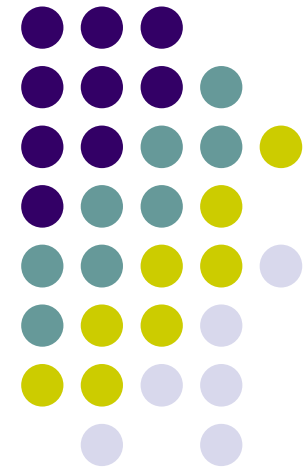


自律型ロボットを用いた プログラミング教育 ～ ロボカップジュニアを通しての学生教育～

東京都立産業技術高等専門学校
ものづくり工学科
黒木 啓之





背景

- 言語プログラミングを苦手とする学生の増加
 - 言語が理解できないのではなく、与えられた課題を解く方法がわからない アルゴリズムがわからない = 論理的思考能力の欠如(朝日新聞記事)
- 加減乗除するような課題ではつまらない
 - 授業に対するモチベーションの低下に繋がる
- ロボット講座が至るところで実施される
 - 理科離れ・工学離れが理由
 - 老若男女が興味を持ち、どの講座也大盛況



- 自律型ロボットにはプログラムが必要
 - プログラムをした結果がロボットの動作として反映
 - 生徒・学生には興味を持たせながら実施が可能
 - (付属の)プログラミングツールはアルゴリズムを教育するには最適

本発表の内容

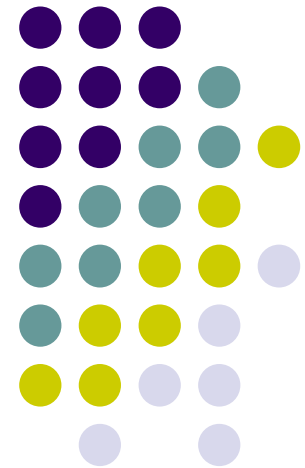
- 自律型ロボットを用いたプログラミング教育を実践している例について紹介
- 「ロボカップジュニア」を通じたプログラミング教育も紹介



自律型ロボットを用いる利点

- 学生のわからないところ – if文やfor文
= 論理的思考の部分
- ロボットの動作には
 - 壁にぶつかったら回避 条件分岐
 - ロボットをずっと動かす 繰り返し(無限ループ)
 - プログラムの結果がロボットの動作として現れる
- 楽しい目標を設定できる
 - サッカーをする・ライトレースをするなど, おもしろい題材となっている(ダンスなどは女の子にとっても適当な題材) モチベーションをかなり上げられる

