スクイークで学ぶ
ゲームプログラミング

～世界聴診器をめぐる大冒険～

http://mts-j.hiho.jp/

デジタルハリウッド大学大学院
みんなでたのしくスクイーク・プロジェクト

スクイークで学ぶゲームプログラミング

～世界聴診器をめぐる大冒険～

2007年2月

デジタルハリウッド大学大学院
みんなでたのしくスクイーク・プロジェクト

はじめに

これからの子ども達に必要となるのは、いかに社会が変化しようと、自分で課題を見つけ、自ら学び、考え、行動し、問題を解決することができる資質や能力だといわれています。 また、他者と協調したり、他者を思いやる心、感動する心なども豊かな人間性を育む上では大切でしょう。

私たちが行っているプロジェクト活動（ワークショップや教材制作など）の目的のひとつは、子ども達がこうした「生きる力」を育むために、教育現場において情報通信技術（ICT）が正しく活用され、より創造性あふれる教育活動が促進されるように、様々な教育関係者の皆様をご支援することにあります。

今回、私達はこうした目的を実現するために、「スクイーク」と「世界聴診器」という教育用のソフトウェア／ハードウェアを利用しました。スクイークと世界聴診器は、子ども達が自由な発想で作品を作る過程において、自ら考えた課題に対して積極的に関わり、問題を発見し解決していくプロセスを体験することができる、すばらしい教育ツールです。

スクイークは、「描いた絵を自由に動かす」というシンプルなコンセプトを通じて、科学や数学の基礎的な概念を理解するとともに、創造力や論理的思考能力などを育むことができるといわれています。また、音や光、電圧などをパソコンに取り込むことができる「世界聴診器」をスクイークと一緒に活用することで、様々な科学実験や電子工作を行い、ハードウェアとソフトウェアの基本的な仕組みを学ぶこともできます。

将来、コンピュータ・サイエンティストやスーパー・プログラマ、デジタルコンテンツ・クリエータなどを目指すたくさんの子ども達が、このテキストやワークショップへの参加を通して、「大切な何か」を感じてくれることを心から願っています。

デジタルハリウッド大学大学院
みんなでたのしくスクイーク・プロジェクト
スタッフ一同

このテキストの使い方

このテキストは、スクイークの基本操作をある程度知っている人を対象に書かれています。ワークショップの補助教材や復習用としてご活用ください。

1 ワークショップ中の補助教材として

このテキストには、スクイークや世界聴診器を活用したたくさんのプログラミングテクニックが掲載されています。ワークショップで自分なりの作品を作るときに、必要に応じて参照してください。
たのしい作品をつくるためのヒントが見つかるかもしれませんよ！

2 ワークショップの復習用に

ペイントツールやハロ、ビューワには、ワークショップで覚えたこと以外にも、たくさんのたのしい機能が隠されています。このテキストを参考に、ワークショップで学んだことを復習してください。
スクイークを使って、いろいろなことを試してみてみましょう！

3 プログラムの仕組みを考える

このテキストに掲載しているプログラミングテクニックでは、その仕組みや考え方を詳しく解説してあります。何をどんな風に組み合わせると、どういう動きをするのか、プログラムの仕組みをじっくり考えましょう。
こうしたプログラミングテクニックを参考に、もっとすごいプログラムをたくさん作ってみましょうね！

スクイークとは

スクイーク(Squeak)は、「パーソナルコンピュータの父」といわれているアラン・ケイ博士が中心となって開発を進めているパソコン用のソフトウェアです。1970年代前後に開発されたオブジェクト指向プログラミング環境である「Smalltalk」をベースとして、1996年にWeb上に無償で公開さ れました。現在でも多くのプログラマの支持を受け、日々機能の改良が行われています。

詳しくは、ホームページをご覧ください。

http://mts-j.hiho.jp/

世界聴診器について

世界聴診器は、音や光、電圧などをパソコンで取り扱うことができる、スクイークと一緒に使うたのしいアイテムのことです。情報処理推進機構(IPA)が実施した平成15年度未踏ソフトウェア創造事業のプロジェクトとして阿部和広氏と林徹也氏が共同開発しました。光センサ、温度センサ、電圧センサからの入力信号を音の変化に変えて出力するハードウェアと、パソコンのマイク端子から入力された音の変化をスクイーク上で数値(音の周波数)として扱うための“世界聴診器モーフ”と呼ばれるソフトウェアから構成されています。
世界聴診器を使うには、専用のスクイークをダウンロードする必要があります。詳しくは下記の公式サイトをご覧ください。

世界聴診器をめぐる冒険　http://swikis.ddo.jp/WorldStethoscope/2

目次

第1章　スクイークの基本操作

1. インストールと起動
2. 絵を描く(ペイントツール)
3. ハロの使い方
4. ビューワの使い方
5. スクリプトの作り方
6. テストタイルの使い方
7. 作品の保存と読み込み
8. いろいろな部品

第2章　プログラミングテクニック

プログラムの基本
9. ハンドルでクルマを操作
10. コースに沿って動くクルマ
11. テニスゲーム

操作に関するプログラム
12. ジョイスティックで操作する
13. キーボードで操作する
14. ボタンで操作する
15. 簡単なボタンの作り方

動きの制御
16. 何かにぶつかると元に戻る
17. 瞬間移動する
18. 行ったり来たりする
19. 壁を通り抜けない
20. 手を振る動き

画像の処理
21. アニメーション
22. 画像を変える
23. キャラクタが図形を描く

ゲームプログラミング
24. 得点をつける
25. ランダムな動き
26. リセットボタン
27. シューティングゲーム

第3章　世界聴診器の使い方

28. 世界聴診器の使い方
29. 口笛で操作するクルマ
30. 世界聴診器とは？
31. 世界聴診器を使った実験
32. ゲームコントローラを作る

ITクラフトマンシップ・プロジェクト

このテキストは、IT人材の早期発掘・育成を目的とするITクラフトマンシップ・プロジェクトの一環として、経済産業省の支援を受けて制作されました。

ITクラフトマンシップ・プロジェクト

ITクラフトマンシップ・プロジェクトの目的は、次代を担う若者に対し、ITに対する意識を高め、高度な技術を学ぶことのできる場を提供することにより、将来に向けてIT界を志す動機を与え、さらに各地域においてITに長けた若者を早期に発掘し、コンピュータ・サイエンティスト、スーパー・プログラマといった専門家となり得る人材を継続的に輩出できる環境を整備していくことにあります。
（経済産業省のWebサイトからの抜粋）

2006年度の活動実績

私たちデジタルハリウッド大学大学院の2006年度の活動実績は次のとおりです。

(1)教材のパッケージ化
スクイークと、世界聴診器を活用した教材パッケージを開発しました。

(2)プログラミング教室の開催
開発した教材の効果の実証のために、学生や社会人等を対象とした指導者育成ワークショップと児童生徒向けのプログラミング教室を開催しました。

(3)「モノ作り」を通した国際交流
ネパールの子ども達と日本の子ども達による、ビデオ会議システムを活用した国際交流（作品発表会）を実施しました。

詳しくは、Webサイトをご覧ください。
http://mts-j.hiho.jp/

経済産業省ITクラフトマンシッププロジェクト採択事業
国際競争力のあるITクラフトマン育成プログラム
～世界聴診器のゲームプログラミングへの応用～

第１章　スクイークの基本操作

1 インストールと起動

先ず最初にパソコンにスクイークをインストールします。
スクイークは無料で使える「フリーソフト」です。

スクイークは公式サイト「スクイークランド」からダウンロードできます。

スクイークランド
http://squeakland.jp/

■スクイークのダウンロード

Windowsに日本語版スクイークをインストールするには左の絵をクリックしてください。
Windows 95, 98, 2000, ME, NT 3/4, XPに対応します。

Mac OS Xに日本語版スクイークをインストールするには左の絵をクリックしてください。
Mac OS X以降に対応します。

※スクイークは他のOSにも対応しています。詳しくはスクイークランドのダウンロードページを参照してください。

■スクイークの起動

Squeak

デスクトップにあるねずみのアイコンをダブルクリックするとスクイークが起動されます。

■スクイークの終了

ナビゲーターフラップの「終了」をクリックして、「はい」を選ぶと終了します。

※終了前に作品の保存を必ずしておきましょう。
（「7. 作品の保存と読み込み」参照）

2 絵を描く（ペイントツール）

ナビゲータフラップの をクリックして、絵を描くための道具「ペイントツール」を起動しましょう。

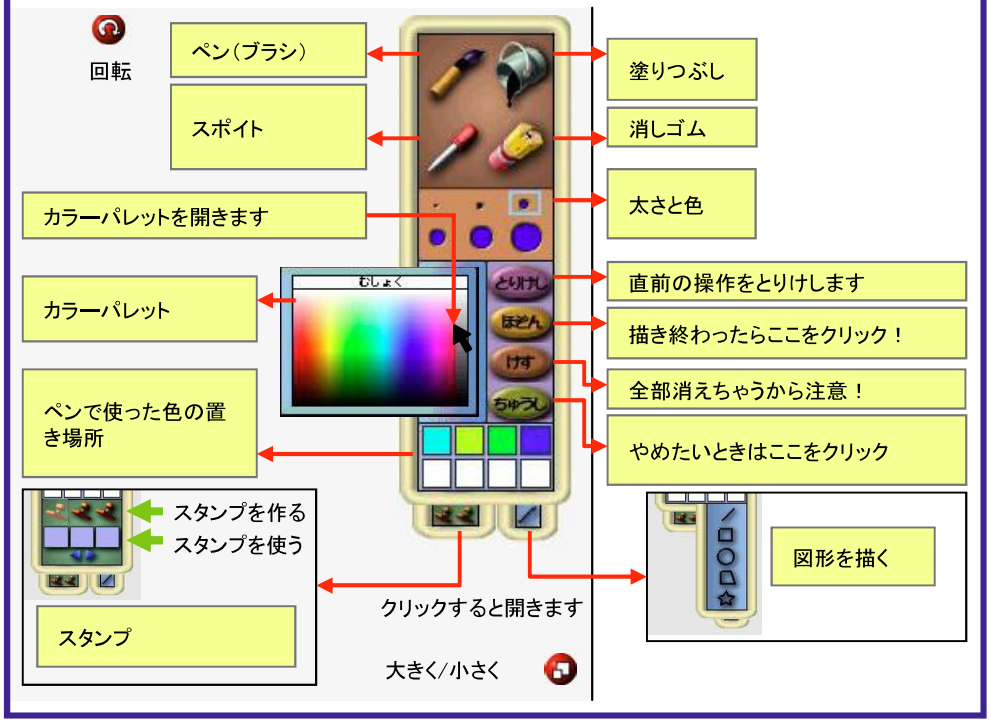
■ペイントツールの基本操作

先ず最初に、ペン・バケツ・スポイト・消しゴムの4つの機能を確認してみましょう。

絵を描いたら「ほぞん」ボタンを押すと、描いた絵が画面に張り付きます。
「とりけし」「けす」「ちゅうし」ボタンを押すとどうなるかも試してみましょう。

■ペイントツールの使い方

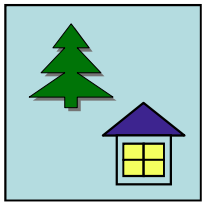
便利な機能がたくさんあります。どんな風に使えるか確認してみましょう。



■大事な注意点

①書き終わったら「ほぞん」ボタンを押す！
ひとつの絵を描いたら「ほぞん」ボタンを押して元の画面に戻りましょう。

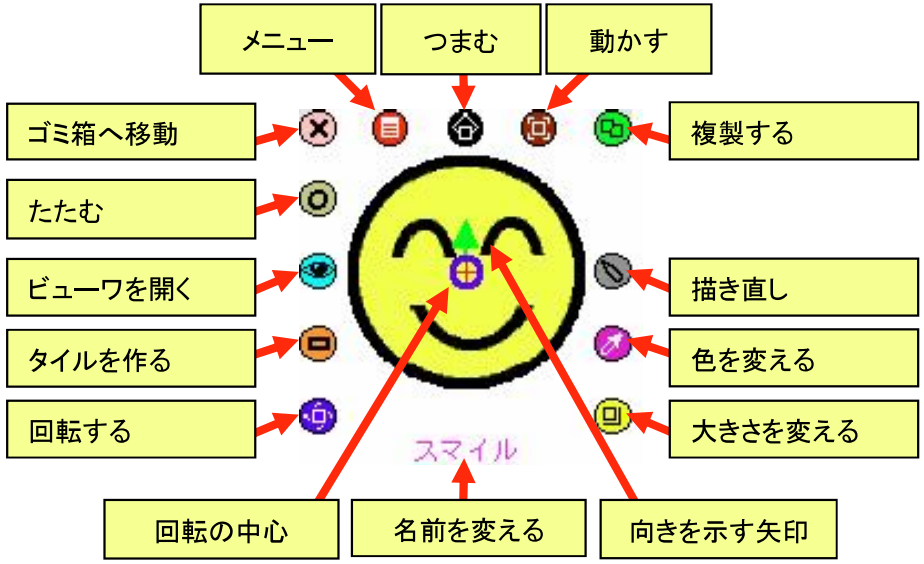
②絵はひとつづつ描くこと！
たとえば、木と家の絵を描くときには、まず木を書いて「ほぞん」してから、もう一度ペイントツールを起動して家の絵を描きましょう。



③「けす」ボタンは全部消えちゃうから注意！
もし間違って「けす」ボタンを押してしまったときは、すぐに「とりけし」ボタンを押して元に戻りましょう。ただし「とりけし」ボタンは1回だけしか戻りません。

3 ハロの使い方

ハロは、その画像に対して何か操作をするときに使う、小さなアイコンのことです。



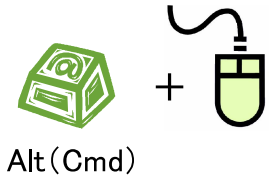
■ハロの基本操作

ハロは、画像の周りに現れる小さなアイコンのこと。ハロを使うと、絵や図形の大きさを変えたり、まわしたり、コピーしたり、いろいろな操作ができます。いくつかのハロを実際に操作してみましょう。

■ハロの出し方

ハロの出し方はいくつか方法があります。
それぞれ試してみて、自分が使いやすい方法でハロを出しましょう。

①Alt(Cmd)＋左クリック



絵や部品の色がついているところにマウスポインタを置いて、キーボードの[Alt]キー(あるいは[Cmd]キー)を押しながらマウスを左クリックするとハロがでます。

Windowsの場合　：　[Alt]キー
Macの場合　　：　[Cmd]キー

②マウスの真ん中ボタンクリック

絵や部品の色がついているところにマウスポインタを置いてマウスの真ん中のボタンをクリックするとハロがでます。

③ 描いた絵の上にマウスポインタを置く

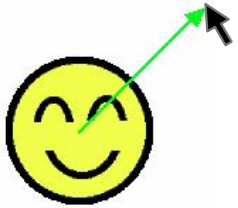
描いた絵の色がついているところにマウスポインタを置いてしばらく動かさないで待っているとハロが出ます。

