



# 実践ロボットプログラミング

LEGO Mindstorms NXT で目指せロボコン!

WEB : <http://robot-programming.jp/>

著者 : 藤吉弘亘, 藤井隆司, 鈴木裕利, 石井成郎

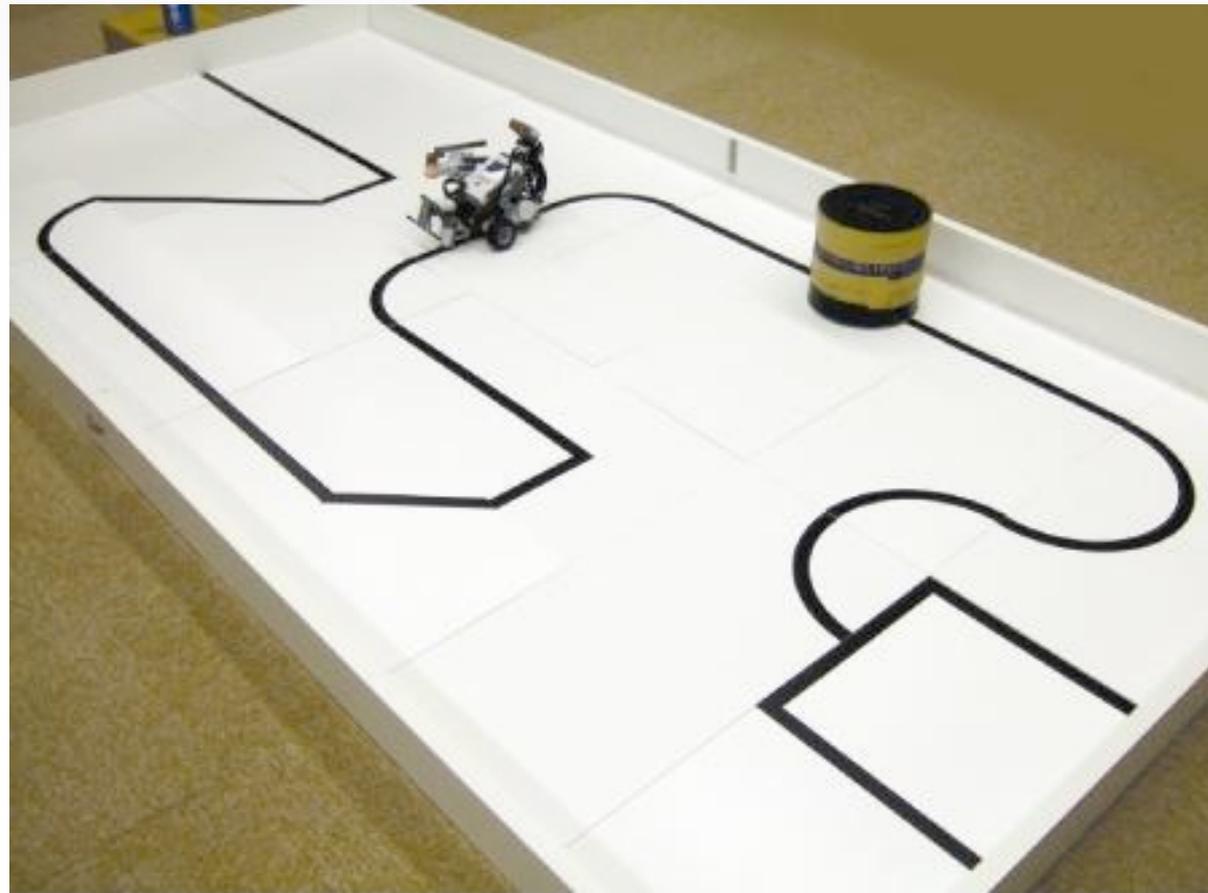
E-mail : [support@robot-programming.jp](mailto:support@robot-programming.jp)