自律型ロボットを用いた プログラミング教育



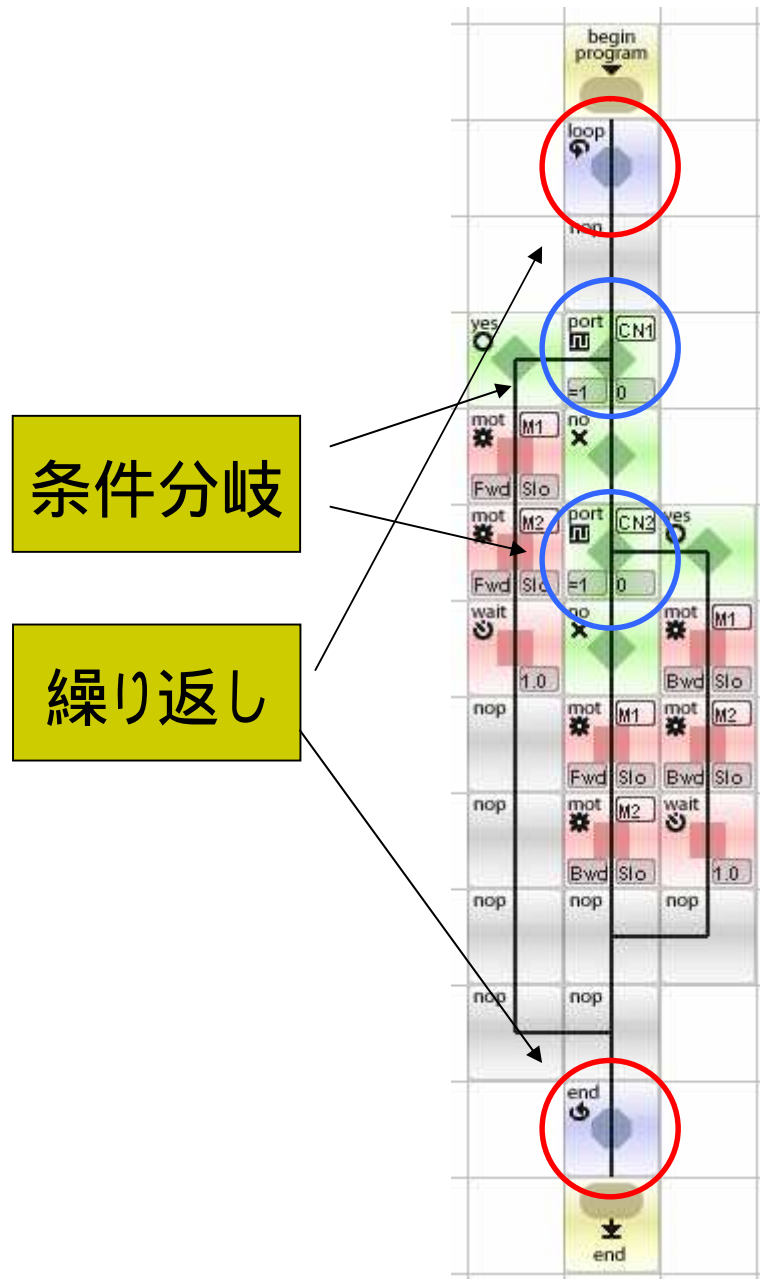


背景・目的

- 自律型ロボットはプログラムで制御
- プログラミング言語には大きく分けて2種類
 - 言語型 (BasicやC言語など)
 - ブロック型 (タイル型とも呼ばれる)
 - ↳ フローチャートそのもの
- 2つのツールを上手く使って学習
 - ブロック型 アルゴリズムを教育 ↳
 - 言語型 プログラミング言語を教育