※部品などはこの方法でハロが出ない場合もあります。

■大事な注意点

ハロを出すときには、マウスポインタを絵や図形の色の付いているところに置きましょう。色が付いていないところにマウスポインタを置いてもハロはでてきません。

■進む方向を変える



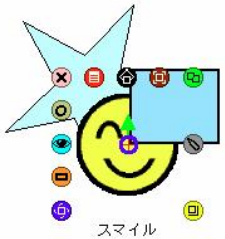
向きを示す矢印をつまんで引っ張ると、進む方向を変えることができます。

■回転の中心を移動する

「回転の中心 」を[Shift]キーを押しながらつまむと、回転するときの中心を移動することができます。

■「つまむ」と「動かす」の違い

いくつかの画像が重なっているときには、「つまむ」と「動かす」をうまく使い分けましょう。



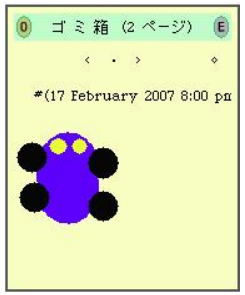
つまむ ： つまんだ画像が一番上にきます。

動かす ： 下側にある画像も、位置を変えずにそのまま移動できます。

■ゴミ箱の使い方

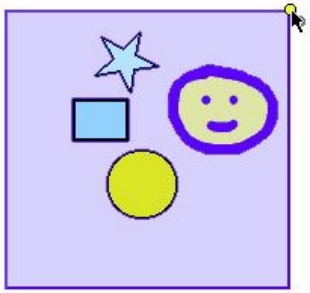
ハロの「ゴミ箱へ移動 」をクリックすると、画像がゴミ箱に捨てられます。

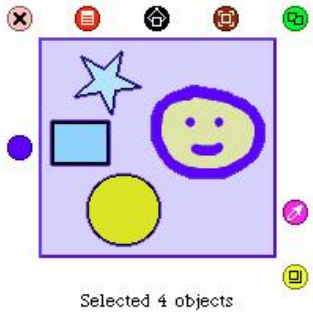
一度ゴミ箱へ捨てた画像も、ゴミ箱を開くと、そこからまた取り出すことができます。



■複数の絵を一度に選択する方法

[Shift]キーを押しながら、マウスをドラッグ（左ボタンを押しながらマウス移動）してみましょう。複数の画像を選択して、選択画像全体のハロを出すことができます。





複数の画像を[Shift]キーを押しながらマウス操作で選択します。

全体のハロがでてくるので、まとめて移動したり、コピーしたりできます。

■「兄弟」の作り方

[Shift]キーを押しながら複製 をすると、「兄弟」を作ることができます。

「兄弟」は、同じひとつのスクリプトをいっしょに利用します。つまり、同じスクリプトを一緒に利用する「兄弟」ということですね。同じ動きをするたくさんのキャラクタを作るときに利用しましょう。

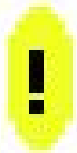
「兄弟」はひとつのスクリプトを一緒に使います。

普通に複製したときには、スクリプトもコピーされるので、スクリプトは別々になります。

4 ビューワの使い方

ビューワは、プログラミングのための道具箱のようなものです。どんなことができるか確認しておきましょう。

■ビューワの基本操作



ビューワの左端にある「！」ボタンをクリックすると何が起こるか確認してみましょう。

ハロの「つまむ」「動かす」「回転する」を使って、絵を動かしたときに、ビューワの「x座標」「y座標」「向き」の数字がどう変わるかを調べてみましょう。

■ビューワの使い方

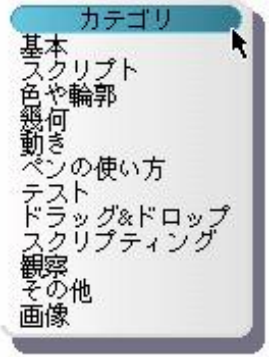
ビューワを小さくたたむときはここをクリック。画面の端に小さくなったら、もう一度クリックすると元に戻ります。

ビューワを画面から消すにはここをクリック
（削除はされません）

絵や部品の名前はハロでも変えられますが、ビューワでも変えられます。

■カテゴリ

ビューワにはたくさんのタイルが登録されています。タイルは「カテゴリ」ごとに分類されているので、それぞれのカテゴリにどんなタイルがあるかを確認してみましょう。



カテゴリ欄を追加

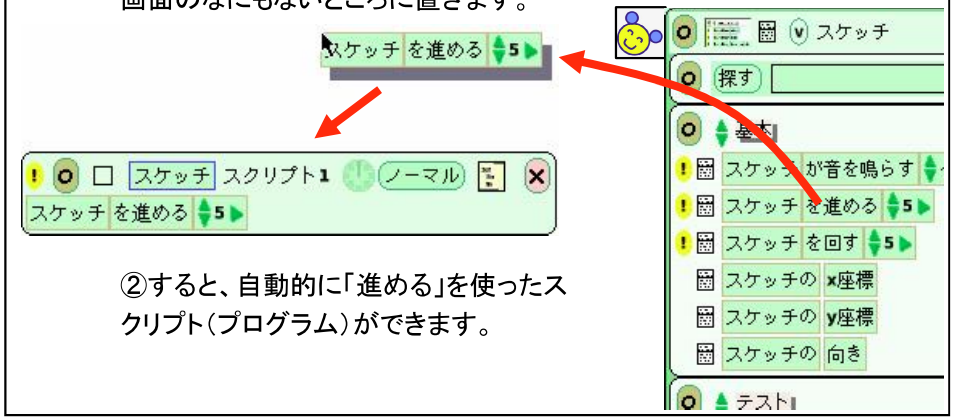
カテゴリの選択

追加したカテゴリを消したいときはここをクリック。
（削除はされません）

5 スクリプトの作り方

スクイークではプログラムのことを「スクリプト」と呼びます。
まず最初に「進める」を使ったスクリプトを作ってみましょう。

①ビューワの「進める」をドラッグして、
画面のなにもないところに置きます。



②すると、自動的に「進める」を使ったスクリプト（プログラム）ができます。

■スクリプトの基本操作

スクリプトを作るための枠のことを「スクリプター」といいます。
スクリプトを作ったら、どんなスクリプトかわかるように名前を付けておきましょう。

スクリプトに名前をつけることができます

スクリプトの実行・停止

数値を変えてみましょう。どんな風に動くかな？

■スクリプトをしまう

作ったスクリプトは、ビューワの中にしまっておくことができます。

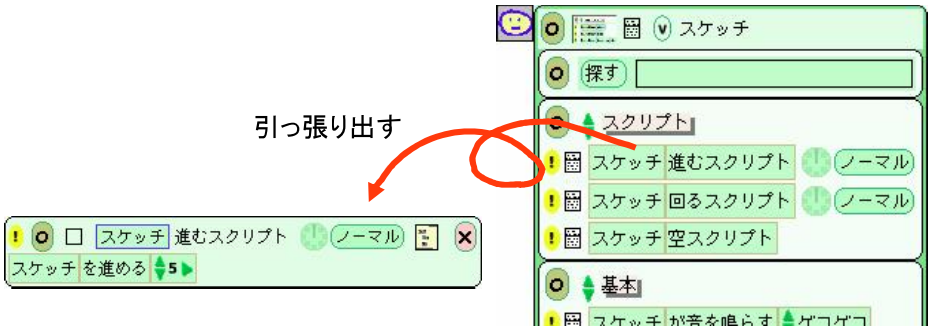
作ったスクリプトをしまっておくときはここをクリック。
画面から消えますが、削除はされません。

作ったスクリプトを捨てたいときはここ をクリックするとゴミ箱に捨てられます。

■スクリプトを取り出す

作ったスクリプトを取り出すには、ビューワの「スクリプト」カテゴリから引っ張りだしてください。

引っ張り出す



ひとつのビューワの中には、複数のスクリプトを作ることができます。
たとえば、上のビューワには、「進むスクリプト」と「回るスクリプト」というふたつのスクリプトが登録されていることがわかります。

※スクリプトには分かりやすい名前をつけておきましょう。

■タイルの組み込み方

スクリプターにいろいろなタイルを組み込むことで、スクリプトを作ります。
「進める」「回す」「音を鳴らす」など、ビューワに登録されているたくさんのタイルを使ってスクリプトを作ってみましょう。

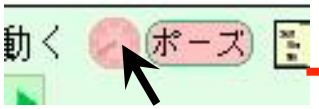
タイルの左上をつまむと、組み込みがやり易いようです。

■進んで回るプログラム

「進む」と「回す」でスクリプトを作ったら、数値を変えて動かしてみましょう。
たとえば、数値をマイナスにするとどんな動きをするでしょうか？

■スクリプトを早く動かす

スクリプトを早く動かすには「刻み値」を変えてみましょう。「刻み値」を変えるには、スクリプターの時計マークをマウスで長押しします。
「8」が指定されていれば、1秒間にスクリプトが8回実行されます。
数値を大きくしてみましょう。



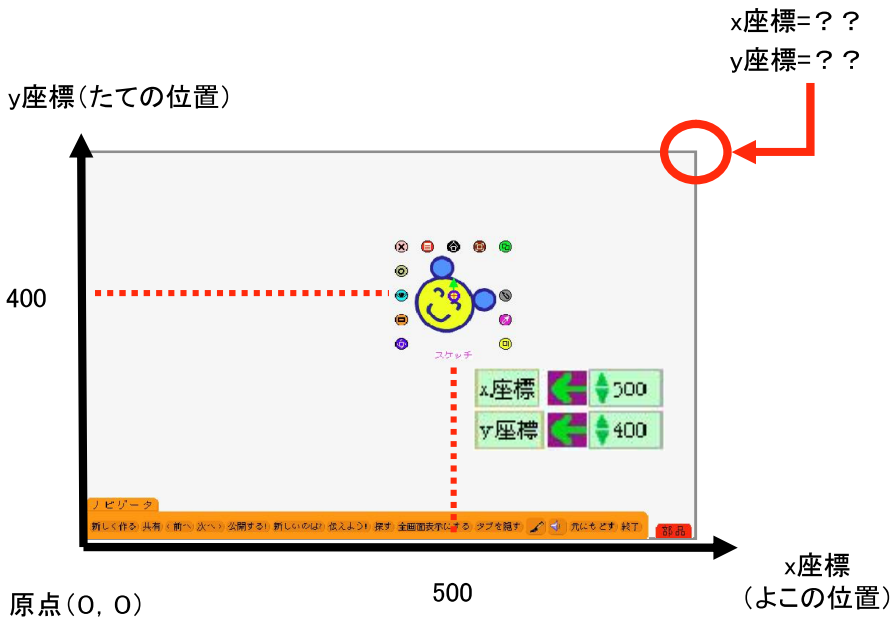
ここをしばらく押していると
数字のメニューがでてきます。

x座標とy座標

x座標とy座標は、画面上の場所のことです。

使っているパソコンの種類によって画面の大きさが違うので、自分のパソコンの画面のサイズ（x座標とy座標の最大値）を事前に確認しておきましょう。

使っているパソコンの画面の大きさが
どれくらいか確認しておきましょう。



6 テストタイルの使い方

プログラミングをする上で一番大切な機能がこのテストタイルです。どんなことができるかしっかり確認しましょう！

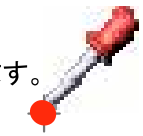
「テストタイル」を使うときには、
ビューワの「テスト」カテゴリにある
タイルを使います。

テストタイルの「テスト」の文字の横に、
タイルを組み込みます。

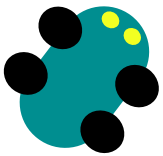
■テストタイルの基本操作

テストタイルに色を指定するときには、スポイトを使って色を吸い取りましょう。

色をクリックすると
スポイトが出てきます。



スポイトで
指定したい色を
吸い取りましょう。



■テストタイルの組み込み

テストタイルを出すには、スクリプターの をクリックします。

ここを押すと
テストタイルが
出ます

まず「進む」スクリプトを作りましょう。

テストタイルがでてきたら・・・

色が変わったここに組み込みます

テストタイルがスクリプトに組み込まれました。

ビューワの「テスト」カテゴリや、「基本」カテゴリなどにあるタイルを、
テストタイルに組み込んでみましょう。

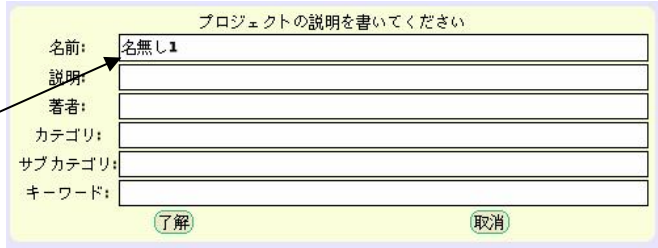
7 作品の保存と読み込み

パソコンの調子が悪くなった場合に備えて、作品を作っている途中でもときどき保存をしましょう。

■作品の保存

作った作品を保存するときには、ナビゲーターフラップから **公開する！** をクリックします。

①作品の名前を書いて「了解」ボタンを押します。



作品の名前を記入します

②保存する場所を選んで、「保存」ボタンを押します。

ここをクリックして選択します

■作品の読み込み

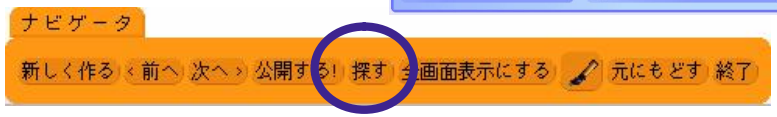
保存した作品を読み込むには、ナビゲーターフラップから **探す** をクリックして、保存してある作品を選びます。

①読み込む場所を選びます。

ここをクリックして選択します

②作品を選んで、「了解」ボタンを押します。

読み込む作品を選択します



■保存される場所

保存した作品は、パソコンの次の場所に格納されています。

Windowsの場合　：　Cドライブの「My Squeak」フォルダ

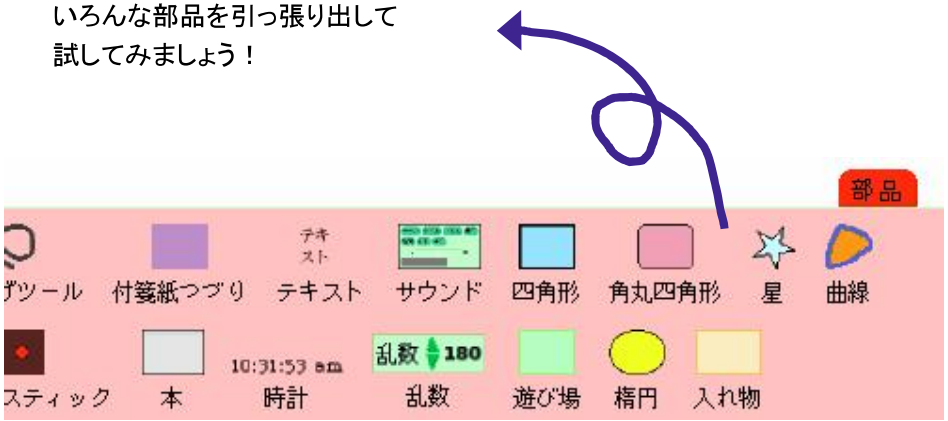
Macの場合　　：　デスクトップにある「My Squeak」へのショートカット

このフォルダにはいってる「.pr」という拡張子がついたファイルがスクイークの作品です。公開するごとにファイル名には3桁の連番（003とか）が付くので、メールで送るときには、一番新しいファイルを選んでコピーしましょう。