## ■モデリング入門

## ■モデリングとは

## モデリングのイメージ

- ・ 大会コースが発表された！！！！



## モデリングのイメージ

- ・ 目標： 優勝するぞ！！！！ ← **すぐ決まる (^^)**
- 速く走るロボットとプログラム ← **すぐには作れない (一一)**

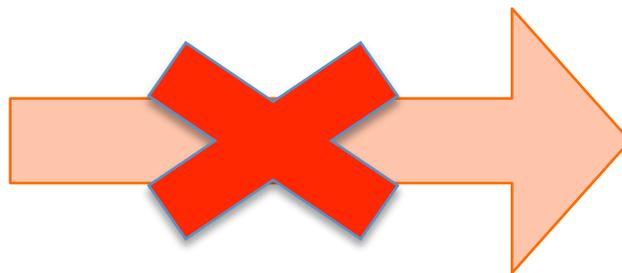


プログラム+ロボット

```
task main()
{
  SetSensorLight(IN_3);
  while(true) {
    if(Sensor(IN_3) > 40){
      OnFwd(OUT_C, 75);
      Off(OUT_A);
    } else{
      OnFwd(OUT_A, 75);
      Off(OUT_C);
    }
  }
}
```

## モデリングのイメージ

- ・ 目標： 優勝するぞ！！！！ ← **すぐ決まる (^^)**
- 速く走るロボットとプログラム ← **すぐには作れない (一一)**



プログラム+ロボット

```
task main()
{
  SetSensorLight(IN_3);
  while(true) {
    if(Sensor(IN_3) > 40){
      OnFwd(OUT_C, 75);
      Off(OUT_A);
    } else{
      OnFwd(OUT_A, 75);
      Off(OUT_C);
    }
  }
}
```





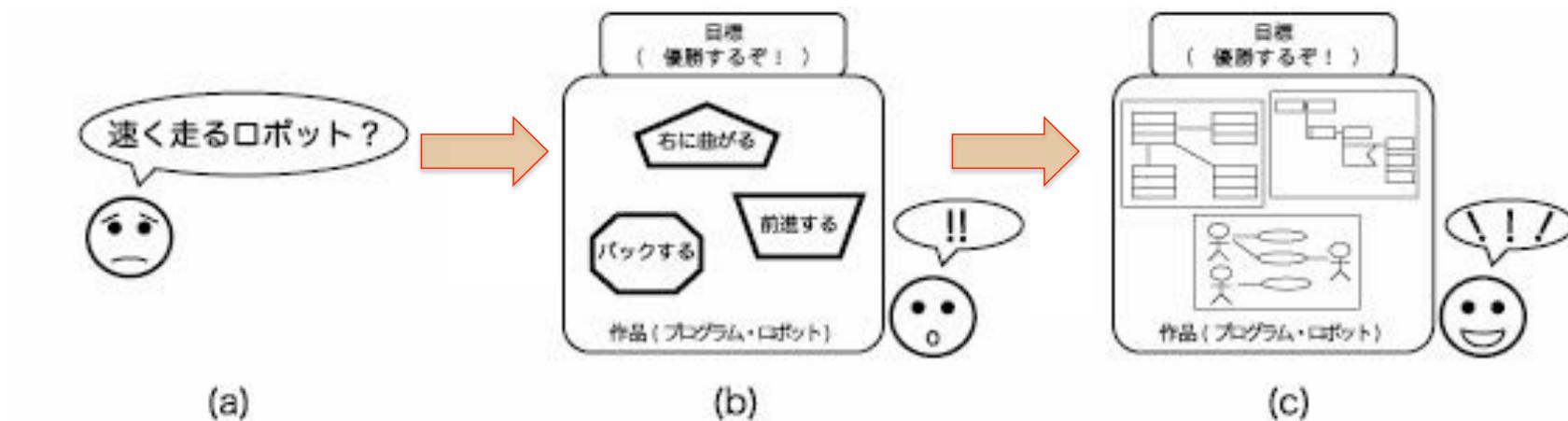
## モデリングのイメージ

- ・ モデリング
  - 実現したい目標を
    - ・ 具体化 可視化 詳細化



- 設計図を作ること！！

← わかりやすくなる (^^)



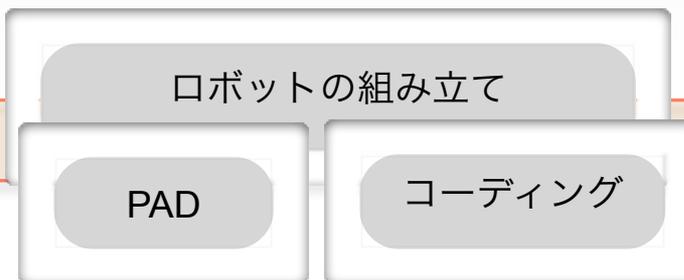
# モデリングのイメージ

## ・ モデリング

- 実現したい目標を 具体化 可視化 詳細化
- 設計図を作ること！！

やりたい  
こと?

← 遠い →  
(---)



実現!

優勝  
するぞ!



