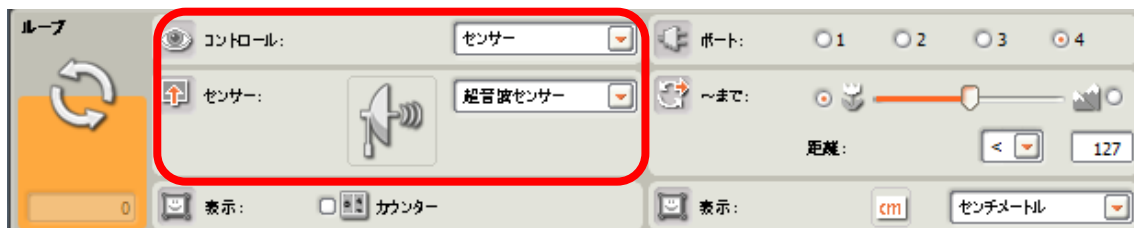
2. 右の移動ブロックをドラッグして、左のループブロックの中に入れてみましょう。



3. 移動ブロックの「待機時間」を無限に設定しましょう。

4. ループブロックを選択し、設定パネルの「コントロール：」を「センサー」に設定しましょう。

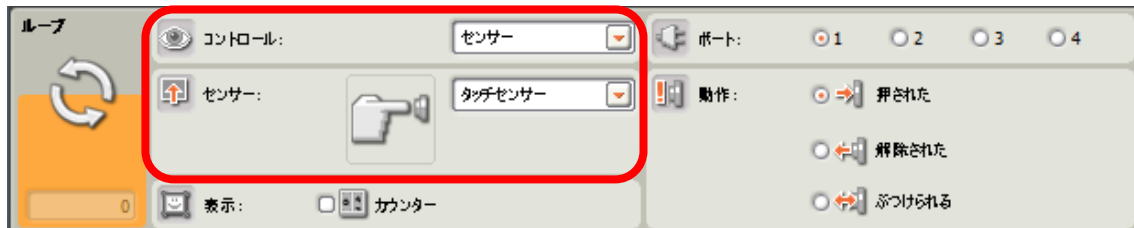
5. 「センサー：」を「超音波センサー」に設定しましょう。



6. プログラムの名前を「reidai3-1」としてNXT にダウンロードし、動作させてみましょう。

●3-2. 無限前進→障害物にぶつかったら停止 …ブッシュセンサー反応で停止

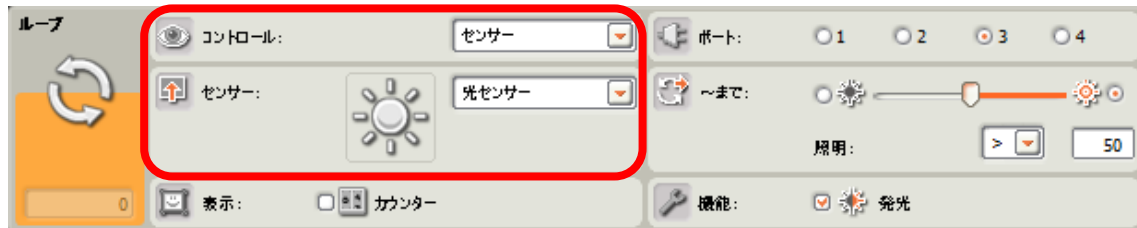
1. 移動ブロックとループブロックを繋げて配置します。
2. 移動ブロックをドラッグして、ループブロックの中に入れましょう。
3. 移動ブロックの「待機時間」を無限に設定しましょう。
4. ループブロックを選択し、設定パネルの「コントロール：」を「センサー」に設定しましょう。
5. 「センサー：」を「タッチセンサー」に設定しましょう。