8 いろいろな部品

「部品」フラップにはたくさんの部品が登録されています。
どんな部品があるのか引っ張り出してみましょう。

いろんな部品を引っ張り出して
試してみましょう！



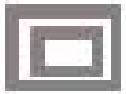
■オブジェクトのカタログ

「部品」フラップに登録されている以外にも、スクイークにはいろいろな部品が組み込まれています。

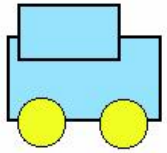
「オブジェクトのカタログ」には、たくさんの部品が登録されているので、どんな部品があるのかを調べてみましょう。

■長方形ツール、輪投げツール

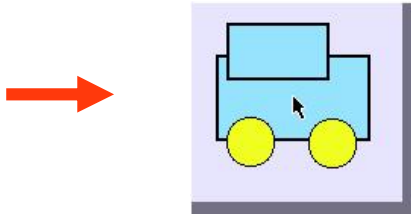
画面にあるものを切り取って、ひとつの画像にすることができます。長方形ツールだと四角く、輪投げツールだと自由な範囲で、切り取ることができます。


長方形ツール

輪投げツール



「四角形」や「楕円」などの画像を使ってひとつの絵を組み立てます。



長方形ツールを使って範囲指定すると、ひとつの画像になってコピーされます。コピーしたら、ハロ「書き直し」で、画像の周りの色を「むしょく」で塗りつぶします。

■全スクリプト

全てのスクリプトをまとめて操作するときに使います。

全てのスクリプトをストップします。

全てのスクリプトを1回だけ実行します。

全てのスクリプトをスタートします。

「このスクリプトを実行するとき」が「ノーマル」になっていると、全スクリプトでは動きません。「ポーズ」にしておきましょう。

■テキスト（文字の入力）

画面上に文字を入力したいときには、部品から「テキスト」を引っ張り出します。

テキスト

テキスト

①文字の入力

部品から「テキスト」を引っ張り出したら、「テキスト」と書いてある文字を選択して、別の文字を入力しましょう。

文字が選択された状態です。
周りが緑色になります。

②文字の大きさ

文字を大きくしたいときは、ハロの「フォント」をクリックしてフォントサイズを変更できます。

もっと大きくしたいときには、ハロの「回転する」で、ちょっとだけ回転させてから、「大きさの変更」で大きくすると文字が大きくなります。

大きな字

③文字の色

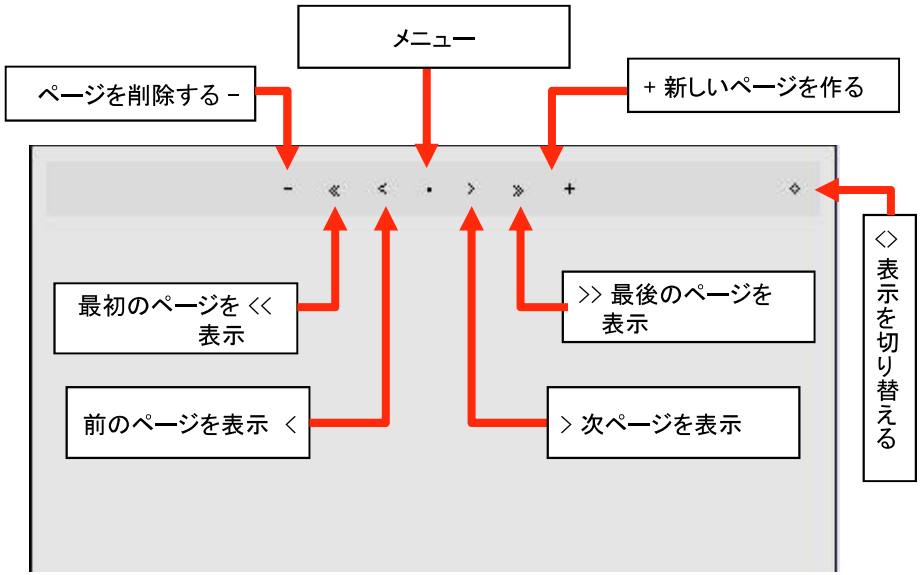
「テキスト」のビューワを開いて、「色や輪郭」カテゴリを開いてください。
ここでテキストの色や背景の色を変えることができます。

「色や輪郭」カテゴリ

■本

部品「本」を使うと、オリジナルの絵本や、ロールプレイングゲームを作ることができます。

本



■本を使うときの注意点

①本のなかでハロを出すとき

何回か続けてクリックすると絵のハロがでてきます。
これは、最初にクリックしたときには、「本」自身や、「本」に組み込まれている「ページ」のハロがでてくるからです。

②絵が本に組み込まれないとき

それは、絵がそのページにきちんと置かれていなくて、本の表面に浮かんでいるからです。
そのときには、ハロの「を出して「つまむ」を使ってすこしだけ動かしてみてください。
そうすると、そのページに組み込まれます。

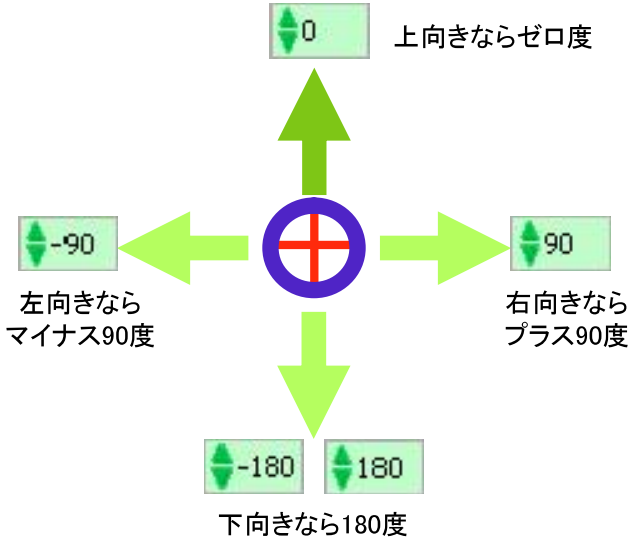
矢印の向きと角度の関係

「向き」は絵の向いている角度を表しています。
絵の向きを示す「矢印」が上を向いているときが「ゼロ度」です。

画像を回転させてみましょう。

ハロの矢印が右斜め上を指していれば、角度は45度です。

矢印の向きと角度の考え方



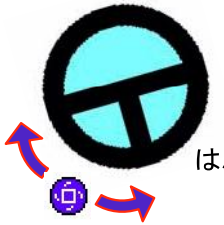
第2章　プログラミングテクニック

プログラムの基本

9 ハンドルでクルマを操作

「進める」と「回す」のスクリプトを応用して、ハンドルで操作するクルマを作ってみましょう。

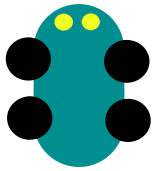
クルマ



はんどる

ハロを使ってハンドルを回して、クルマを左右に操縦できます。

準備するもの



動かすキャラクタをひとつ描きます。
ここでは、クルマを使いますが、乗り物でも、動物でも、なんでもかまいません。

ハンドルを描きます。
どんな形のハンドルでもいいので、自由に描きましょう。

ハロの「回転する」を使って回しやすい形にしておきます。

スクリプト

「ハンドルの向き」の値で、クルマが回るスクリプトです。

【使用するカテゴリ】
クルマ:　「基本」
ハンドル:　「基本」

■ハンドルを回してみる

先ず、ハンドルのハロ「回転する」をつまんでハンドルを回してみましょう。
ビューワの「向き」の数値が変化するのがわかりますか？
このハンドルの「向き」を利用して、くるまをコントロールしてみます。

①ハンドルを で回す

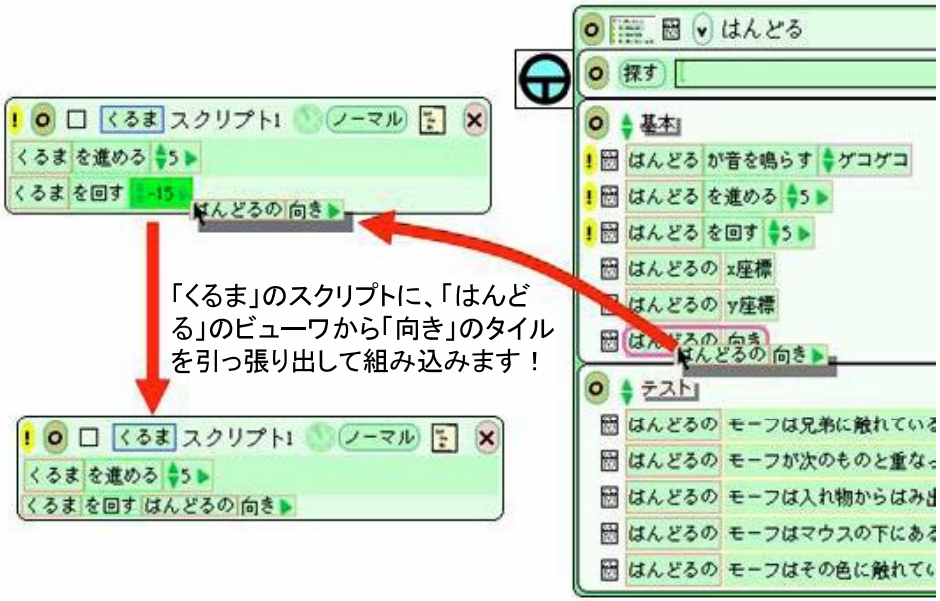
②向きの数値が変わる

■進んで回るプログラム

前にやったとおり、クルマのビューワから「進める」と「回す」を使って、クルマを動かすスクリプトを作っておきましょう。

■「ハンドルの向き」を組み込む

ハンドルのビューワにある「向き」のタイルを引っ張り出して、クルマのスクリプトの「回す」の数値のところに、ドラッグ＆ドロップしてみましょう。
ハンドルの回す角度が「回す」タイルに組み込まれます。



「くるま」のスクリプトに、「はんどる」のビューワから「向き」のタイルを引っ張り出して組み込みます！

■ハンドルで動かす

作ったスクリプトを実行してみましょう。

スタート・ストップ

ハンドルの向きが−15〜+15くらいになるように、少しだけ動かすのが上手くコントロールするコツです。

■もっと使いやすくする

このままだと、ハンドルをちょっと回しただけでクルクル回ってしまうから少し使いにくいかもしれません。こんなときは、次のように設定を変えてみましょう。

①この三角 ▶ をクリック

②「/」に変更。（割り算の記号）

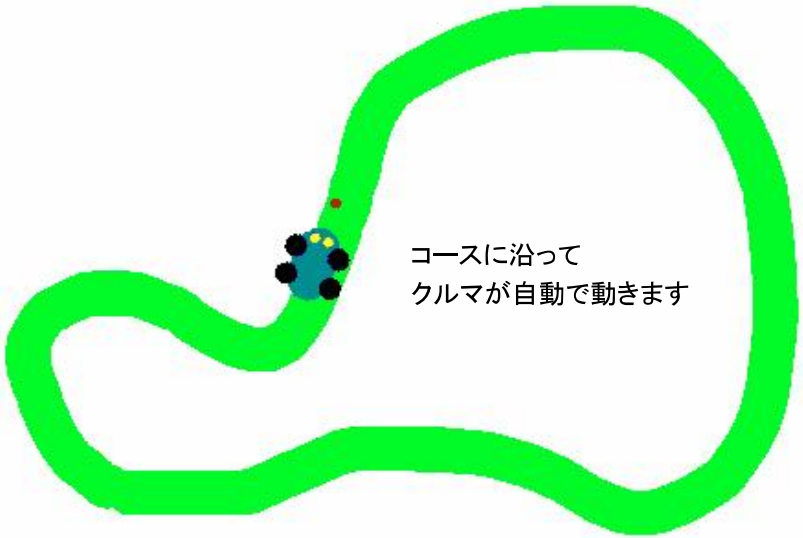
③「10」に変更。

こうすると、「ハンドルの向き」の値を10で割った数値が、クルマを回す値になるので、操作がしやすくなります。

プログラムの基本

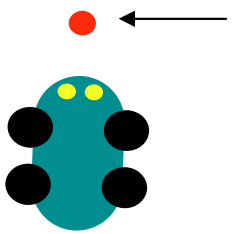
10 コースに沿って動くクルマ

コースに沿って、自動的に動くクルマを作ってみましょう。



準備するもの

クルマの前に、コースを読み取るための点をひとつ、色を変えて描いておきましょう。
これが「センサ」となって、コースを読み取ります。



この赤い丸印が「センサ」です。

センサの位置は、進む方向の先端に置きましょう。
（ハロを出して、進む矢印の方向がセンサのある位置と同じことを確認しておきましょう）

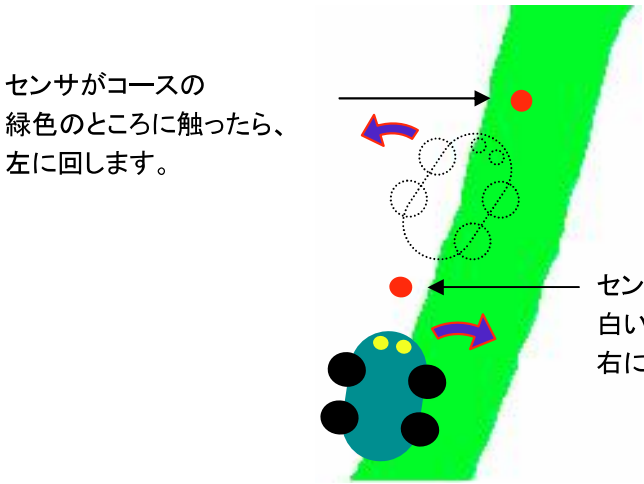
スクリプト

センサがコース上にあるときには＋5度、コース上にないときには−5度回すスクリプトです。

【使用するカテゴリ】
クルマ:　　「基本」、「テスト」

スクリプトの解説

クルマが進んだとき、センサが触った色がコースの外の「白」なら右に、コース内の「緑色」なら左に回すようにすれば、コースの縁に沿ってクルマが動きます。

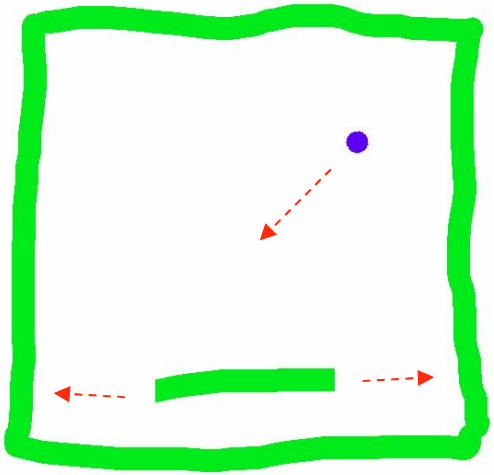


うまく縁に沿って動かないときは、クルマを回す角度を大きくしたり小さくしたりしてみて、ちゃんとコースを読み取るように変えてみましょう。

プログラムの基本

11 テニスゲーム

壁に当たるとはねかえるボールをラケットで打ち返す、簡単なテニスゲームです。

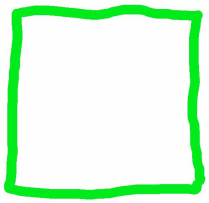


ボールがテニスコートの中を跳ね返りながら動きます。

※ラケットは、マウスでつかんで動かします。

準備するもの

ボールの画像



コートの画像
コートとラケットは同じ色にしておきましょう。

ラケットの画像
ラケットはマウスでつまんで動かします。

スクリプト

ボールが壁やラケットの色に触れたら、90度回るようにします。

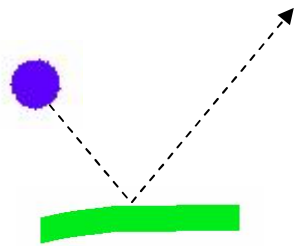
【使用するカテゴリ】
ボール:　　「基本」、「テスト」

スクリプトの解説

①ボールが進みます。

②ボールがコートとラケットの色に触れていたら、90度回転します。

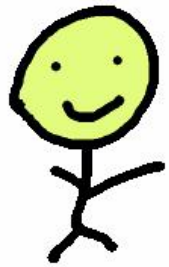
うまく動いたら、回す角度を変えて、ボールの跳ね返り方を調整してみましょう。
ボールの速さを変えるには、「進める」の数字を増やしてもいいですが、「刻み値」を変えたほうがスムーズに動きます。
（「5. スクリプトの作り方」参照）