(b)



(c)

プログラム+ロボット

```
task main()
{
  SetSensorLight(IN_3);
  while(true) {
    if(Sensor(IN_3) > 40){
      OnFwd(OUT_C, 75);
      Off(OUT_A);
    } else{
      OnFwd(OUT_A, 75);
      Off(OUT_C);
    }
  }
}
```



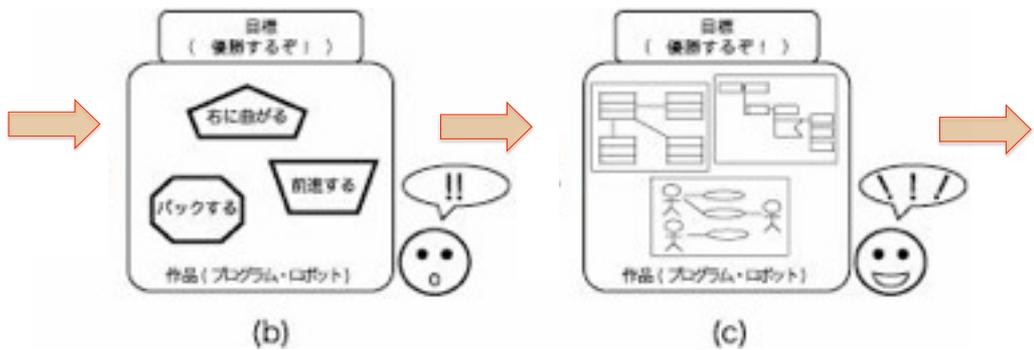
# モデリングのイメージ

- ・ モデリング
  - 実現したい目標を 具体化 可視化 詳細化
  - 設計図を作ること！！

やりたいこと?



優勝するぞ!



プログラム+ロボット

```
task main()
{
  SetSensorLight(IN_3);
  while(true) {
    if(Sensor(IN_3) > 40){
      OnFwd(OUT_C, 75);
      Off(OUT_A);
    } else{
      OnFwd(OUT_A, 75);
      Off(OUT_C);
    }
  }
}
```

## モデリングのイメージ

- ・ モデリング
  - 実現したい目標を 具体化 可視化 詳細化
  - 設計図を作ること！！

やりたい  
こと?

モデリング  
(設計図)

ロボットの組み立て

PAD

コーディング

実現!

プログラム+ロボット

```

task main()
{
    SetSensorLight(IN_3);
    while(true) {
        if(Sensor(IN_3) > 40){
            OnFwd(OUT_C, 75);
            Off(OUT_A);
        } else{
            OnFwd(OUT_A, 75);
            Off(OUT_C);
        }
    }
}
    
```

PLAN

計画をたてる

Do

計画を実行する

SEE

実行結果を  
評価する



# モデリングのイメージ

- ・ モデリング
  - 実現したい目標を 具体化 可視化 詳細化
  - 設計図を作ること！！



プログラム+ロボット

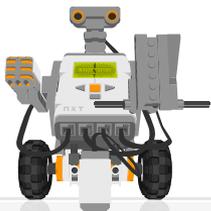
```

task main()
{
  SetSensorLight(IN_3);
  while(true) {
    if(Sensor(IN_3) > 40){
      OnFwd(OUT_C, 75);
      Off(OUT_A);
    } else{
      OnFwd(OUT_A, 75);
      Off(OUT_C);
    }
  }
}
  
```



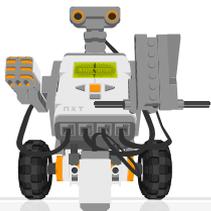
## ■コース攻略をモデリング





## コースモデリングの手順

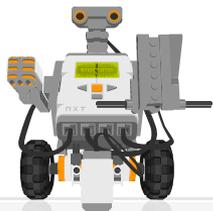
1. ルールの確認
2. 必要な機能（やりたいこと）の確認
3. 機能（やりたいこと）の内容と必要な情報を整理
4. 機能（やりたいこと）の詳細（実現方法）を記述
5. 機能と機能の関連を記述



## コースモデリングの手順

### 1. ルールの確認

2. 必要な機能（やりたいこと）の確認
3. 機能（やりたいこと）の内容と必要な情報を整理
4. 機能（やりたいこと）の詳細（実現方法）を記述
5. 機能と機能の関連を記述