6. プログラムの名前を「reidai3-2」としてNXTにダウンロードし、動作させてみましょう。

●3-3. 無限前進→ライン到達で停止 …光センサー反応で停止

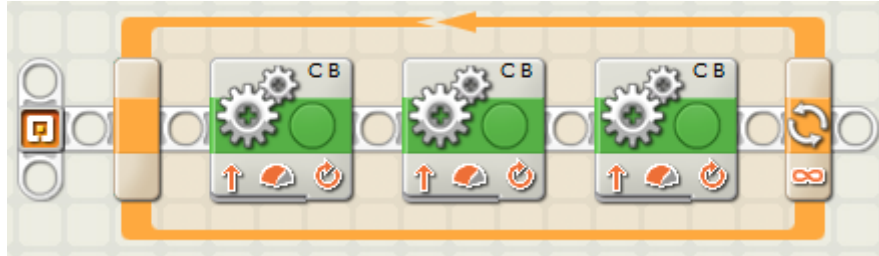
1. 移動ブロックとループブロックを繋げて配置します。
2. 移動ブロックをドラッグして、ループブロックの中に入れましょう。
3. 移動ブロックの「待機時間」を無限に設定しましょう。
4. ループブロックを選択し、設定パネルの「コントロール：」を「センサー」に設定しましょう。
5. 「センサー：」を「光センサー」に設定しましょう。



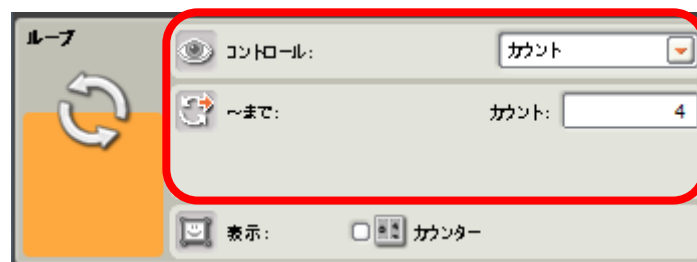
6. プログラムの名前を「reidai3-3」としてNXTにダウンロードし、動作させてみましょう。

●3-4. 3秒前進→ターン→3秒前進→180度旋回（超信地旋回）…4カウント
トープ

1. 移動ブロック3つとループブロックを繋げて配置します。
2. 移動ブロック3つをドラッグして、ループブロックの中に入れてみましょう。



3. 左と右の移動ブロックを3秒だけ前に進む設定にしましょう。
4. 真ん中の移動ブロックをその場で180度旋回するように設定しましょう。
5. ループブロックを選択し、「コントロール：」を「カウント」にしましょう。
6. 「カウント：」を「4」に設定しましょう。

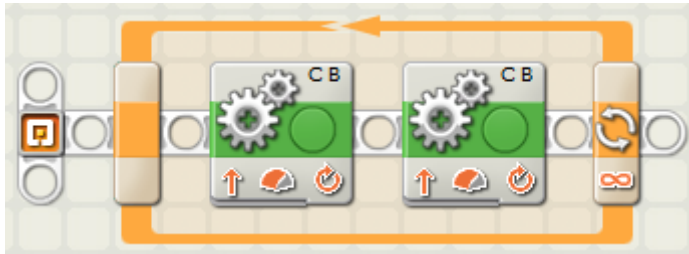


7. プログラムの名前を「reidai3-4」としてNXTにダウンロードし、動作させてみましょう。

●3-5. 3秒前進→180度旋回（超信地旋回）→……→ …… 無限ループ

今まではある条件を達成するまで繰り返すようにプログラミングしていました。
今回は、永遠に同じ動作を繰り返し続けるプログラムを作成してみましょう。

1. 移動ブロック2つとループブロックを繋げて配置します。
2. 移動ブロック2つをドラッグして、ループブロックの中に入れてみましょう。



3. 左の移動ブロックを3秒だけ前に進む設定にしましょう。
4. 右の移動ブロックをその場で180度旋回するように設定しましょう。
5. ループブロックを選択し、「コントロール：」が「永久」になっていることを確認しましょう。
6. プログラムの名前を「reidai3-5」としてNXTにダウンロードし、動作させてみましょう。

特に、このように永遠に同じ処理を行うループの事を無限ループと言います。

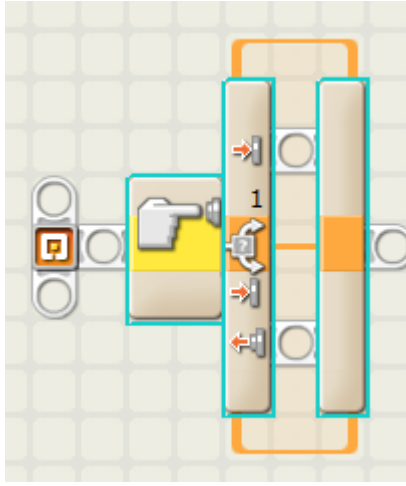
●3-6. 3秒前進→90度ターン→…→ … 無限ループ

1. 移動ブロック2つとループブロックを繋げて配置します。
2. 移動ブロック2つをドラッグして、ループブロックの中に入れましょう。
3. 左の移動ブロックを3秒だけ前に進む設定にしましょう。
4. 右の移動ブロックをその場で90度旋回するように設定しましょう。
5. ループブロックを選択し、「コントロール：」が「永久」になっていることを確認しましょう。
6. プログラムの名前を「reidai3-6」としてNXTにダウンロードし、動作させてみましょう。