操作に関するプログラム

12 ジョイスティックで操作する

ジョイスティックを使って、キャラクタを上下左右に動かしてみましょう。

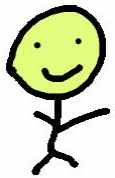


歩く人

描いた絵を、ジョイスティックで
上下左右に動かしましょう。

ジョイスティック

準備するもの



歩く人
ジョイスティックで動かすキャラクタを描きます。

ジョイスティック
「部品」フラップから引っ張り出してください。

スクリプト

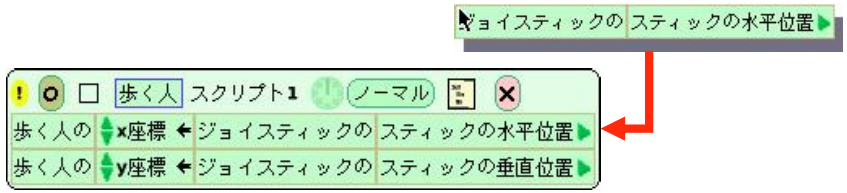
ジョイスティックの「水平位置」「垂直位置」の数値を「歩く人」のx座標、y座標に足すスクリプトです。

【使用するカテゴリ】
歩く人:　　「基本」
ジョイスティック:「ジョイスティック」

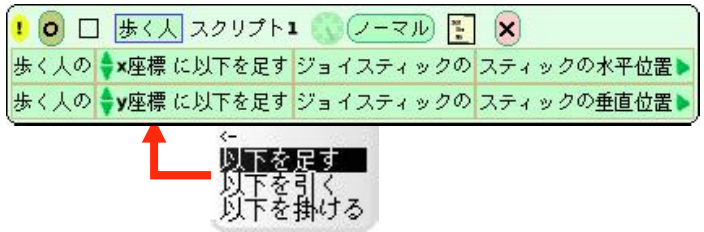
スクリプトの解説

①先ず「歩く人」の基本カテゴリのx座標とy座標を使ってスクリプトを作ります。

②次にジョイスティックの「ジョイスティック」カテゴリから、水平位置と垂直位置を引っ張り出して、x座標、y座標の数値部分へ組み込みます。



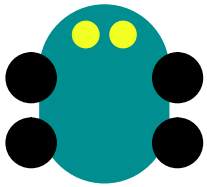
②最後に、スクリプトの「x座標」「y座標」と書いてある部分をクリックして「以下を足す」を選びます。



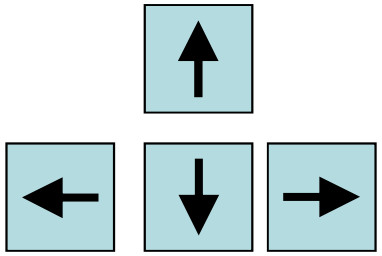
操作に関するプログラム

13 キーボードで操作する

パソコンのキーボードで、キャラクタを操作します。

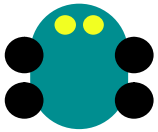


クルマ



キーボードの矢印キーで
操作します

準備するもの



動かしたい画像をひとつ
描いておきます。

「ワールド」について

このスクリプトでは、「ワールド」の
ビューワを使います。画面の何もないと
ころでハロを出してみてください。すると、
画面全体のハロがでてきます。
これが「ワールド」のハロです。

スクリプト

「←」キーが押されたら左に、「→」キーが押されたら右に曲がります。

【使用するカテゴリ】
クルマ: 「基本」
ワールド: 「入力」

スクリプトの解説

①「←」キーが押されたら－5度回します。（左回り）
②「→」キーが押されたら5度回します。（右回り）

ワールドのビューワを出したら、カテゴリを「入力」にしてください。
何かのキーボードを押すと「最後の入力文字」の値が変化することがわかります。この値を使って、スクリプトを作りましょう。

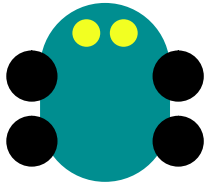
キーを押すと
この値が変ります

このスクリプトでは、キーを押していないときでも回り続けてしまいます。
キーを押したときだけ回したいときは、スクリプトの最後に「最後の入力文字」に空白を代入するタイルを追加してみましょう。

操作に関するプログラム

14 ボタンで操作する

画面にボタンを描いてマウスでクリックすることで、キャラクタを操作します。



クルマ

クリックするとクルマが左に曲がります

左に曲がる

クリックするとクルマが右に曲がります

右に曲がる

準備するもの

クルマ（動かしたい画像）

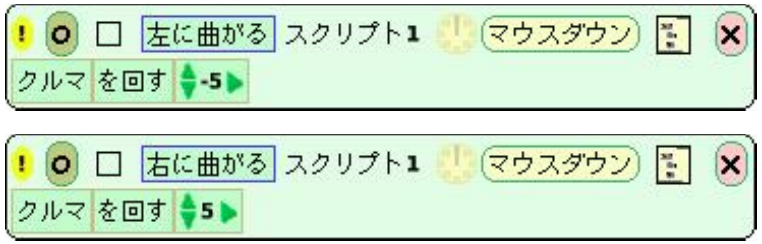
右に曲がるボタン（自分の好きな画像を書いてください）

左に曲がるボタン（自分の好きな画像を書いてください）

このスクリプトは、ボタンに組み込みます。
ボタン画像のビューワを開いてから、スクリプトを作ってください。

スクリプト

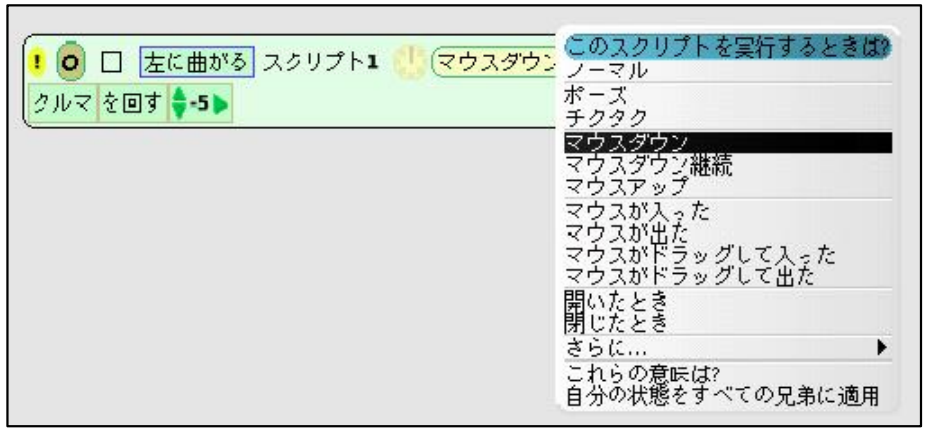
それぞれのボタンに、クリックされたときにクルマを回すスクリプトを組み込みます。



【使用するカテゴリ】
クルマ:　「基本」
左に曲がる:　「スクリプト」
右に曲がる:　「スクリプト」

スクリプトの解説

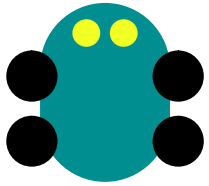
スクリプターのここをクリックして、「マウスダウン」にすると、マウスをクリックしたときにだけスクリプトが実行されます。



操作に関するプログラム

15 簡単なボタンの作り方

ボタンの画像を描かなくても、スクリプトを作るだけで簡単にボタンを作ることができます。



クルマ

クリックするとクルマが
左に曲がります

クリックするとクルマが
右に曲がります

準備するもの

クルマ（動かしたい画像）

このスクリプトは、クルマに組み込みます。
クルマのビューワを開いてから、スクリプトを作ってください。

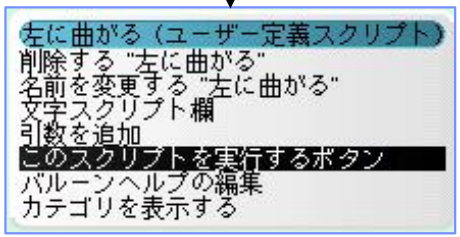
スクリプト

クルマに2つのスクリプトを作ります（左に曲がる、右に曲がる）
その後で、このスクリプトを実行するためのボタンを作ります。

【使用するカテゴリ】
クルマ:　　「基本」、「スクリプト」

スクリプトの解説

「左に曲がる」と「右に曲がる」のスクリプトをそれぞれ作ったら、「スクリプト」カテゴリを開いて、ボタンを作ることができます。



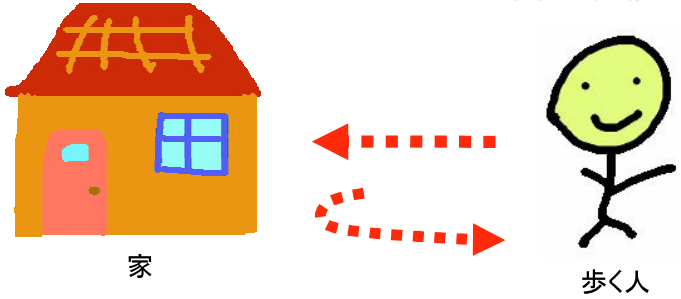
ビューワの をクリックして
「このスクリプトを実行するボタン」を
選択するとボタンが出来ます。

動きの制御

16 何かにぶつかると元に戻る

何かにぶつかったら元の位置に戻ります。
x座標、y座標の考え方を理解しましょう。

「歩く人」が「家」にぶつかったら
元いた位置に移動します。



準備するもの

歩く人（動かしたい画像）
このキャラクタに、スクリプトを組み込みます。

家（ぶつかるもの）

「歩く人」をマウスでつかんで動かしてみましょう。そのときの、x座標、y座標の値が、「歩く人」の画面上の座標になります。

スクリプト

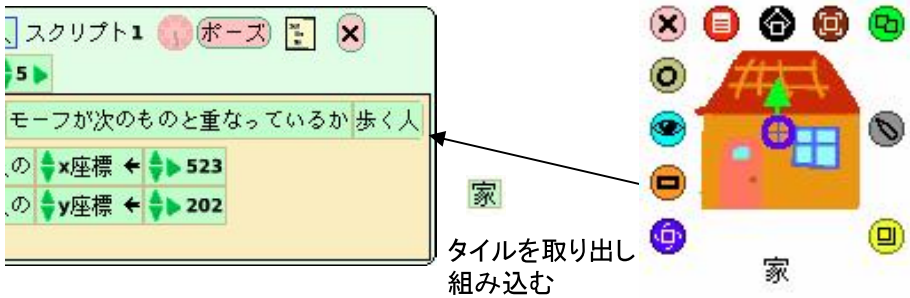
「歩く人」が「家」に重なったときに、「歩く人」のx座標とy座標に値を指定して、移動させます。

【使用するカテゴリ】
歩く人：　「基本」、「テスト」

スクリプトの解説

①「歩く人」を進めます。
②「家」と重なったときに、元いた位置のx座標とy座標の値（x=523、y=202）を指定します。（ をつまんで組み込みます）

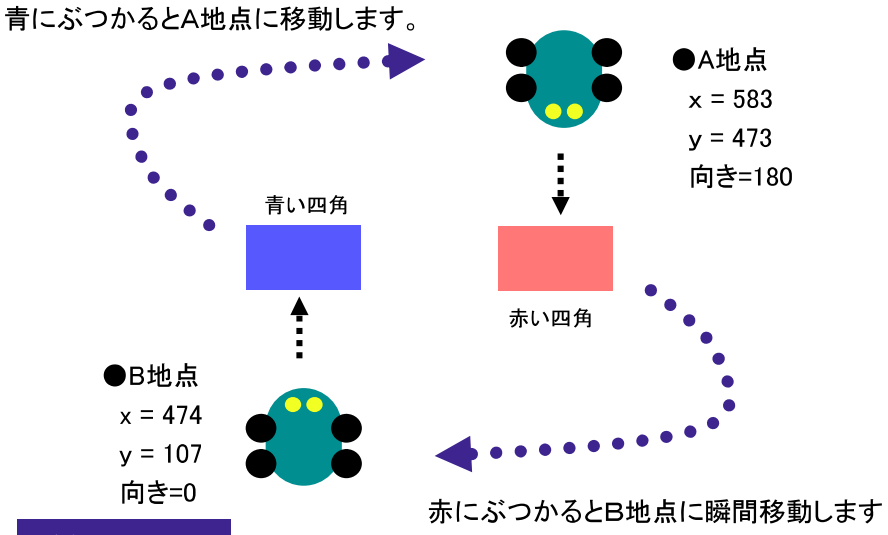
テストタイル「次のものと重なっているか」を使うときには、家のハロの を クリックして「家」タイルを取り出し、テストタイルに組み込みます。



動きの制御

17 瞬間移動する

ぶつかったものの色に応じて、
移動する場所が変わるクルマです。



準備するもの

クルマ（動かしたい画像）
このキャラクタに、スクリプトを組み込みます。

ワープゾーン1（青い四角）

ワープゾーン2（赤い四角）

スクリプト

「クルマ」が色に重なったときに、x座標とy座標を指定して移動させます。
その際に、「向き」の値も変化させることに注意しましょう。

【使用するカテゴリ】
クルマ:　　「基本」、「テスト」

スクリプトの解説

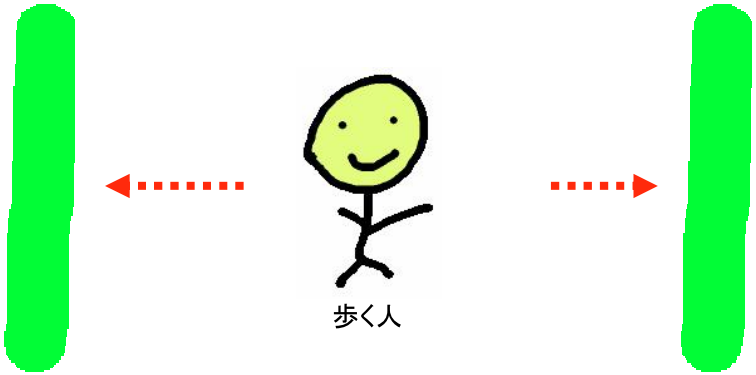
①クルマを進めます。
②青い四角の上にあるとき、x座標、y座標、向きを指定します。
③赤い四角の上にあるとき、x座標、y座標、向きを指定します。