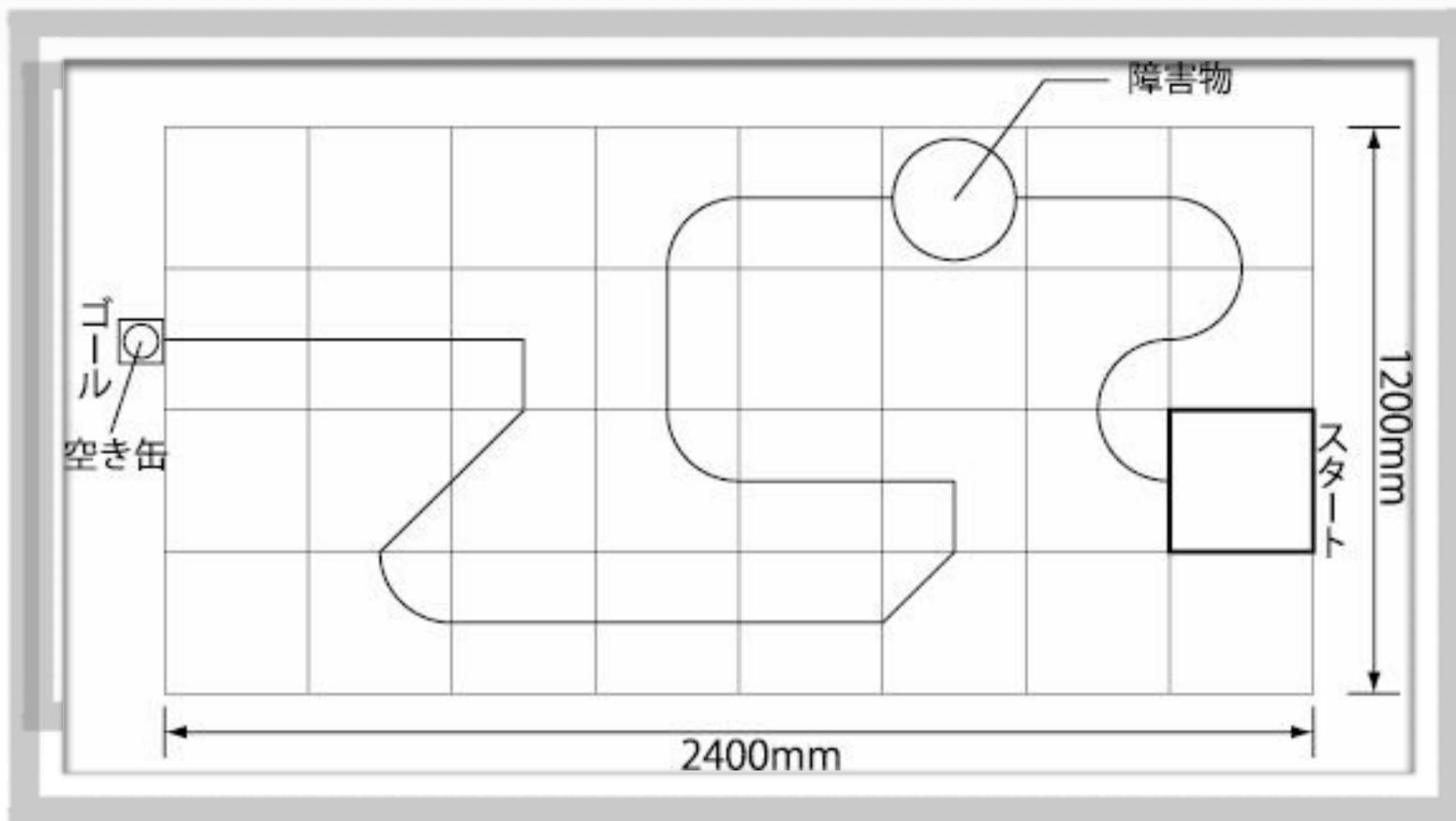
## コースモデリングの手順

1. ルールの確認
2. 必要な機能（やりたいこと）の確認
3. 機能（やりたいこと）の内容と必要な情報を整理
4. 機能（やりたいこと）の詳細（実現方法）を記述
5. 機能と機能の関連を記述