関連付けをし
上手く移行

ブロック型プログラミング言語



条件分岐

繰り返し

- フローチャートそのものの構造
- アルゴリズム教育には最適

ロボット紹介ーサッカーロボ915

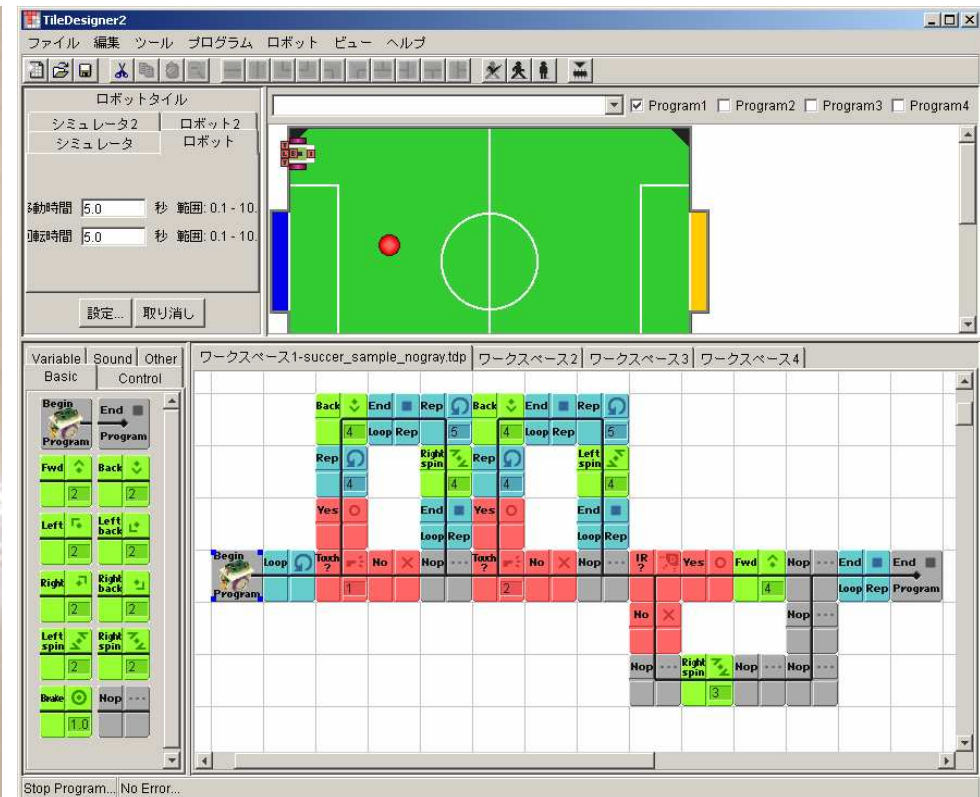
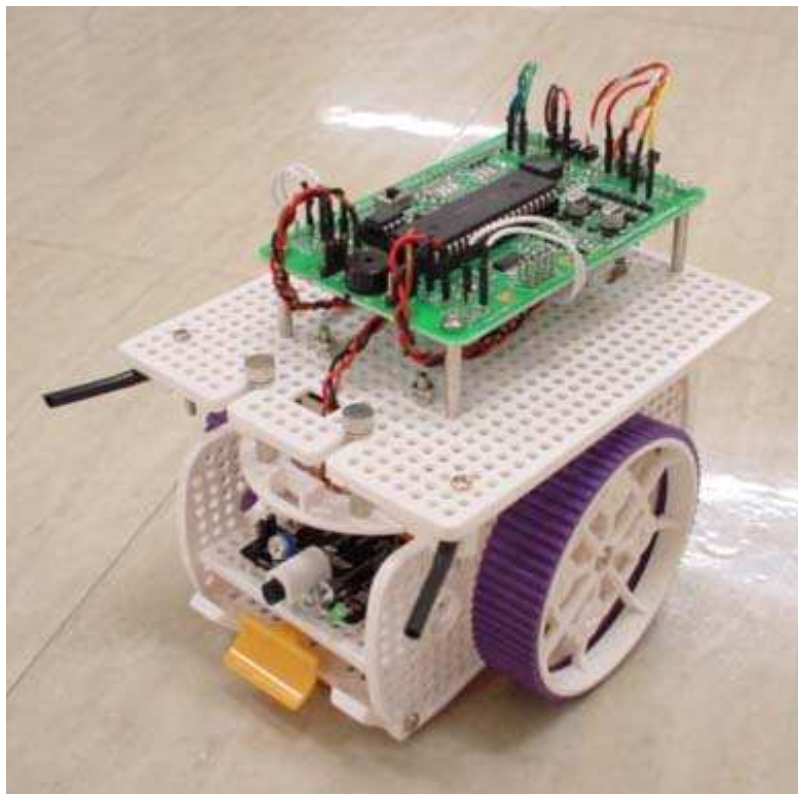