「向き」はクルマが上向きの場合が0度、下向きの場合は180度です。

動きの制御

18 行ったり来たりする

キャラクターが左右に行ったり来たりする動きです。



壁にぶつかると行ったり来たりします

準備するもの

歩く人（動かしたい画像）
このキャラクタに、スクリプトを組み込みます。

壁（同じ色にしておきましょう）

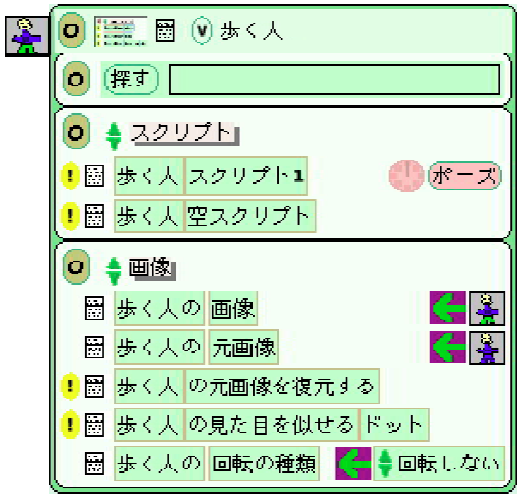
スクリプト

色にぶつかったときに180度回転するスクリプトです。

【使用するカテゴリ】
歩く人：　「基本」、「テスト」、「画像」

スクリプトの解説

このスクリプトだけだと、回転したときに上下が反対になってしまうので、
画像カテゴリの「回転の種類」を「回転しない」にしておきます。



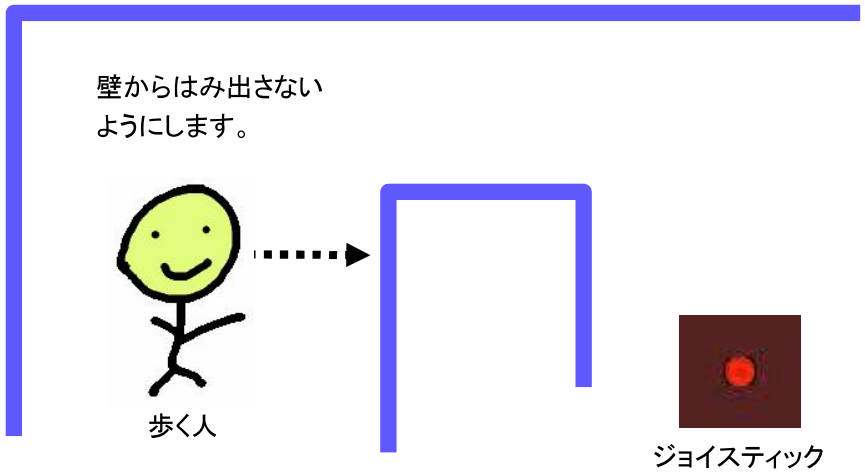
「回転しない」
にします

動きの制御

19 壁を通り抜けない

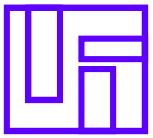
ジョイスティックを使った迷路ゲームを作るときに、キャラクターが壁を突き抜けないように設定します。

壁からはみ出さない
ようにします。



準備するもの

歩く人（動かしたい画像）
このキャラクタに、スクリプトを組み込みます。



キャラクタが動き回る迷路を作りましょう。

ジョイスティックを「部品」フラップから引っ張り出してください。

スクリプト

ジョイスティックの「水平位置」「垂直位置」の数値を「歩く人」のx座標、y座標に足すスクリプトです。

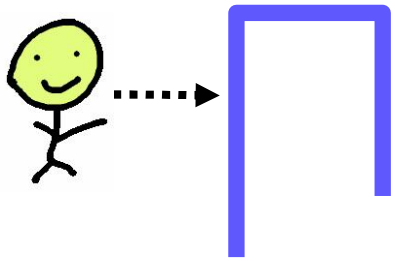
【使用するカテゴリ】
歩く人：　「基本」
ジョイスティック：「ジョイスティック」

スクリプトの解説

①ジョイスティックで動かすスクリプトです。
（「12. ジョイスティックで操作する」参照）

②歩く人が道の色に触れているときに、直前（①）で足したx座標とy座標を引きます。

この方法は、ひとつの例にすぎません。
変数を使ったり、いろいろ工夫してみると、他にも同じことをもっと上手に実現する方法があるかもしれません。
いろいろなやり方を考えて、試してみましょう！



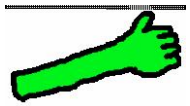
動きの制御

20 手を振る動き

手を左右に振る動き（バイバイ）を作ってみましょう

手を左右に振ります。

準備するもの



動かす手を、人の体とは別に描きます。
回転の中心を手の端のほうに移動しておきます。
（「3. ハロの使い方」参照）

手と組み合わせて使う
体部分を描きます。

変数の作成
手のビューワを開いて変数「動く角度」を新たに作ります。

スクリプト

手を振る動きは、手を左右に回転させて、行ったり来たりするように動かします。

【使用するカテゴリ】
手:　　「基本」、「変数」

スクリプトの解説

①変数「動く角度」の値だけ、手を回します。
（手を回すスクリプトを作ってから、変数「動く角度」をセットします）

②テストタイル1
「手の向き」が20度よりも大きくなったら、変数「手の動く角度」に-5を代入します。（手が右側に動き始めます）

③テストタイル2
「手の向き」が-20度よりも大きくなったら、変数「手の動く角度」に5を代入します。（手が左側に動き始めます）

※手の振り方を変えたいときには、角度の数値を調節してみましょう。

■変数の作り方

「変数」とは、ある値をしまっておくための、特別な場所のことです。
たとえば、「動く角度」という名前の変数を作っておけば、この変数に指定した文字や数値を格納しておくことができます。

①変数の作り方

スクリプターの をクリックすると、変数を作ることができます。

ここをクリック！

変数の名前を指定して、了解ボタンを押します。

ビューワの「変数」カテゴリに指定した名前の変数が作られます。

②変数の削除

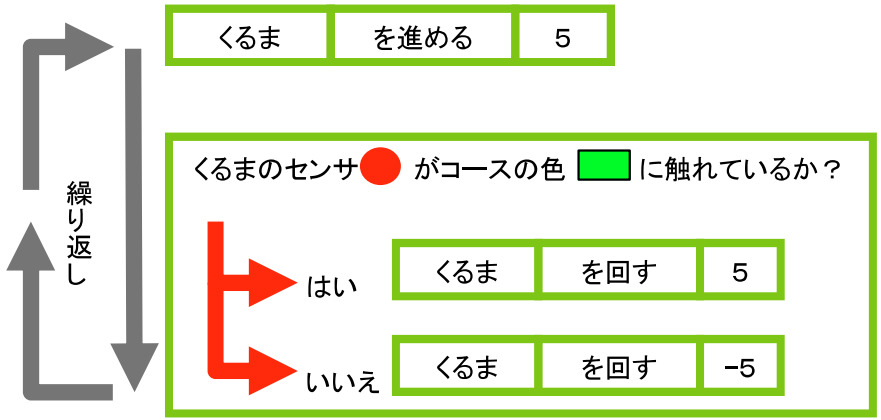
変数の前にある をクリックすると、変数を削除することができます。

ここをクリック！

条件判断とアルゴリズム

テストタイルを使うことで、次のように、ある条件が「はい」のときと「いいえ」のときでそれぞれ違った命令を実行することができます。

ある条件によってプログラムの流れを変えることができるこの「条件判断」の考え方を使えば、これからいろんな動きのプログラムを作ることができますね。



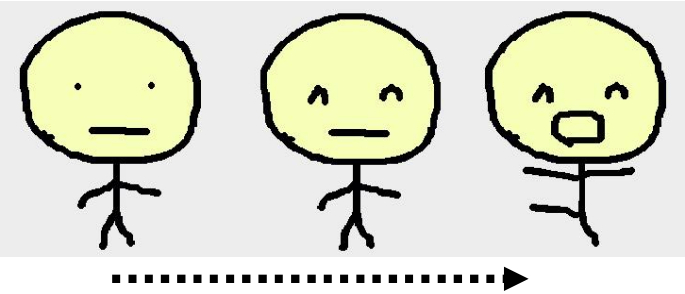
こんな風に、コンピュータを使ってある特定の目的を達成するための処理手順のことを「アルゴリズム」といいます。

つまり、クルマがコースに沿って動かすための「考え方」のことをアルゴリズム、そしてこのアルゴリズム（考え方）を、スクイークのようなプログラミング言語を使って具体化したものが「プログラム」です。

画像の処理

21 アニメーション

パラパラまんがのようなアニメーションを作ります。

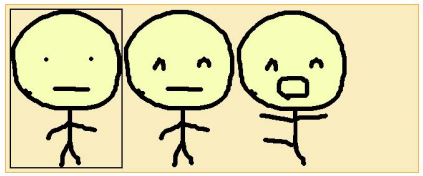


順番に表示されます

ポイント！
1枚描いたら、ハロで「複製」してから、「描き直し」ですこし絵を修正すれば、少しずつ動いているような絵が描けます。

準備するもの

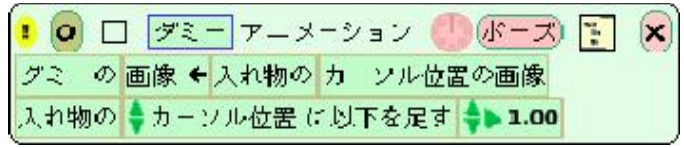
アニメーション用画像
ダミー用画像
入れ物(部品フラップ)



少しずつ違う絵を描き、一枚ずつ保存します。
部品フラップから「入れ物」を出して、
画像を順番に入れます。
ダミーの絵を描きます。どんな絵でもいいです。

スクリプト

「入れ物」に登録した画像を、「ダミー」へ次々とセットすることで、アニメーションのように動くキャラクタができます。



【使用するカテゴリ】
ダミー画像:　「画像」
入れ物:　「集合」

スクリプトの解説

①ダミーの「画像」に、入れ物カーソル位置の画像をセットする。

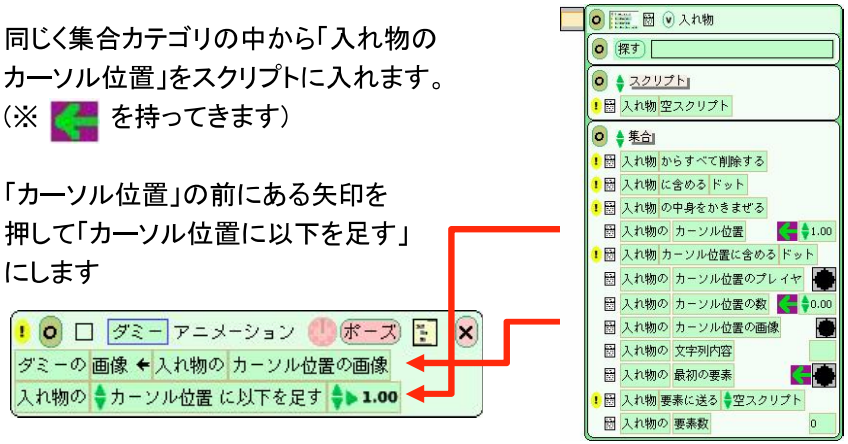
ダミーの画像カテゴリから「画像」タイルを使って
スクリプトを作ります。(※ を持ってきます)

入れ物の集合カテゴリから「入れ物カーソル位置の画像」を
スクリプトの「画像」部分に入れます

②「入れ物」のカーソル位置に1を足す。

同じく集合カテゴリの中から「入れ物の
カーソル位置」をスクリプトに入れます。
(※ を持ってきます)

「カーソル位置」の前にある矢印を
押して「カーソル位置に以下を足す」
にします



画像の処理

22 画像を変える

飛んでいる小鳥が枝に止まったときに、
画像を入れ替えます。

とり1

とり1が枝にとまると
画像がとり2変わります

とり2

準備するもの

とり1
この画像にスクリプトを組み込みます。
向きを示す矢印を木の枝の方向にしておきます。

とり2
木の枝に止まったときの画像を描いておきます。

木の枝
とり1が木の枝に触ったときに、とり2の画像を表示します。

スクリプト

とり1が木の枝に触れたときに、とり2の画像に入れ替えるスクリプトです。

【使用するカテゴリ】
とり1：　「基本」、「テスト」、「スクリプティング」、「画像」
とり2：　「画像」

スクリプトの解説

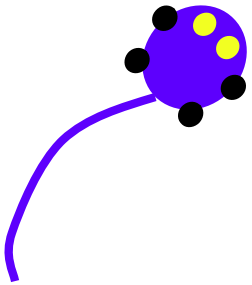
①とり1を進めます。
②とり1が木の枝の色に触ったときに、次のタイルが実行されます。
・とり1の画像に、とり2の画像を入れ替えます。
・とり1を進めるスクリプトをポーズします。
・音が鳴ります。

「とり1」の画像に「とり2」の画像を組み込みます。

画像の処理

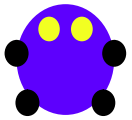
23 キャラクタが図形を描く

キャラクタが動いたところに線が引かれていきます。
角度を指定して、いろいろな図形を描いてみましょう。



キャラクタが動くと、
動いたところに線が引かれます。

準備するもの



スケッチ

動かすキャラクタを
ひとつ描きます。

「ペンの使い方」カテゴリにある「ペンが下りているか」を「はい」にしておくと、キャラクタが動いたときに、線が引かれます。

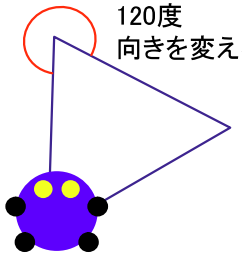
スクリプト

進んで、回る、スクリプトを作ります。
「回す」角度を工夫すると、いろいろな図形を描くことができます。

【使用するカテゴリ】
スケッチ:　　「基本」

スクリプトの解説

このスクリプトでは、スケッチを120度回すことで、正三角形を描きます。



この正三角形の描きかたを参考に、次の図形を描くときに、数値をどのように変えればいいかを考えてみましょう。

- 正方形
- 五角形
- 六角形
- 円

また、回す値を、130度、144度、150度などにするとどうなるかも試してみましょう。

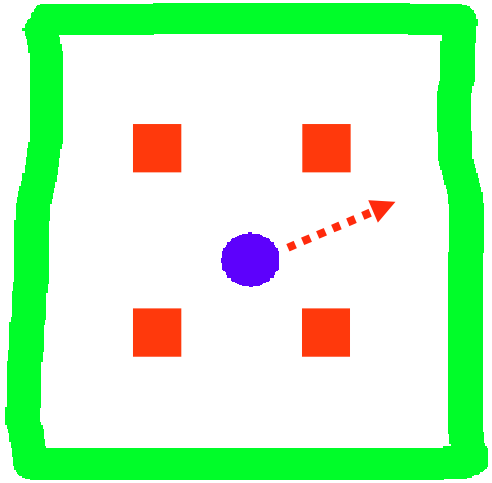
描いた図形を消すときには、「ペンの使い方」カテゴリの、「ペンの軌跡を全て消す」を使います。