## 2. 必要な機能（やりたいこと）の確認

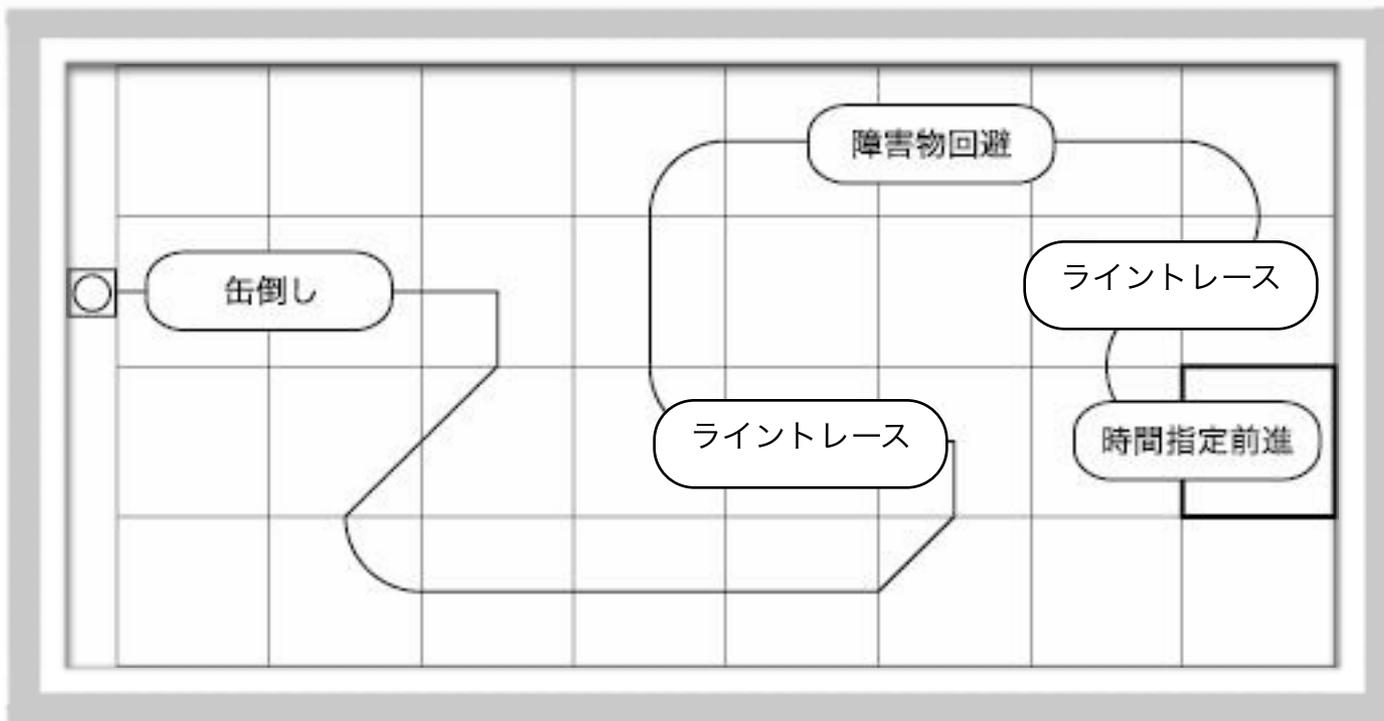
- どんな機能が必要だと思いますか？
- K君とU君の場合は. . . .

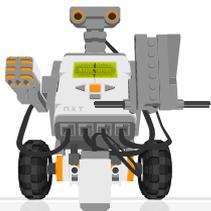




## 2. 必要な機能（やりたいこと）の確認

- ・ どんな機能が必要だと思いますか？
- ・ K君とU君の場合は. . . .