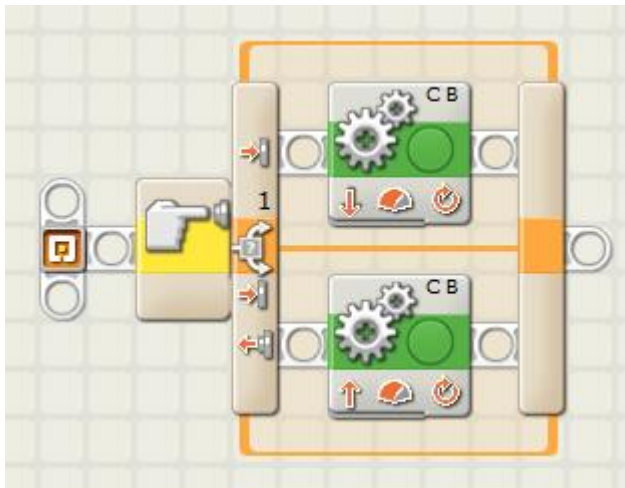
■例題4 スイッチ

今まで待機ブロックやループブロックについて学んできました。これにより、プログラミングの幅が広がってきたと思います。今回は、条件の判断を行うプログラムについて学びましょう。ここでは、スイッチブロックを使用します。スイッチブロックは「与えられた条件に合っていれば〇〇を、合っていないければ××をする」という処理を行います。