! をクリック

ゲームプログラミング

24 得点をつける

ボールが赤い丸にぶつかると得点が加算される仕組みを作りましょう。



ボールが ■ にぶつかると
点数が増えます。

とくてん
30
得点 リセット

準備するもの

ボールの画像（青色）
ボールを動かすスクリプトと、得点をリセットするスクリプトを組み込みます。

ぶつかると得点が増える四角（赤色）

壁（緑色）

0

得点を表示する「テキスト」
部品から「テキスト」を引っ張り出して、最初は「0（ゼロ）」をテキストに記入しておきます。

スクリプト

赤い四角にボールがぶつかったときに、得点をプラスします。
得点をリセットするボタンも作りましょう。

【使用するカテゴリ】
ボール：　「基本」、「テスト」

スクリプトの解説

①ボールを進めます。

②ボールが、壁にぶつかったときに、100度回転させます。

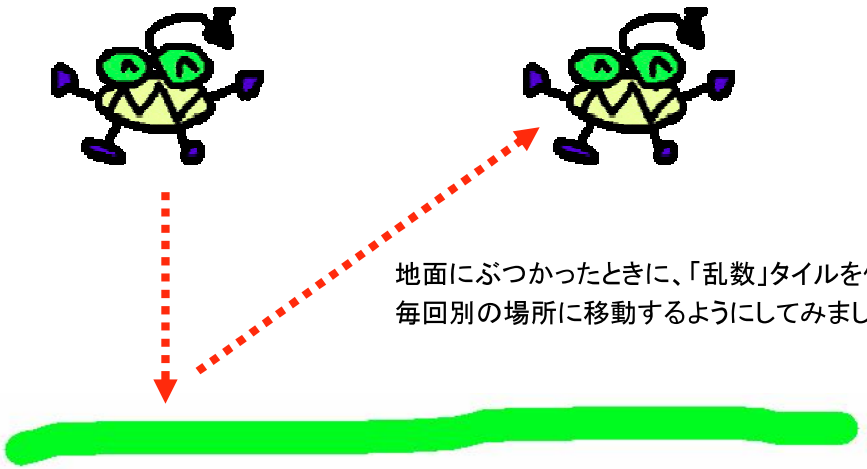
③ボールが、赤い四角にぶつかったときには、同じく100度回転させ、さらに、得点に1をプラスします。

④得点をリセットします。ボタンの作り方は「15. 簡単なボタンの作り方」参照。

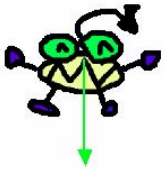
ゲームプログラミング

25 ランダムな動き

「乱数」タイルを使ってランダム(無作為)な動きをつけてみましょう。



準備するもの



インベーダー
動かしたいキャラクタを描きます。
向きを示す矢印を下向きにしておきましょう。

地面
インベーダーがぶつかったら移動する、地面を描きます。

「乱数」タイル
「乱数」タイルに「500」と指定すると、0～500のランダム(無作為)な数値をスクリプトで使うことができます。

スクリプト

インベーダーが地面にぶつかると、不規則な場所に移動します。

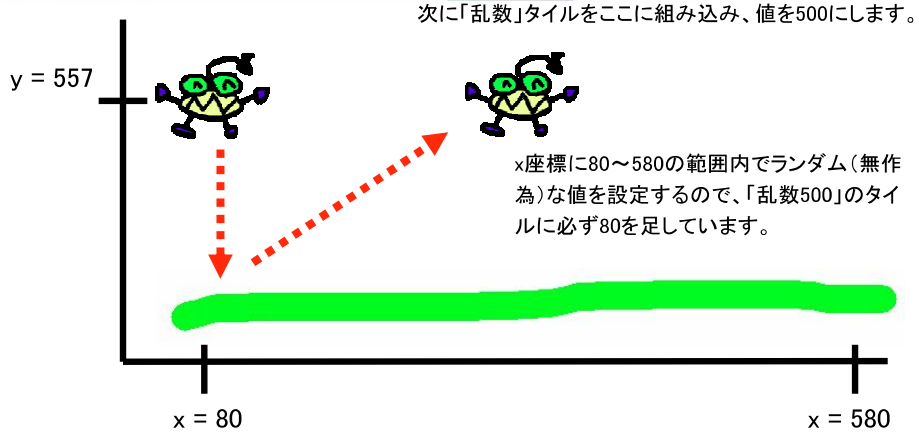
【使用するカテゴリ】
インベーダー: 「基本」、「テスト」

スクリプトの解説

インベーダーが地面の色に触れたときに、x座標(横の位置)とy座標(縦の位置)に値をセットします。x座標には「乱数」タイルを利用して、ランダム(無作為)な値を使います。

先ず ▶ をクリックして「＋1」の式を追加。「+80」にします。

次に「乱数」タイルをここに組み込み、値を500にします。

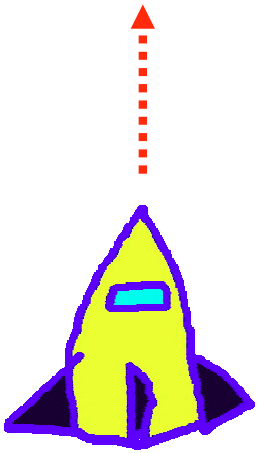


x座標に80～580の範囲内でランダム(無作為)な値を設定するので、「乱数500」のタイルに必ず80を足しています。

ゲームプログラミング

26 リセットボタン

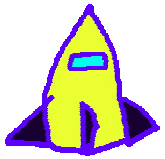
発射ボタンを押すとミサイルが発射します。リセットボタンで元の位置に戻るようにしましょう。



発射ボタンを押すと、
上に向かって動き始めます。

リセットボタンを押すと、
最初の状態に戻ります。

準備するもの



ロケットの画像

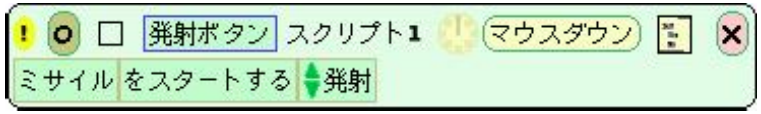
ボタンの画像
ここでは、「14. ボタンで操作する」の方法でボタンのスクリプトを作ります。

リセットボタンの画像

スクリプト

先ず、ミサイルを進めるスクリプトを作ります。
次に、そのスクリプトをスタートさせるスクリプトをボタンに組み込みます。

①

②


③

【使用するカテゴリ】
ミサイル：　「基本」、「スクリプティング」
発射ボタン：　「スクリプト」
リセットボタン：「スクリプト」

スクリプトの解説

①ミサイルを進めます。
スクリプトの名前を「発射」にしておきます。

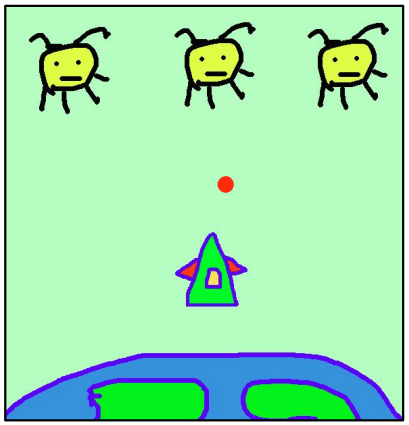
②発射ボタンをクリックしたときに、スクリプト「発射」をスタートさせます。
（「マウスダウン」のときにスクリプトが動きます）

③リセットボタンをクリックしたときに、ロケットを最初の位置（x=208、y=194）に移動してから、スクリプト「発射」を停止（ポーズ）します。

ゲームプログラミング

27 シューティングゲーム

インベーダーから地球を守るシューティングゲームを作ってみましょう。上級者向けだから少し難しいですよ！



迫り来るインベーダーから地球を守ります。
ロケットを操縦して、ミサイルでインベーダーを倒しましょう。

次のいずれかの場合、ゲームオーバーです。
(1)インベーダーが地球を侵略
(2)ロケットがインベーダーに衝突

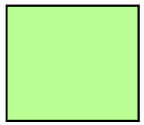
準備するもの

「遊び場」と「ボタン」を「部品」フラップから引っ張り出します。
インベーダー、ロケット、ミサイル、地球、ボタンは、全て、「遊び場」の中に組み込みましょう。「遊び場」が、このゲームのプレイフィールドになります。

インベーダ

向きを示す矢印を下向きにしておきましょう。

ボタン


遊び場

地球

ロケット

ミサイル

ロケット

ロケットをマウスで動かす

ロケットのx座標とy座標に、「遊び場」のマウスのx座標とy座標を代入することで、マウスの動きにあわせて、ロケットが動きます。

【使用するカテゴリ】
ロケット:　「基本」
遊び場:　「遊び場」

ミサイルを発射する

ロケットをマウスでクリックしたときに、ミサイルを「初期化」してから、ミサイルを動かすスクリプトをスタートさせます。

【使用するカテゴリ】
ミサイル:　「スクリプティング」

ミサイルを初期化する

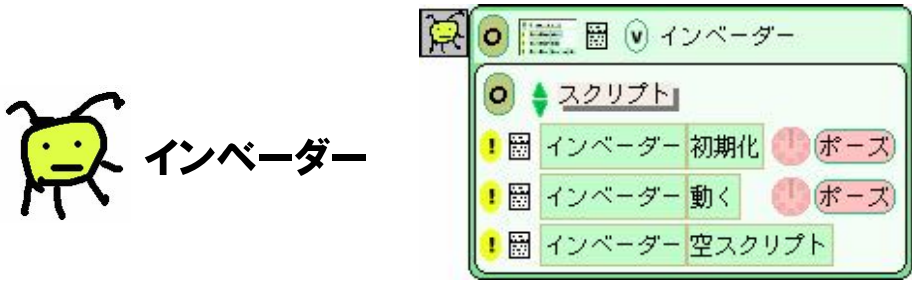
ミサイルを発射する直前に、隠していたミサイルを表示して、ミサイルの最初の位置をロケットの現在の位置にします。

【使用するカテゴリ】
ミサイル:　「基本」、「その他」
ロケット:　「基本」

ミサイルを動かす

ミサイルを動かして、画面をはみ出したら(y座標>650)、スクリプトをとめて、ミサイルを隠します。

【使用するカテゴリ】
ミサイル:　「基本」、「スクリプティング」、「その他」



インベーダーを初期化する

インベーダが最初に現れる位置を、乱数を使ってしています。

【使用するカテゴリ】
インベーダー:　「基本」　　※乱数タイルは部品フラップから

インベーダーを動かす

インベーダーがミサイルと重なったら、インベーダーを初期化します。また、地球と重なった場合には、インベーダーとロケットのスクリプトをポーズします。

[ミサイル]のタイルはミサイルのハロをクリックすると出てきます。

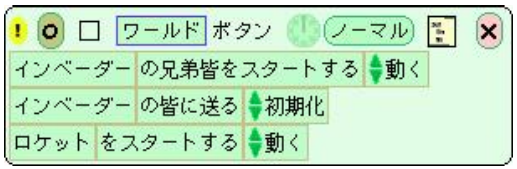
【使用するカテゴリ】
インベーダー:　「基本」、「テスト」、「スクリプティング」
ロケット:　「スクリプティング」

スタートボタン

ボタンのスクリプトを作る

ボタンのハロの をクリックするとスクリプターが出てきます。

ゲームをスタートするためのスクリプトをボタンに、組み込みます。

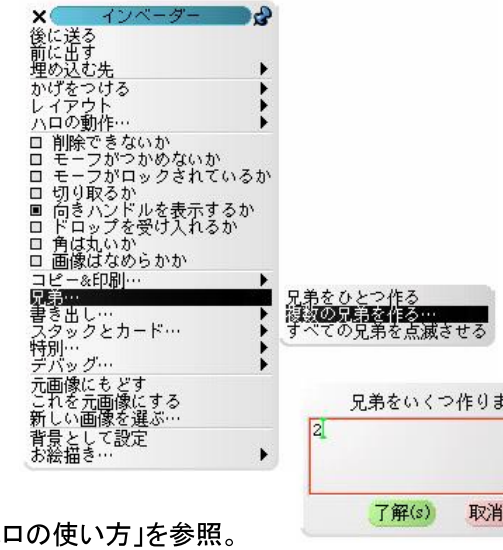


【使用するカテゴリ】
インベーダー: 「スクリプティング」
ロケット: 「スクリプティング」

ボタンのハロ (メニュー)をクリックして[ラベルを変更]を選ぶと、ボタンの名前を変更できます。

インベーダーの「兄弟」を作る

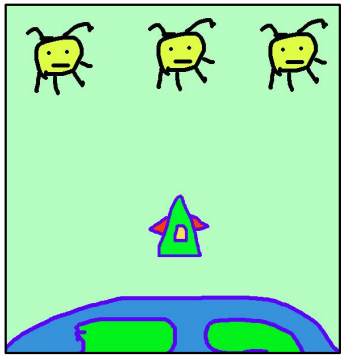
インベーダーのメニュー をクリックして、「兄弟」を2つ作ります。



「兄弟」については「3. ハロの使い方」を参照。

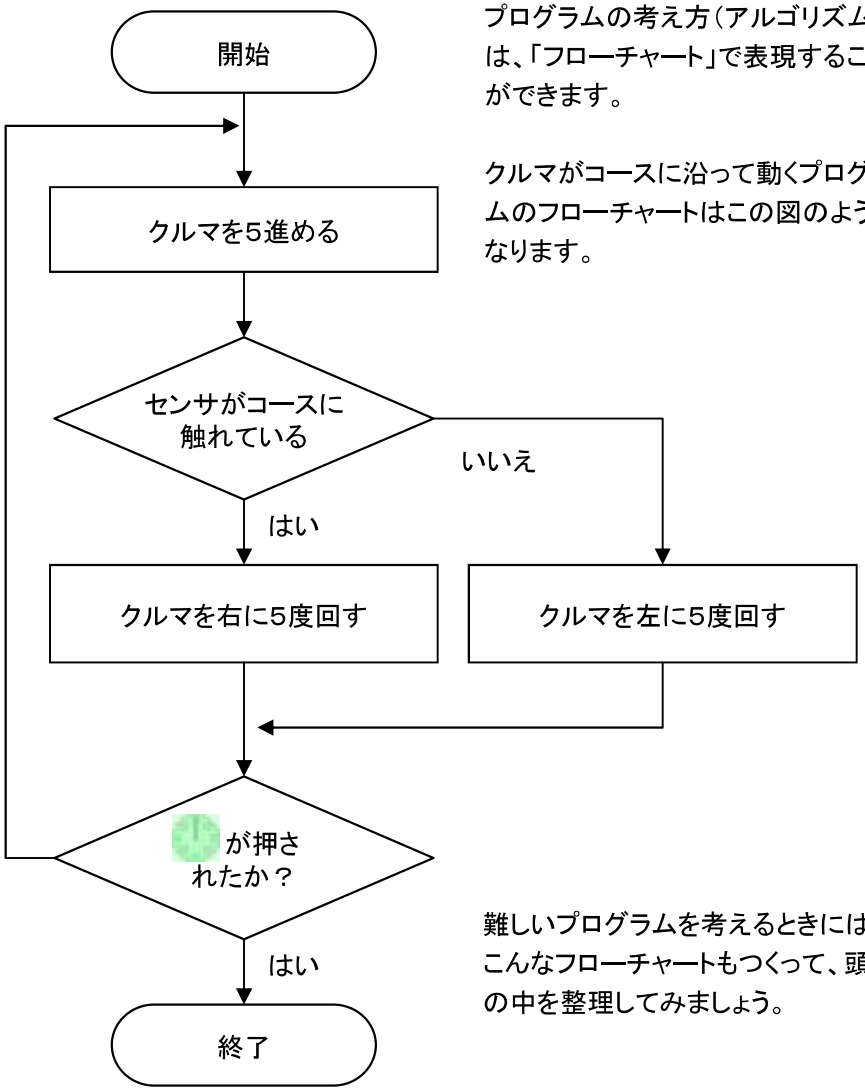
作ったゲームを改良してみよう！

シューティングゲームが完成したかな？
もっと使いやすく改良してみよう！
(1)得点をつける。
(2)音を出す。
(3)得点が増えるに従って、インベーダーがだんだん早く動く。
(3)キャラクタをアニメーションで表現する。
・・・などなど