図2 SOCCER ROBO 915とTileDesigner

ロボット紹介ーレゴマインドストーム

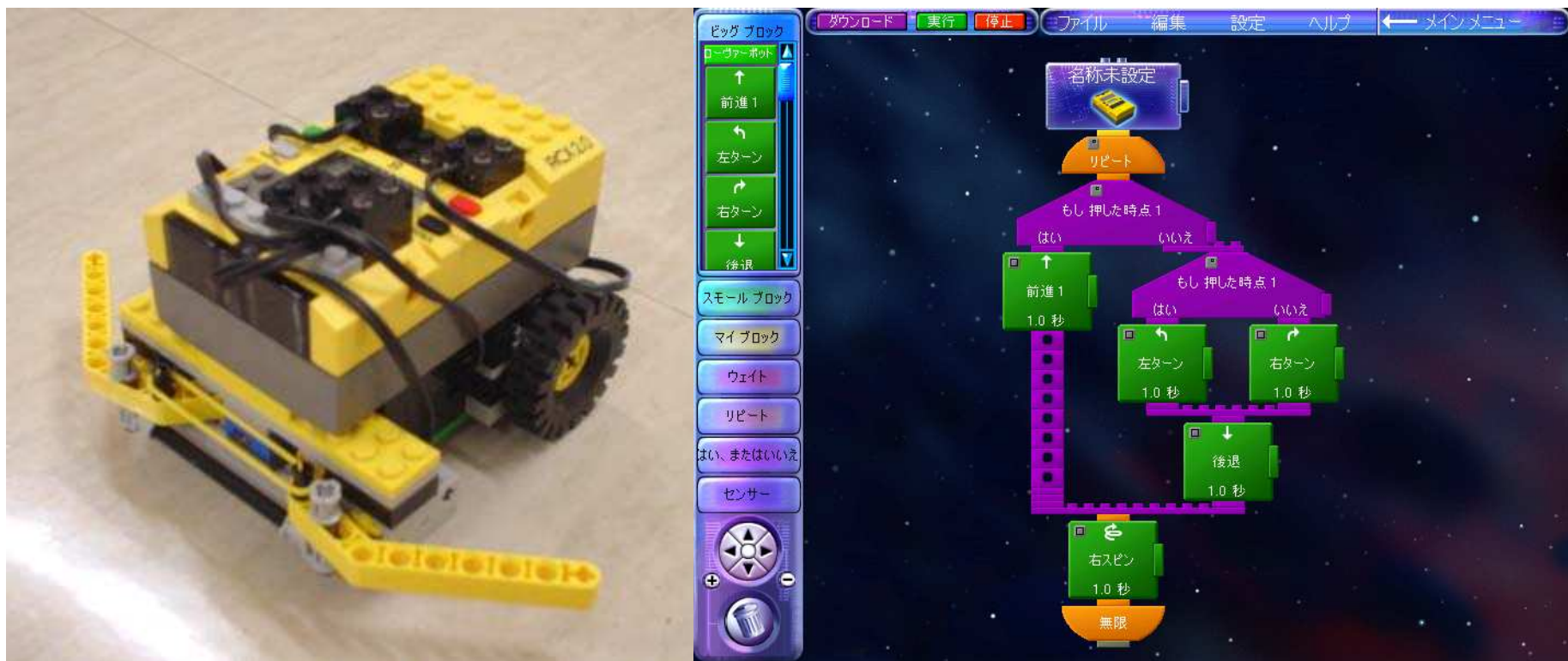
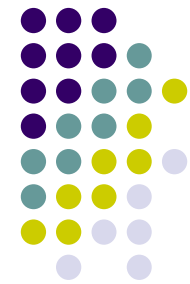