スイッチブロックは下の図のように 2 本道に分岐をします。

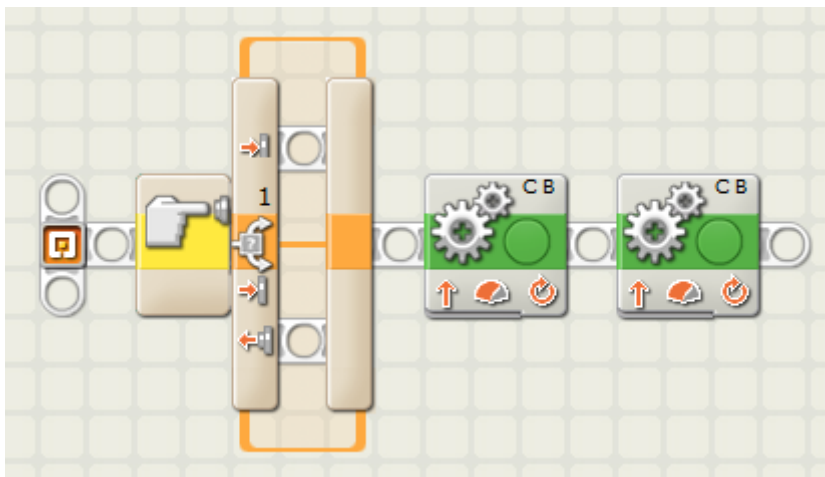


下の図の場合は、「タッチセンサーが押されていない場合は前に進み、押されている場合は後ろに戻る」というプログラムになります。

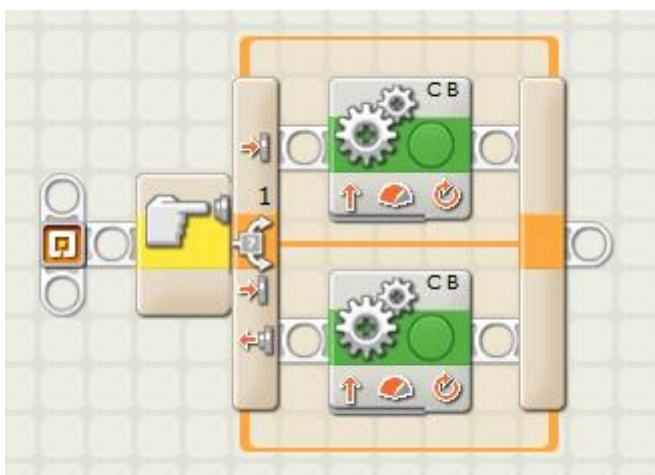


●4-1. 前進→障害物にぶつかったら→180度旋回（超信地旋回）→前進

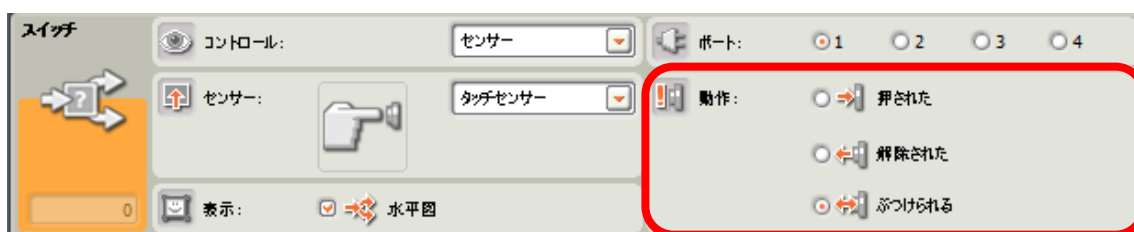
1. 移動ブロック2つとスイッチブロックを繋げて配置します。



2. 移動ブロック2つをドラッグして、スイッチブロックの中に入れましょう。



3. スイッチブロックを選択し、「操作：」の設定を「ぶつけられる」にしましょう。



4. 上の移動ブロックを無限に進む設定にしましょう。

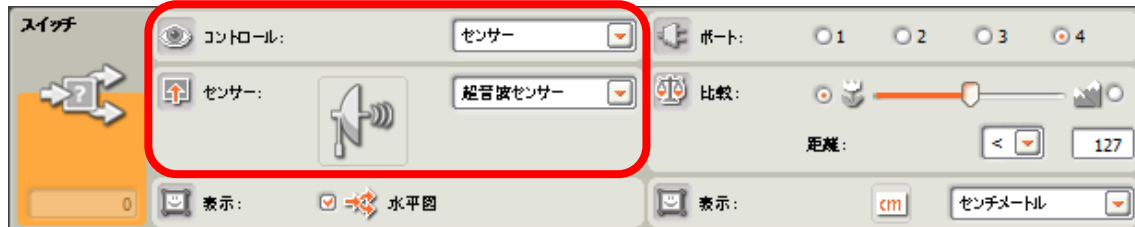
5. 下の移動ブロックをその場で 180 度旋回するように設定しましょう。

6. 最後に、作成したプログラムが無限ループするようにしましょう。

7. プログラムの名前を「reidai4-1」として NXT にダウンロードし、動作させてみましょう。

●4-2. 前進→障害物に近づいたら→180度旋回（超音波旋回）→前進

1. 移動ブロック2つとスイッチブロックを繋げて配置します。
2. 移動ブロック2つをドラッグして、スイッチブロックの中に入れましょう。
3. スイッチブロックを選択し、「センサー：」の設定を「超音波センサー」にしましょう。



4. 上の移動ブロックを無限に前に進む設定にしましょう。
5. 下の移動ブロックをその場で 180 度旋回するように設定しましょう。
6. 最後に、作成したプログラムが無限ループするようにしましょう。
7. プログラムの名前を「reidai4-2」として NXT にダウンロードし、動作させてみましょう。

●4-3. 前進→ライン到達→180度旋回（超信地旋回）→前進

1. 移動ブロック2つとスイッチブロックを繋げて配置します。
2. 移動ブロック2つをドラッグして、スイッチブロックの中に入れましょう。
3. スイッチブロックを選択し、「センサー：」の設定を「光センサー」にしましょう。
4. 上の移動ブロックを速く前に進む設定にしましょう。
5. 下の移動ブロックを遅く前に進む設定にしましょう。
6. 最後に、作成したプログラムが無限ループするようにしましょう。
7. プログラムの名前を「reidai4-3」としてNXTにダウンロードし、動作させてみましょう。