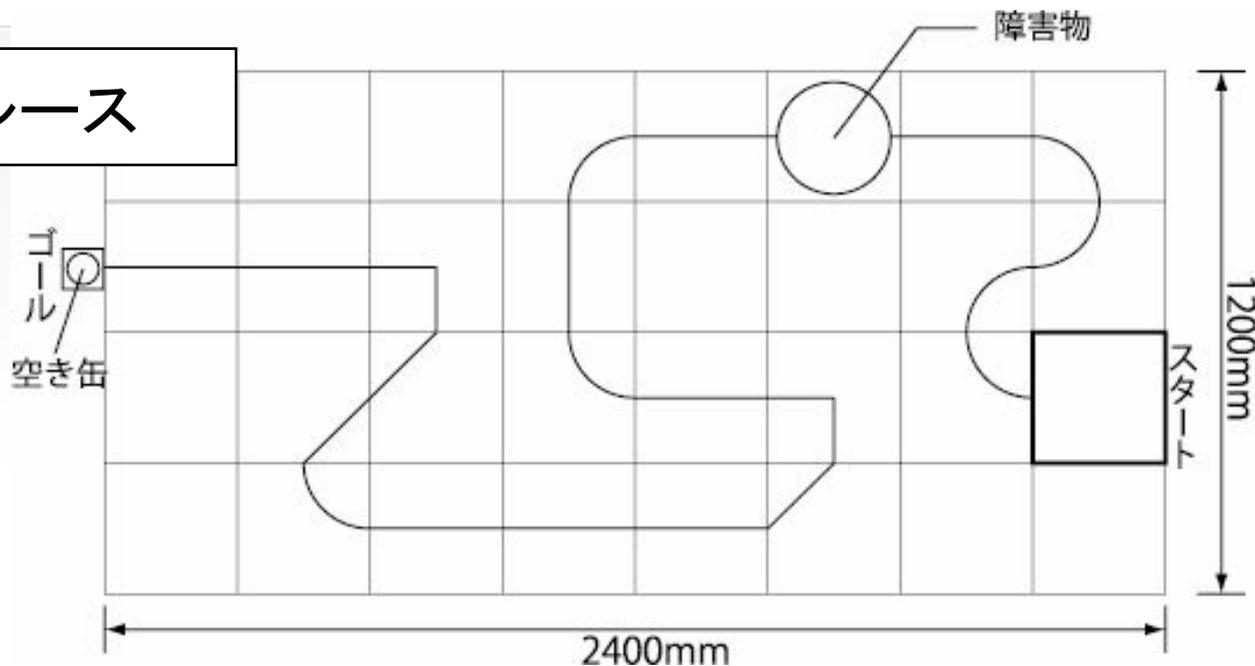
## コースモデリングの手順

1. ルールの確認
2. 必要な機能（やりたいこと）の確認
3. 機能（やりたいこと）の内容と必要な情報を整理
4. 機能（やりたいこと）の詳細（実現方法）を記述
5. 機能と機能の関連を記述



### 3. 機能（やりたいこと）の内容と必要な情報の整理 P.154

#### ライトレース



B. ライトレース：黒線を追跡して走行する

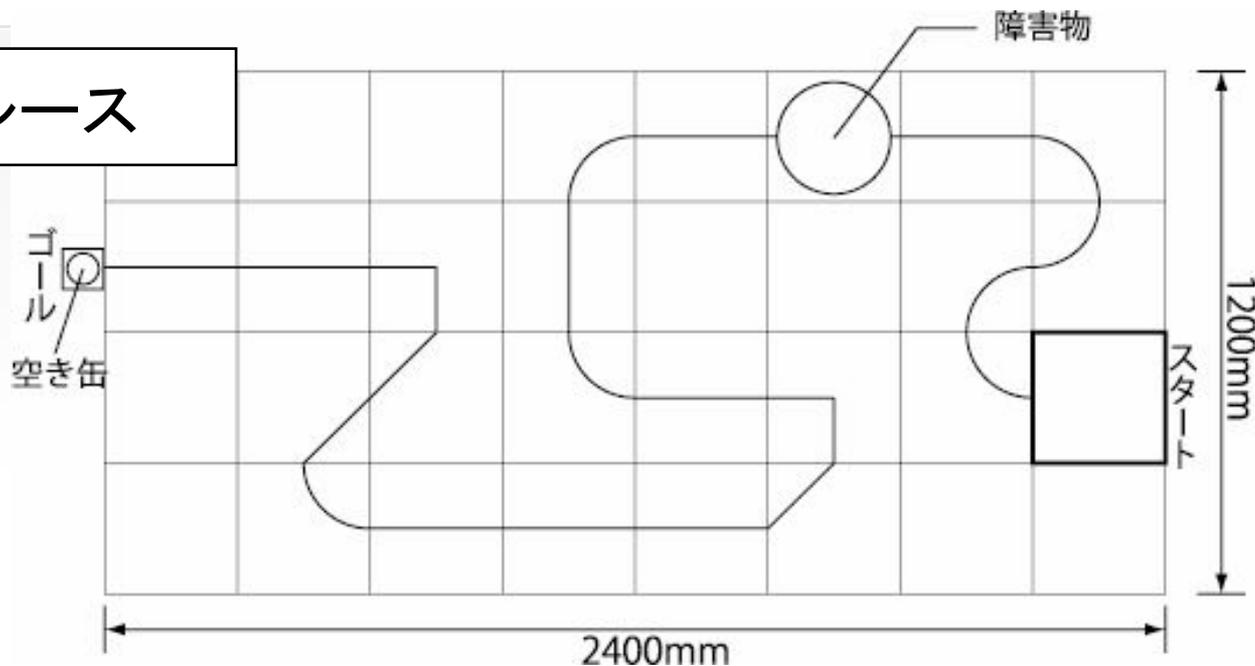
#### 1. 光量を測定する

機能の内容(実際にさせたい動作)  
回転する (動詞)