もっと楽しいゲームにするにはどうしたらいいか
工夫してみましょう！

フローチャート（流れ図）について



プログラムの考え方（アルゴリズム）は、「フローチャート」で表現することができます。

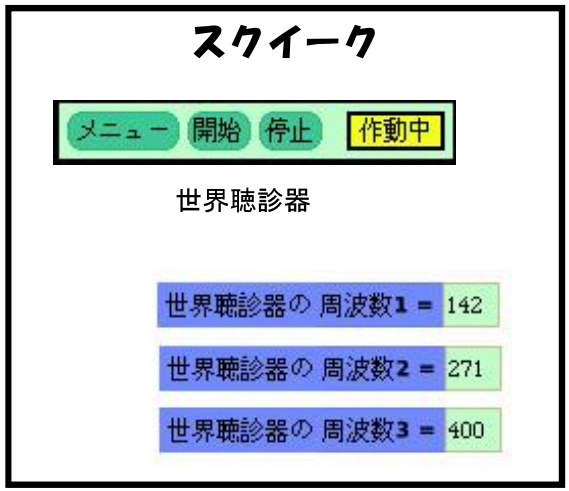
クルマがコースに沿って動くプログラムのフローチャートはこの図のようになります。

難しいプログラムを考えるときには、こんなフローチャートもつくって、頭の中を整理してみましょう。

第3章　世界聴診器

28 世界聴診器の使い方

スクイークに世界聴診器を組み込むと、音を周波数を使った実験ができます。



いろんな音の周波数を
調べてみましょう。

■世界聴診器の部品を使うために

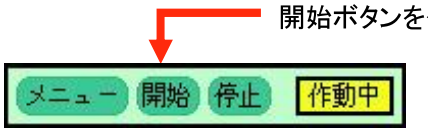
世界聴診器の部品を利用するときには、スクイーク本体と世界聴診器専用ファイルがインストールされている必要があります。
世界聴診器の専用ファイルはこちらからダウンロードしてください。

世界聴診器をめぐる冒険
http://swikis.ddo.jp/WorldStethoscope/2

世界聴診器専用ファイルはダウンロードしたZIPファイルを展開してできる「SqueakPlugin.image」を任意の場所にコピーし、そのファイルをダブルクリックすることで、世界聴診器用のスクイークを起動することができます。

■世界聴診器の基本操作

「部品」フラップから「世界聴診器」を引っ張り出しましょう。
「開始」ボタンを押すと、動き始めます。



開始すると「作動中」
の表示が黄色になります。

■周波数、電話番号、音量について

パソコンにマイクをつないで、世界聴診器を「開始」しましょう。世界聴診器のビューワから「サウンド」カテゴリを表示してみてください。
「周波数」や、「音量」の数値が、めまぐるしく変化しているのがわかりますか？

音が伝わるのは、空気が振動しているからです。この空気の振動を数値で表したものを、「周波数」といいます。周波数は、ヘルツ(Hz)という単位で表され、1秒間に繰り返される振動(波)の数のことを指しています。
一般に、人間の耳に聞こえる音の高さは、100Hz～10KHz(約10,000Hz)だといわれています。

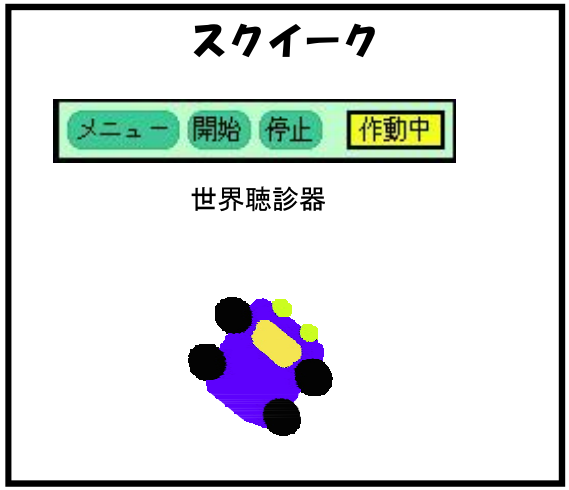
①周波数1～3
聞こえている音を分析して、信号の大きい周波数を順番に1から3まで取り出します。

②電話番号
電話の「ピポパ」音が聞こえると、数字の何番が押されたかを表示します。
携帯電話でためしてみましょう。

③音量1～3
音の大きさを数値で表示します。

29 口笛で操作するクルマ

世界聴診器の部品を利用して、音に反応して動くクルマを作ってみましょう。



口笛でクルマを操作します。

準備するもの

クルマ
（動かしたい画像）

世界聴診器（部品から引っ張り出しておきます）

世界聴診器の「サウンド」カテゴリで、自分の口笛の周波数の数値を確認しておきます。（周波数1の値）

スクリプト①

先ず最初に、音に反応して、右に曲がるスクリプトを作ってみましょう。
口笛の音（周波数が1500～1800くらい）がしたときにクルマを右に5度回します。

【使用するカテゴリ】
クルマ　・・・　「基本」、「スクリプティング」
世界聴診器　・・・「サウンド」

スクリプト②

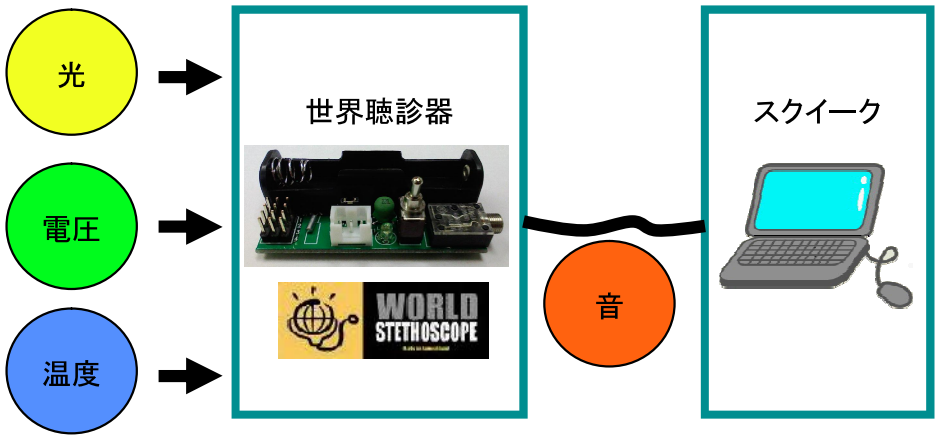
今度は、右にも左にも曲がるようにしてみましょう。
高い音の口笛（周波数が1500～1800くらい）のときには、右へ5度回し、低い音の口笛（周波数が1100～1300くらい）のときには、左へ5度回します。

スクリプトの解説

テストタイルで口笛の音の高さ（周波数1）を判断して、音がしたときに、その音の高さの値に応じてクルマを回します。

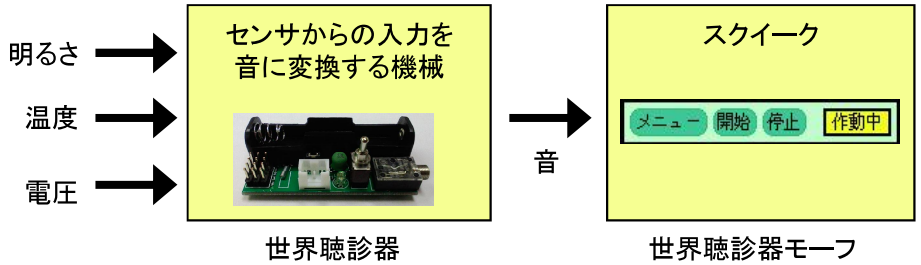
30 世界聴診器とは？

世界聴診器を使うと、光や電圧、温度の変化を音に変換して、スクイークに取り込むことができます。



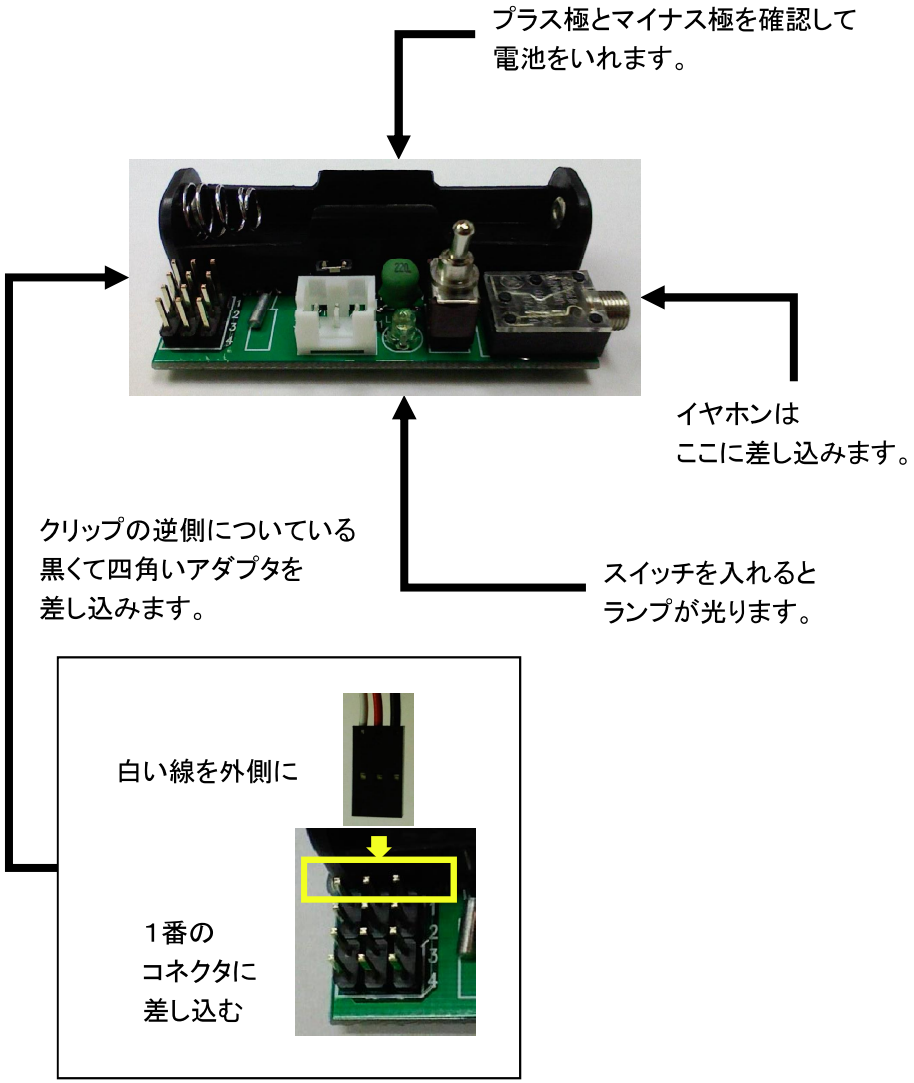
■世界聴診器の仕組み

世界聴診器は、光や電圧を音に変換する機械（ハードウェア）と、スクイーク上で、音の周波数をスクイークに取り込むための世界聴診器モーフ（ソフトウェア）から構成されます。



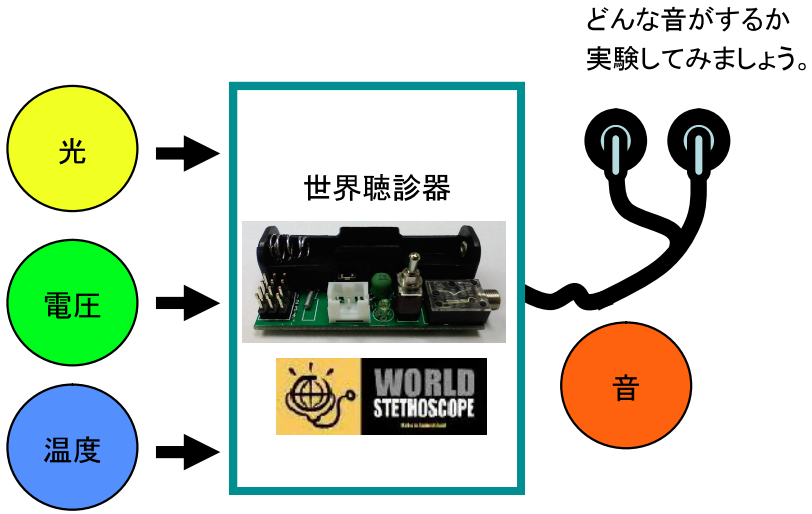
■世界聴診器の組み立て

①イヤホンとクリップ、電池をセットします。
②スイッチを入れて赤ランプが点灯することを確認します。
③片方のイヤホン（R側）から、プププという音がきこえることを確認します。



31 世界聴診器を使った実験

世界聴診器の、電圧センサや光センサを使って、いろいろな実験をしてみましょう。



■大事な注意点

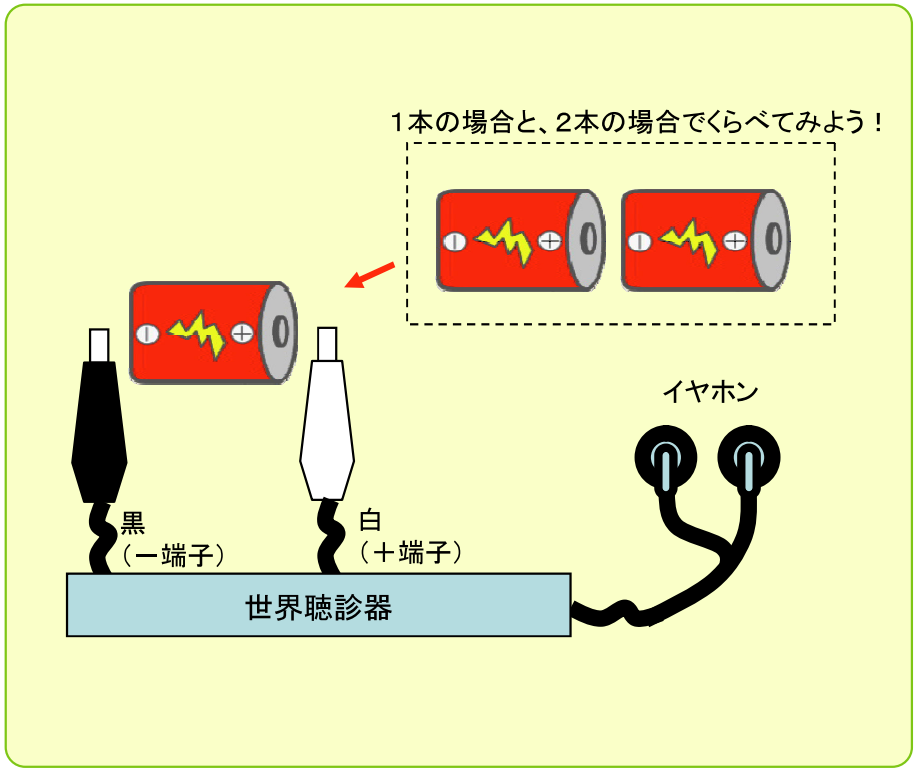
部品をなくさないで！
世界聴診器には、小さな部品もあるので、なくさないように気をつけてください。