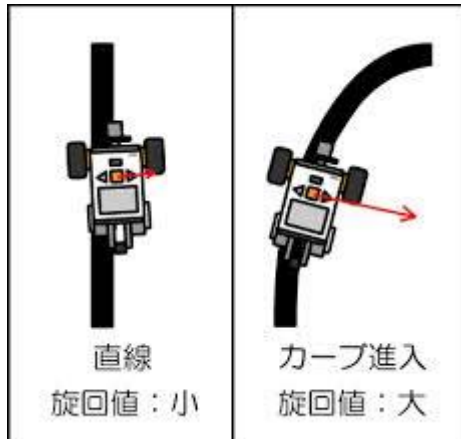
●4-4. 床が明るい→速い 床が暗い→遅い

1. 移動ブロック2つとスイッチブロックを繋げて配置します。
2. 移動ブロック2つをドラッグして、スイッチブロックの中に入れましょう。
3. スイッチブロックを選択し、「センサー：」の設定を「超音波センサー」にしましょう。
4. 上の移動ブロックを無限に前に進む設定にしましょう。
5. 下の移動ブロックをその場で 180 度旋回するように設定しましょう。
6. 最後に、作成したプログラムが無限ループするようにしましょう。
7. プログラムの名前を「reidai4-4」として NXT にダウンロードし、動作させてみましょう。

■問題1 ライントレーサー

これまでいろいろな処理について学んできました。今まで使ってきたブロックを組み合わせることによっていろいろなプログラムを組むことができます。

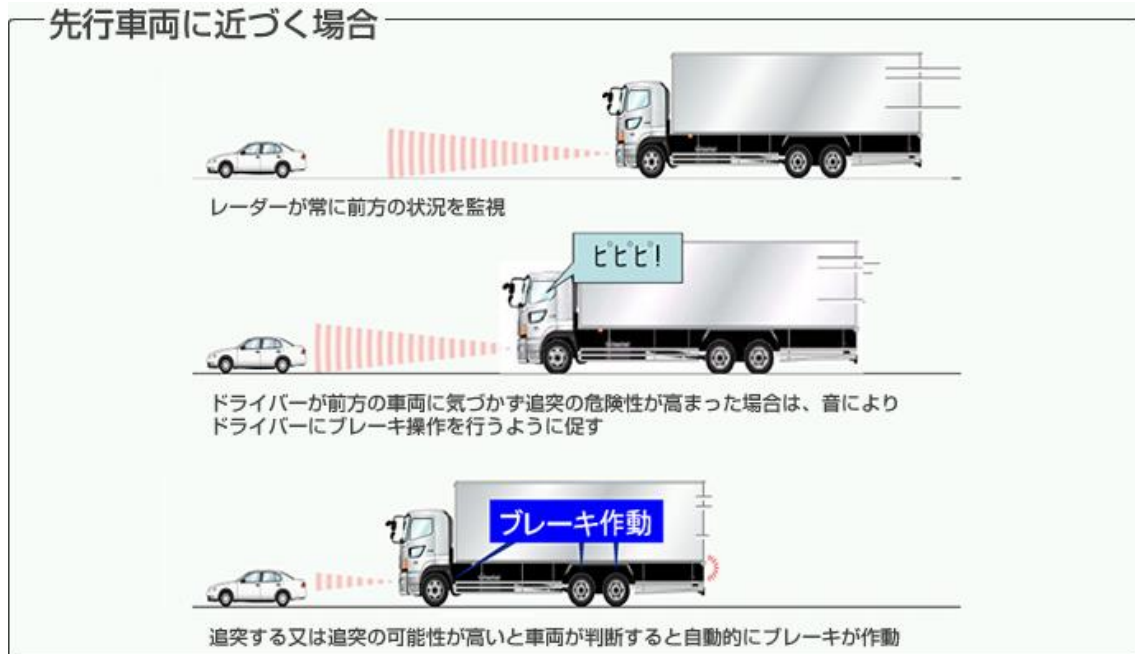
ここでは、床に描かれた黒い線をたどっていくプログラムを作ってみましょう。



1. ライン上をトレースする
2. ライン上をトレースするが、障害物にぶつくと 180 度旋回して戻る。
3. ライン上をトレースするが、障害物があれば避ける。

■問題2 衝突被害軽減ブレーキ

現在の車には、事故防止のために障害物に近づくと自動的にブレーキをかける機能が備わっています。この機能のことを衝突被害軽減ブレーキといいます。



今回は、衝突被害軽減ブレーキをプログラミングしてみましょう。

1. プッシュセンサーを押すと前進するが、障害物が近くにあると前進しない
2. プッシュセンサーを押したら前進→障害物が近づくと警告音を鳴らす→更に近づくと緊急停止
3. 障害物との距離により速度を変える(50cm より近くなれば遅く、30cm より近くなれば更に遅く、10cm)