

No.14 「Squeak で学ぶプログラミング(5)」

年 組 番 ID

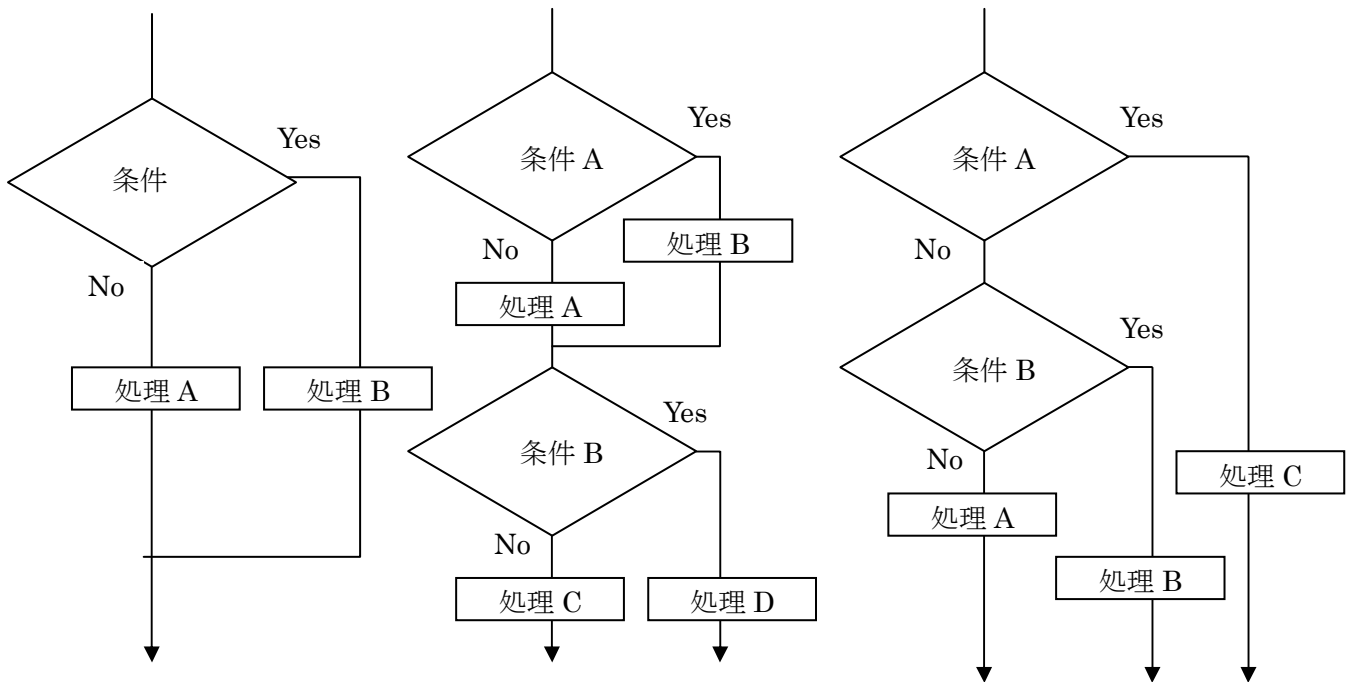
氏名

前回のプリントから車がコース内を走るようなプログラムを考えている。これを一般に「ラインレース」といい、実用として、工場内で床に描かれたラインの上を荷台車が自動的に動いている。この場合、光センサーがライン上とライン外の反射率を比較してラインを検知している。今回は、さらに動きの精度を高めるため、センサーが2つの場合を考えていく。

(8) 場合分け【条件分岐】の応用

プログラミングの世界では、「流れ図 (フローチャート)」という図を使い、上から下へとプログラムの流れを表します。とくに条件分岐にはひし形の記号と、Yes と No の分岐線を使います。

また「進む」とか「回る」という処理を、長方形で表します。



条件分岐

直列の条件分岐

入れ子の条件分岐

(9) 道路から外れたことの判定

「道路から外れた」ということを判定するためには、
ア. センサーが道路外の色に触れている → Yes
イ. センサーが道路の色に触れている → No
という、2通りの方法がある。

道路に様々な色を使ったり、中央線を使ったりしていくと、アの判定の方が楽になります。

このプリントの作業では、アの判定、つまり、「道路外の色に触れている → Yes」を「道路から外れた」として使います。

(10) センサーが左右2つの場合

※「9999 ライトレース複センサー」で保存する。

①直列の条件分岐による方法

車「複センサー直列の条件分岐」

テスト 車の（左センサーの色）モーフはその色に触れているか（道路外の）色

はい

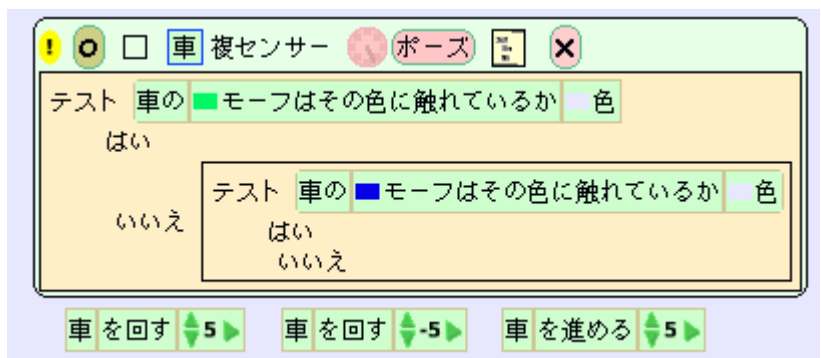
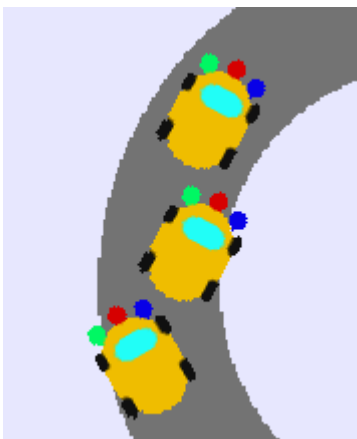
いいえ

テスト 車の（右センサーの色）モーフはその色に触れているか（道路外の）色

はい

いいえ

②「入れ子」の条件分岐による方法



①と②の考え方の手順（アルゴリズム）は、どちらが優れているでしょうか？ また理由は？