必要な情報や量  
光量 (名詞)

### 3. 機能（やりたいこと）の内容と必要な情報の整理 P.154

## ライトレース



B. ライトレース：黒線を追跡して走行する

#### 1. 光量を測定する

##### ① 光量

機能の内容(実際にさせたい動作)  
回転する (動詞)

必要な情報や量  
光量 (名詞)





### 3. 機能（やりたいこと）の内容と必要な情報の整理 P.155

- すべての機能が整理できたら 機能モデル を作成しよう！

機能モデル	作業日	月	日
学籍番号 _____	チーム番号: _____		
氏名 KC _____	チーム名: _____		
学籍番号 _____			
氏名 US _____			

---

**コース攻略に必要な機能と情報**

A. 時間指定前進: 決められた時間前進走行する

1. 前進する
2. 決められた時間が経過したら前進をやめる

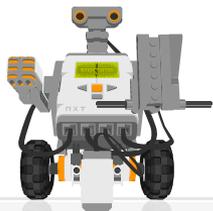
①時間    ②モータの力

B. ライントレース: 黒線を追跡して走行する

1. 光量を測定する
2. 光量に応じて右回転, 左回転する

①光量    ②モータの力

以下 P.155 参照



## コースモデリングの手順

1. ルールの確認
2. 必要な機能（やりたいこと）の確認
3. 機能（やりたいこと）の内容と必要な情報を整理
4. 機能（やりたいこと）の詳細（実現方法）を記述
5. 機能と機能の関連を記述



## 4. 機能（やりたいこと）の詳細（実現方法）を記述 P.156

- ・ 各機能ごとに

機能の実現のために必要なことを確認して行こう！

0. 絵を描きながら確認するといいよ！（イメージスケッチ）

1. 何を実現する機能か？（機能の概要）

2. その機能の動作が始まる時の条件は？

（開始条件）

3. 動作の内容とその順番は？（機能の流れ）

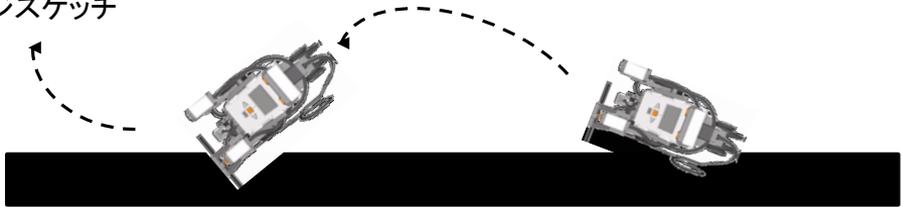
4. その機能の動作が終わるときの条件は？

（終了条件）

- ・ K君とU君の場合は. . . .

# 4. 機能（やりたいこと）の詳細（実現方法）を記述 P.157

- 確認した内容を詳細モデルに記述しよう！

詳細モデル		作業日	月	日
学籍番号	氏名	KC		
学籍番号	氏名	US	チーム番号	チーム名
機能名	ライトレース			
機能の概要	黒線を追跡して走行する			
開始条件	時間指定前進が終わる			
機能の流れ	<ol style="list-style-type: none"> <li>1. 光量を測定する(B-①)</li> <li>2. 黒かどうかの判断をする(B-①)</li> <li>3. 判断の結果に応じて, 右回転, 左回転する             <ol style="list-style-type: none"> <li>3a. 黒だったら右回転する(B-②)</li> <li>3b. 黒でなかったら左回転する(B-②)</li> </ol> </li> </ol>			
終了条件	フィニッシュエリアに達する			
<p>イメージスケッチ</p> 				

- このモデルからPADが書けるかな？ P.160参照

## コースモデリングの手順

1. ルールの確認
2. 必要な機能（やりたいこと）の確認
3. 機能（やりたいこと）の内容と必要な情報を整理
4. 機能（やりたいこと）の詳細（実現方法）を記述
5. 機能と機能の関連を記述

## 5. 機能と機能の関連を記述

- ・ 機能と機能の関連を整理しよう！
- ・ K君とU君の場合は. . . .
  - コースに沿って、機能の動きをイメージ  
(処理の流れから)
  - 機能と機能の移り変わりをイメージ  
(機能中心)



## 5. 機能と機能の関連を記述

- すべての機能の関連を考えて 関連モデル を作成しよう!