図3 MINDSTORMS ROBOTICS INVENTION

ロボット紹介ーロボデザイナー

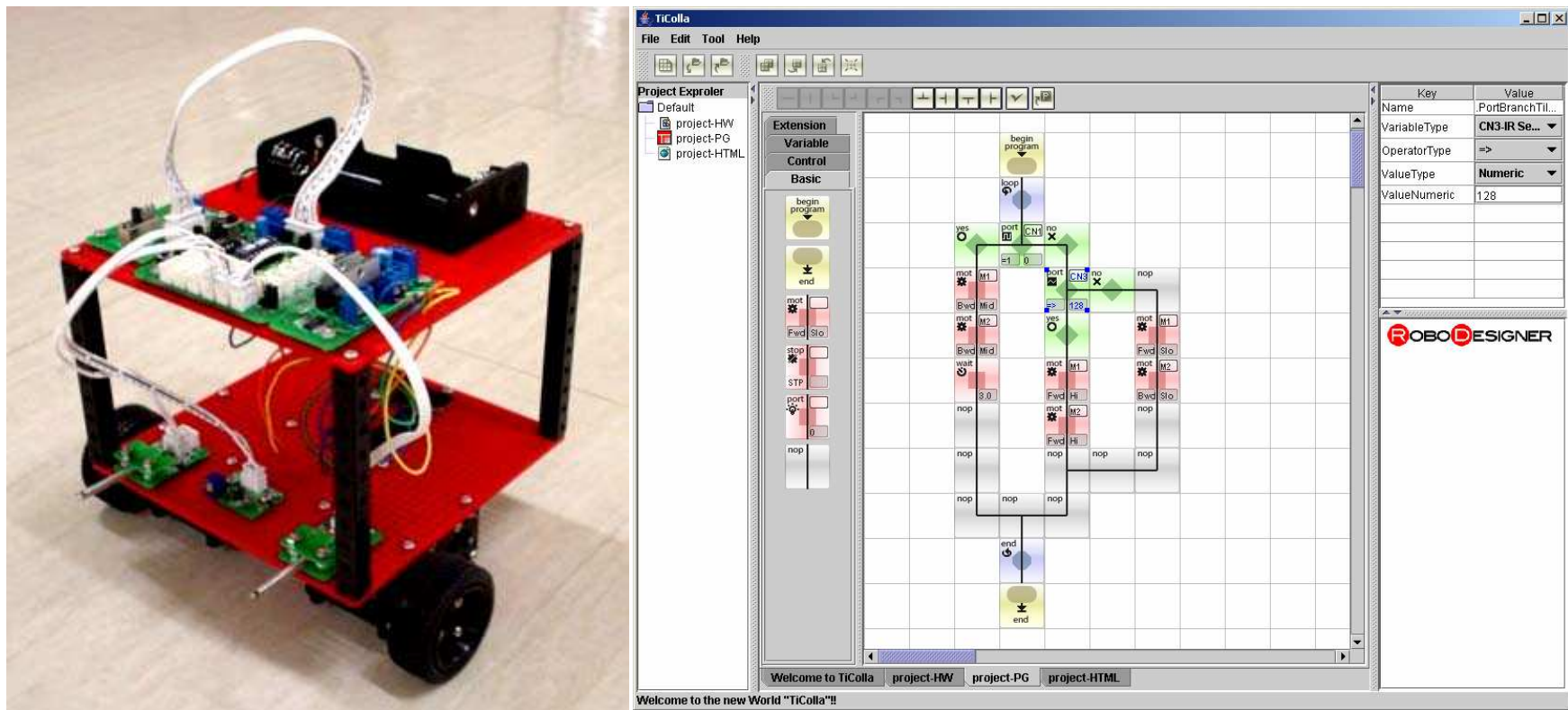
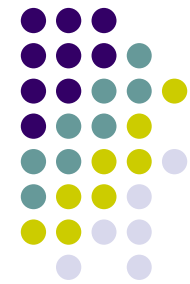
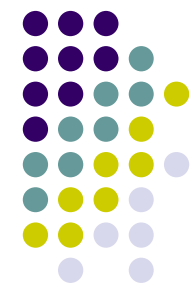


図4 ROBO DESIGNERとTiColla



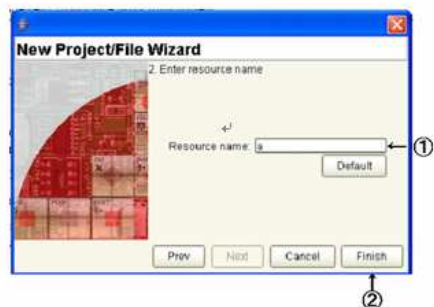
指導書の抜粋—独学用に特化—

『TiColla』の準備

プログラムを起動したら、最初に一番左上にある『File』にマウスのアイコンをあわせるとメニューが開くので、その中から『New』を選びクリックしてください。(右図参照)



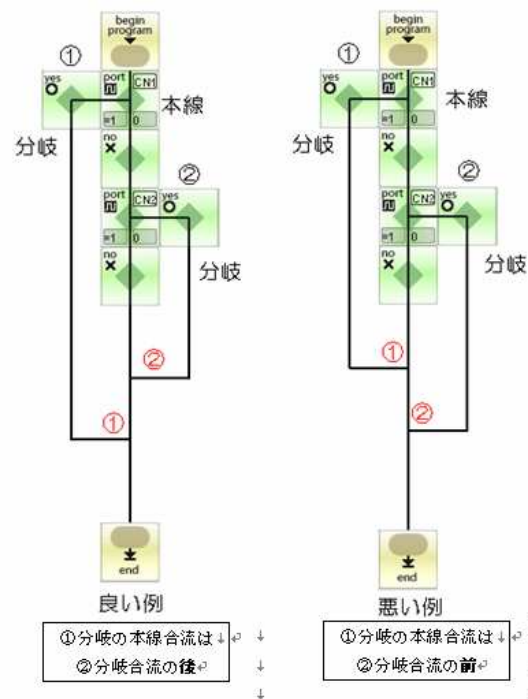
そうするとこのような画面が現れるので、最初に①をクリックして選択し(文字が青く囲まれた状態にする)、次に②の『Next』をクリックします。



さらに画面が変わるので①のテキストボックスに半角英数字でファイル名(！ 必須 上の例では“a”！)を入力し、②の『Finish』をクリックしてください。

！ 注意 !!