作業するときは電源オフ！
世界聴診器を組み立てたり、分解したりするときには、必ず電源をオフにして作業をしてください。

終わったら元に戻して返却！
世界聴診器を使い終わったら、元のとおりに袋にしまってから返却してください。

■電圧センサ

電池の＋極に白いクリップを、－極に黒いクリップをつないでから、イヤホンの音を聞いてみましょう。
次に、電池を2つにするとどうなるかな？　電池がひとつの時と、2つのときで、音の高さに変化があるかどうか調べてください。

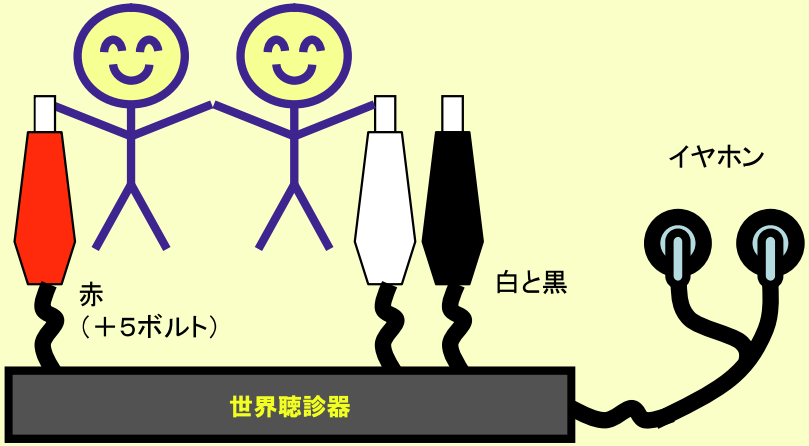


解説

電圧っていうのは、「電気を流す力」のこと。乾電池の電圧は、だいたい1．5ボルトが標準です。2本の電池を直列につないだときは、電圧も2倍になるので、世界聴診器から出てくる音の高さも倍になります。（1オクターブ上）

■体の中を流れる電気

体の中を電気が流れるかを試してみましょう。右手に赤、左手に黒のクリップをもって、音の変化を確かめてみよう。図のように、ふたりでやったらどうなるかな？水道水、雨水、雪、蒸留水などで電気の通し方(電気抵抗)を調べるのも面白い実験です。



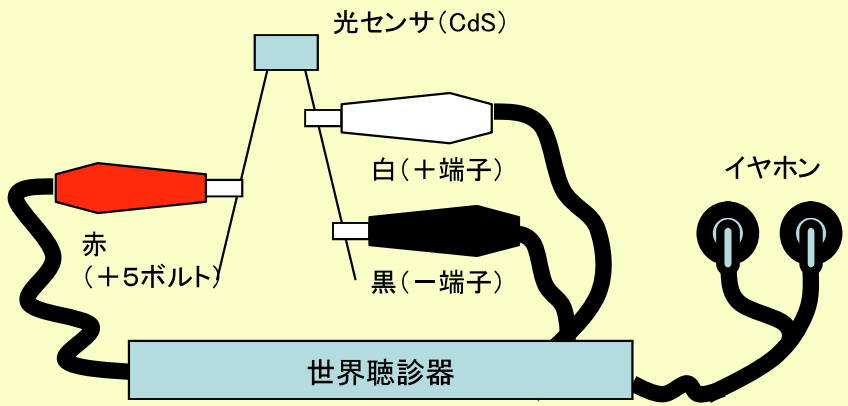
解説

人間の体は70%が水分でできています。純粋な水(純水)は電気を通しませんが、体の中の水分や水道水などにはいろいろな物(イオン)が溶けています（電解液といいます）。電解液は電子を運ぶ性質があるため、電気を通すのです。

31. 世界聴診器を使った実験

■光センサ

光センサ（CdS）を図のように世界聴診器と繋げたら、周りの明るさによって、どんな風に音が変化するかを試してみましょう。
真っ暗ななかに光センサを置いたり、懐中電灯で光センサを照らしてみて、音の高さがどう変わるかな？

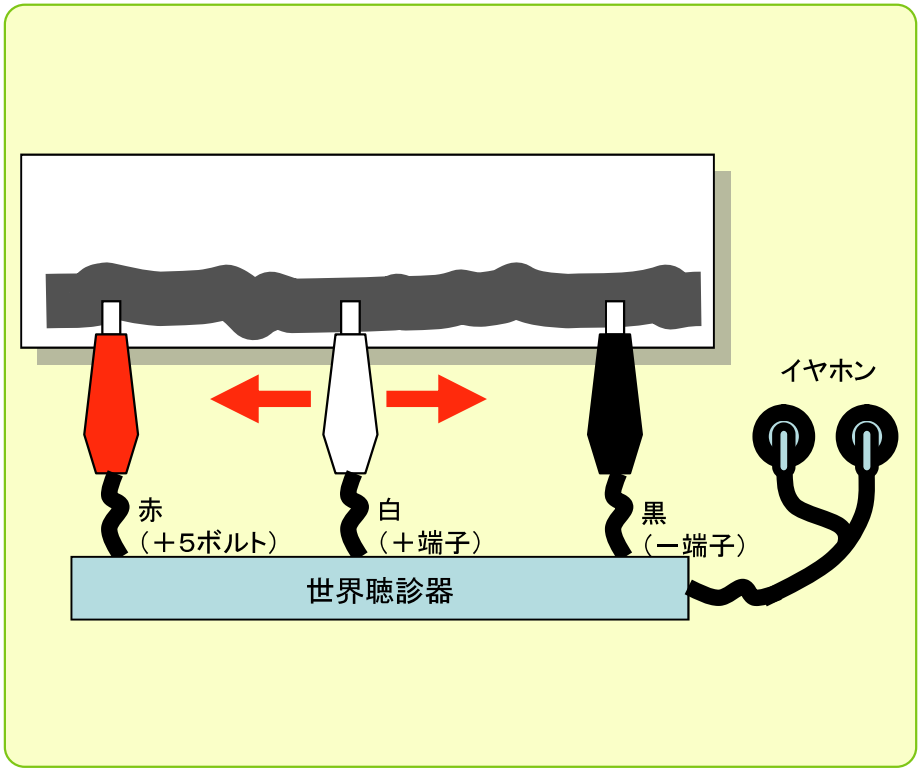


解説

光センサは、暗ければ暗いほど電気抵抗（電気の流れにくさ）が大きくなるという性質をもっているので、明るければ電気が流れやすく、暗いときは電気が流れにくくなります。

■えんぴつオルガン

紙の端に沿って、長さは約10cmくらいの鉛筆で太い線を引きます。線が途切れないように、できるだけ太く、濃く書いてください。
下図のとおり接続して、白い端子を左右に動かして、音を聞いてみてください。
すると、白い端子の位置に応じた高さの音が聞こえます。



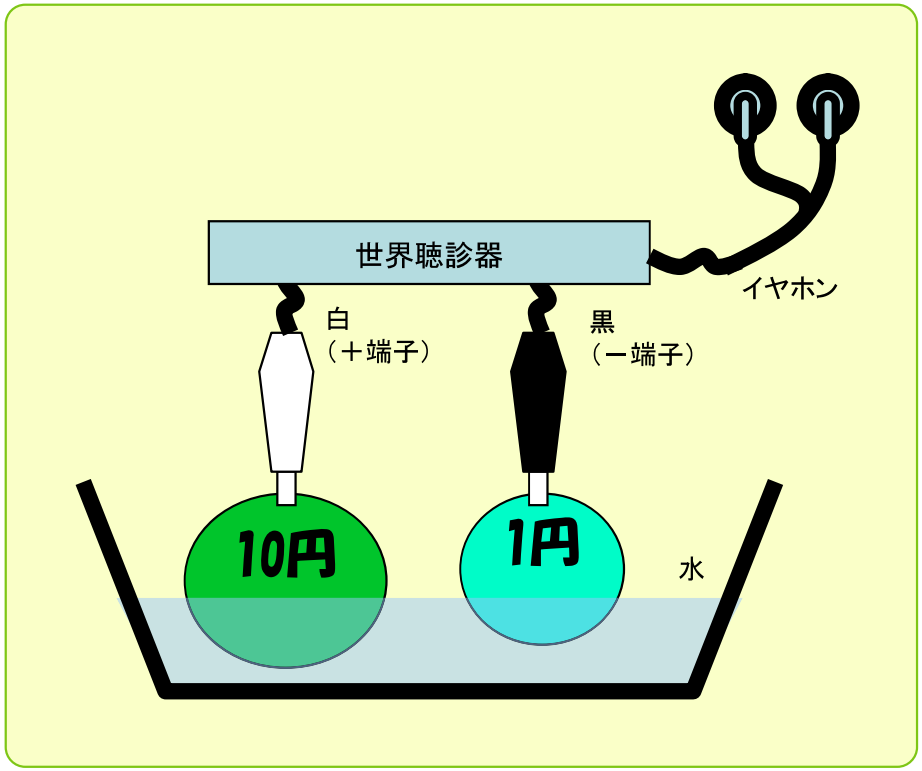
解説

鉛筆の芯は、電気を通す炭素と通さない粘土からできています。「HB」などの「H」の数が増えると炭素の比率が増え、「B」の数が増えると粘土の比率が増えます。これらの違いを調べても面白いでしょう。

31. 世界聴診器を使った実験

■11円電池

水をいれたコップを用意します。10円玉と1円玉を白と黒のクリップを使って水につけると、電池ができます。
図のとおりに、水につけてみましょう。どんな音がするでしょう？
音の高さは乾電池のときと比べてどうですか？

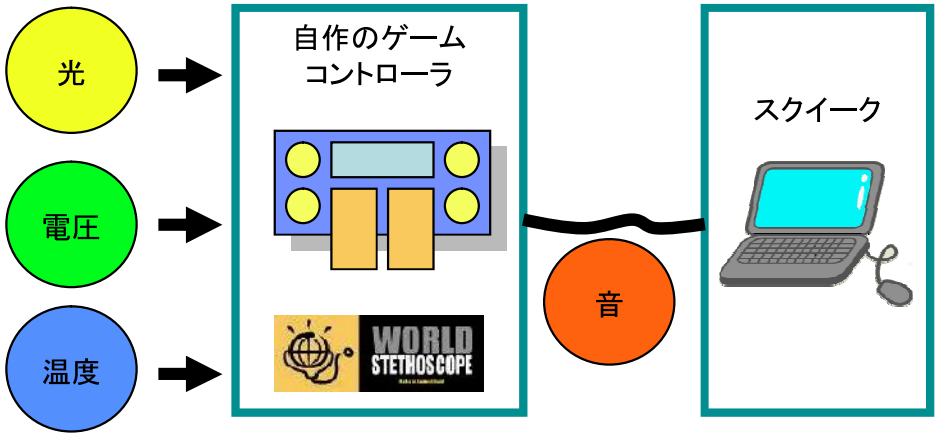


解説

1円玉のアルミニウムと、10円玉の銅、そしてコップにある水が反応して電池になりますが、乾電池と比べたら低い音がすることから、電圧が少ししかないことがわかります。
水のかわりに、野菜や果物も使ってみましょう。（レモンがお勧め）

32 ゲームコントローラを作る

世界聴診器を使えば、自分だけのゲームコントローラがつくれます。どんなコントローラができたのか見てみましょう。



世界聴診器を使って
自作ゲームコントローラを作ります。

■用意するもの

身の回りにあるいろいろなものを組み合わせて、世界聴診器で自分だけのゲームコントローラを作ります。

- 紙コップ
- ボール紙
- アルミホイル
- 紙のお皿
- ストロー
- 電池
- 懐中電灯
- セロテープ
- ガムテープ
- ホッチキス
- はさみ
- カッター
- 画用紙
- セロファン
- 磁石
- 針金
- 風船
- 洗濯ばさみ

などなど、いろいろなものを使って工夫してみましょう。

■ゲームコントローラの例

世界聴診器の光センサや電圧センサを使って、どんなゲームコントローラを作ることができるかみてみましょう。

ジョイスティック型コントローラ

懐中電灯を箱に差し込んで、上下に動かします。箱に組み込んだ光センサで明るさの変化を読み取って、画面のキャラクタを動かします。

びっくり箱型コントローラ

紙コップにいれたスポンジを引っ張り出すと、画面のキャラクタも同じ動きをします。紙コップの底に光センサを組み込み、明るさの変化を読み取っています。

人間型コントローラ

紙で作った人間の手足にクリップ（電極）をつなげると、画面上の女の子がダンスを始めます。手足につないだ電極の電圧の変化を利用しています。

むしめがね型コントローラ

むしめがねを画面の虫に近づけると、虫が大きく表示されます。むしめがねに貼り付けた光センサが画面の明るさを読み取って、明るくなるとズームします。

グローブ型コントローラ

アルミホイルで作ったグローブでペットボトルをキャッチすると、画面のキャラも同じ動きをします。グローブとペットボトルに流れる電圧の変化を利用しています。

このテキストは、IT人材の早期発掘・育成を目的とするITクラフトマンシップ事業の一環として、経済産業省の支援を受けて制作されました。
教材制作やワークショップの開催にご協力いただいた多数の皆様に、この場をお借りして御礼申し上げます。

主催：
・デジタルハリウッド大学院／デジタルハリウッド大学
・経済産業省
・財団法人 日本情報処理開発協会

協力：
・いずみこどもプラザ
・HPスーパーサイエンスキッズ実行委員会
・SM Foundation, Nepal
・SeaGate Institute of Technology, Nepal
・株式会社アクティブラーニング
・株式会社オブジェクトディメンション
・社団法人 学術・文化・産業ネットワーク多摩
・特定非営利活動法人先端教育情報研究所
・特定非営利活動法人グローバルプロジェクト推進機構（JEARN）
・西日本電信電話株式会社（NTT西日本）
・有限会社アユムーンデジタルスターズ

スクイークで学ぶゲームプログラミング
～世界聴診器をめぐる大冒険～

2007年2月28日　初版発行

制作著作　：　デジタルハリウッド大学大学院
みんなでたのしくスクイーク・プロジェクト
http://mts-j.hiho.jp/

本書の内容に関するお問い合わせはこちらまでeメールにてお願いします。
dhsqueakers@gmail.com

Copyright　2005-2007 MTS-Project, Digital Hollywood Graduate School. All rights reserved.