関連モデル 作業日 月 日

学籍番号 \_\_\_\_\_ チーム番号: \_\_\_\_\_

氏名 KC \_\_\_\_\_ チーム名: \_\_\_\_\_

学籍番号 \_\_\_\_\_

氏名 US \_\_\_\_\_

機能中心の関連モデル

機能の関連を実際の流れに沿って図示

ライトレース	
機能の概要	黒線を追跡して走行する
開始条件	時間指定前進が終わる
機能の流れ	1. 光量を測定する(B-①) 2. 黒かどうかの判断をする(B-①) 3. 判断の結果に応じて、右回転、左回転する 3a. 黒だったら右回転する(B-②) 3b. 黒でなかったら左回転する(B-②)
終了条件	フィニッシュエリアに達する

時間指定前進	
機能の概要	決められた時間前進する
開始条件	略
機能の流れ	略
終了条件	指定された時間が経過する

障害物回避	
機能の概要	障害物を回避してコースに復帰する
開始条件	障害物に近づく(C-①)
機能の流れ	1. 右回転(C-②, C-③) 2. 左回転(C-②, C-③) 3. 白黒の判断をする 3a. 黒だったら機能終了(C-④) 3b. 黒でなかったら前進する(C-④)
終了条件	コースに復帰する

処理中心の関連モデル  
はP.158参照

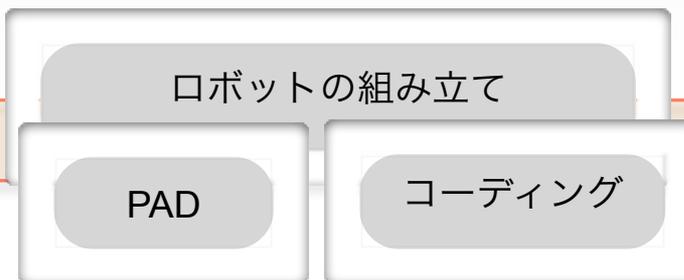
# モデリングのイメージ

## ・ モデリング

- 実現したい目標を 具体化 可視化 詳細化
- 設計図を作ること！！

やりたい  
こと?

← 遠い →  
(---)



実現!

優勝  
するぞ!



(b)



(c)

プログラム+ロボット

```
task main()
{
  SetSensorLight(IN_3);
  while(true) {
    if(Sensor(IN_3) > 40){
      OnFwd(OUT_C, 75);
      Off(OUT_A);
    } else{
      OnFwd(OUT_A, 75);
      Off(OUT_C);
    }
  }
}
```



# モデリングのイメージ

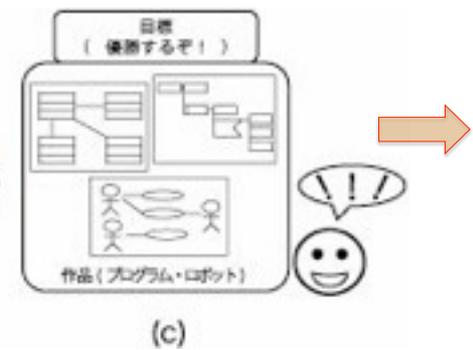
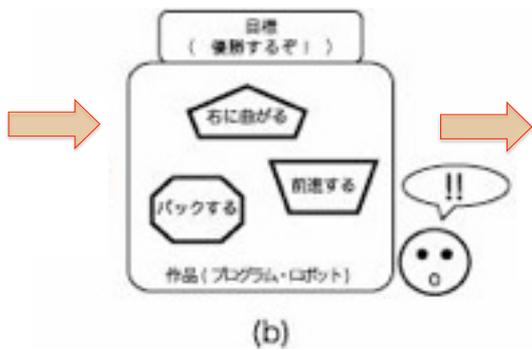
- ・ モデリング
  - 実現したい目標を 具体化 可視化 詳細化
  - 設計図を作ること！！

やりたいこと?



実現!

優勝するぞ!



プログラム+ロボット

```
task main()
{
  SetSensorLight(IN_3);
  while(true) {
    if(Sensor(IN_3) > 40){
      OnFwd(OUT_C, 75);
      Off(OUT_A);
    } else{
      OnFwd(OUT_A, 75);
      Off(OUT_C);
    }
  }
}
```





■演習問題9-1. に挑戦！

■モデルの評価方法を参考に. . . P.159