『TiColla』ではプログラムが1つの Begin から始まるとき、必ず1つの end に入れないといけないという決まりがあります。『port タイル』を用いたことによって、1つの end タイルに戻すためにはプログラムの本線を決め、分岐させたものを本線に戻す作業が必要となります。



上の図を見てください。

重要なのは①分岐と②分岐の関係です。本来、②分岐は①分岐で no 側に分岐した場合にだけ現れます。ですから①分岐で yes 側に分岐した場合、②分岐の存在は無視されるはずですが、右側のような書き方をすると『①分岐で yes に分岐したはずなのに、no 分岐側にあるはずの②分岐に合流』という矛盾が起こります。そのためプログラムが正しい判断をすることが不可能になります。このように『考え方に矛盾が生じると様々なエラーが起こる』というのは C 言語や C++ 言語などのプログラミング言語でも言えることです。



指導書の特徴

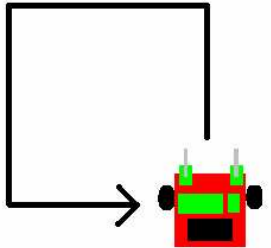
- 独学形式での授業
 - 自分の理解のペースで授業を進めていける
 - 「他人に聞いてはいけない」という制約も設ける
 - ↪ 自分で理解することで理解を深めさせる
- 例題 応用問題の形式
 - 生徒・学生に考えてもらうため
 - 近年、この形での実施が不可能に
- 図は多用するが文章も残す
 - 図が多いと図しか見ない
 - 「読む力」の低下も懸念

アルゴリズム用教科書



繰り返し動作をさせよう

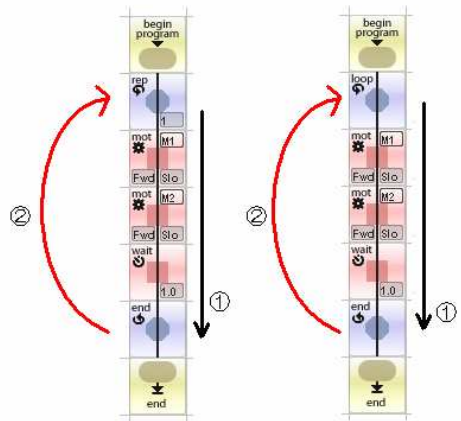
<問題2>
下の図のようにロボットが動くようにプログラミングしなさい。(1周して戻ってくるプログラム。ターンは90°にすること。)



！ヒント！
ある二つの動作を4回書けばよい

← 問題などにも図

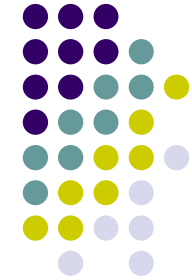
問題はできましたか？難しいことはないのです。前進⇒左回転を4回繰り返すだけです。でも全てのタイルを並べるのは大変だったでしょう？ですから先ほどの問題のように一定の動作を繰り返すようなプログラムを書く場合には『繰り返し』を命令するタイルを使います。繰り返し命令のタイルには rep タイルと loop タイルの2種類が用意されています。



← 実際の例

図2. アルゴリズム用教科書例

プログラミング言語用教科書



5のプログラムでもまた新しいコマンド『if』と『else』が出てきます。これは一般的にセットでif文と呼ばれ、『TI Colla C』に限らずC言語では極めて重要な命令文のひとつなので、説明をよく読んで理解するようにしてください。

```
if(条件)
{
  ...
  命令文
  ...
}
else
{
  命令文
}
```

ifとは『もし指定された条件を満たしたら、中括弧の中に書かれた命令を実行する。』と言うものです。書き方としては、まずifの横にある括弧内に条件を記入します。その後、中括弧内に条件を満たしたときに行わせたい命令を書き込んでいきます。

elseとは『ifに書かれた条件を満たさなかった場合、中括弧の中に書かれた命令を実行する。』というものです。ifとelseはセットですから、書き方は左側にある例のように、まずifを書いてから、下にelseを書きます。(ifとelseはセットですが、場合によってはelseを省略することも可能です)

今回のプログラム例では、ifの条件文に『ch1が“1”になった時(つまりタッチセンサーに反応があったら)』という指定をし、命令文には『後退した後、その場で回転』という書き込みました。そしてelseの命令文には『直進』と書きました。実際にそうなっているか、前のページに戻って確かめてください。

← 用法・用例を詳しく説明

このコマンドはタイルプログラムの



と同じ働きをします。

← タイルと対比

<問3>

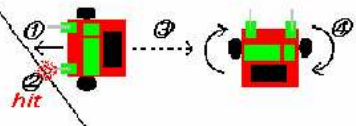
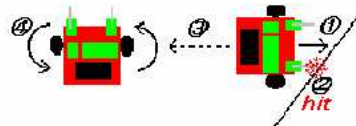
右の図のように

- ・『右側のタッチセンサーが壁に当たった場合はバックして左回転』
- ・『左側のタッチセンサーが壁に当たった場合はバックして右回転』

するようなプログラムを書きなさい。タッチセンサーに反応がないときは常に直進の動作をさせなさい。

！ヒント！

- ・ch1だけでなくch2も使うこと。



← 問題にも図



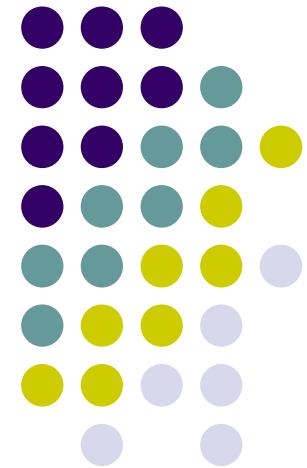
実施結果

- ロボデザイナーで実施
- アルゴリズムの理解も向上
- 楽しく授業ができたとの感想

問題点・課題

- 安価なロボットを導入しがち
- トラブル多い でもそれも学習の一環
- ただしそれは「故障」としか認識されずモチベーションを下げる要因になるだけ

ロボカップジュニアを通じた プログラミング教育





ロボカップとは

- ロボット工学と人工知能の融合・発展のため
- 自律型ロボットによるサッカーが主な題材
- 2050年に人間の世界一チームに勝つのが目標
- 他にレスキュー・PCによるシミュレーション・次世代を担う子供達が参加するジュニアがある
- ジュニアには、サッカーの他、レスキュー・ダンスが存在
- 世界大会まである(今年は大トランタ)
- 大学2年生まで出場でき、中学2年生を境にプライマリとセカンダリに分かれる



目的・方法

- さらに積極的にプログラミングを行わせるには別のモチベーションが必要 競争の原理
- 勝敗を決するような競技を通して「勝ちたい」というモチベーションを持たせる
 - ↳ 積極的にプログラミングするきっかけ
- 「ロボカップ部」に所属する初心者を対象
- アルゴリズム教育よりプログラミング教育に重点を置く ブロック型では勝てないことも身にしみてわかるので言語を積極的に学習

結果

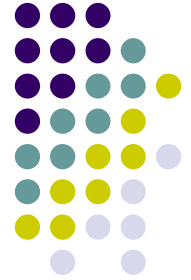


- プログラミング言語教育に対する効果
 - 「勝利をする」という目標からより良いプログラムを書こうという努力が生まれる プログラミング学習が進む
 - 限られたメモリを使って如何に思い通りのプログラムを実現するかなどの技術も習得
 - 学生自身のプログラム能力が向上したと感じている
- 学生の生活指導に対する効果もある



まとめ・課題

- 自律型ロボットを利用したプログラミング教育は
 - 生徒・学生達のモチベーションを向上させるには最適
 - アルゴリズム教育を行ってから言語教育を行う, という方法をとれる
 - クラブ活動などにも(教育的に)応用可能
- ロボットが対象なので遊び感覚が抜けない
- 「例題 応用問題」形式の崩壊
 - 例題はできる(コピーするだけなので)
 - 例題から簡単に類推できる応用問題ができない
 - 頭を使って学習(行動)を行っていない?



将来的に

- 組み込み技術をロボットで教える
 - 既にPIC, FPGA, CPLDを使った教育カリキュラム案を作成
 - 現在, DSPを用いた教育カリキュラムを作成中
(ニューラルネットワーク教育を中心に)