

KASAI.SOFTWARELAB

小・中学生のための
Scratch
プログラミング入門

スクラッチ

河西朝雄著

定価500円+税

小・中学生のための Scratch プログラミング入門

河西 朝雄著

カサイ. ソフトウェアラボ

はじめに

■Scratch（スクラッチ）とは？

- ・米国のマサチューセッツ工科大学で子ども向けに開発されたプログラム開発環境です。
- ・Scratch の Web サイトにサインインするだけで特別な開発環境をインストールする必要がありません。
- ・一般的のプログラミング言語では、多くの「命令」を覚えて、コードを打ち込み、プログラムを書きます。
- ・一方、Scratch では「スクリプト」というブロックをつなぎ合わせることでプログラムを書きます。
- ・操作方法は基本的にマウスによるドラック＆ドロップであるため、比較的簡単にプログラミングができます。

■Scratch プログラミングを学ぶことによって何が得られるのか？

- ・プログラミングという作業を通じて問題解決能力、創造力、論理的思考が高まります。
- ・そうした力を養うことにより学習に対する意欲の向上や、自ら進んで粘り強く学ぼうとする学習態度が養われます。
- ・これはアクティブラーニングという考え方にも通じます。
- ・プログラミング能力は、これからの中間化社会を生きて行くための「教養」です。読み書きの能力を身に付けること同様に重要です。

■教育現場でも注目が集まっています

- ・プログラミングは、中学校の「技術・家庭」や高校の教科「情報」で教えられていますがあまり普及していません。
- ・アメリカをはじめ世界各国ではプログラミング教育の機運が高まっており、日本もその流れに乗ろうとしています。
- ・そこで、次期学習指導要領が始まる 2020 年度からコンピューターのプログラミング教育を小・中学校で必修にする計画です。
- ・Scratch は操作が視覚的で分かり易いので、小・中学生にプログラミングの基礎を教えるのに最適です。

■2021 年度からの大学入試に対応

- ・2021 年度から大学入試制度が大きく変わります。
- ・習得した「知識・技能」の確認に留まらず、「知識・技能を活用する力」、つまり総合的な「思考力・表現力」を問う検査が重要になります。
- ・こうした思考力・表現力を身につけるにはプログラミングは最適な教材と言えます。

本書は小・中学生が Scratch を使ってプログラミングを学ぶためのもので、以下のような 8 回の構成になっています。

- 第1回 Scratch でねこを動かしてみよう
- 第2回 アニメーションに挑戦
- 第3回 図形を描いてみよう
- 第4回 プログラムのデータを管理する変数
- 第5回 データをまとめて管理するリスト
- 第6回 プログラムを制御する
- 第7回 同じ処理をひとつにまとめるブロック
- 第8回 データ構造とアルゴリズム

第1回～第3回はアニメーションや図形を描くなどの視覚的に興味が湧く題材を扱っています。第4回～第7回はプログラミングをする上で必要な基本事項である、変数、リスト（配列）、流れ制御、ブロック（関数、メソッド）について説明しています。第8回はプログラムを使って問題を解くためのアルゴリズムや（algorithms：算法）やデータ構造（data structure）について説明しています。

小・中学生にとってはかなり高度な内容になっていますが、分かり易く説明しているので理解できるはずです。

学校での学習に使う場合は総合学習の時間やクラブの時間を使って 45 分 × 8 回で学習できるようになっています。パソコン教室などのテキストとしても使えます。

本書を通して多くの小・中学生が、これから的情報化社会を生きて行くための「教養」としてのプログラミング能力を身につけてもらえたなら幸いです。

2016 年 7 月 河西 朝雄

目次

第1回 Scratchでねこを動かしてみよう	7
1. Scratchを始めるには?	8
2. Scratchの画面	11
3. ねこを動かしてみよう	12
4. プログラムを作ってみよう	13
5. 何度もくりかえすには?	14
6. 何かしやべってみよう	15
7. 大きさを変えてみよう	17
8. ねこの代わりに恐竜を使ってみよう	19
9. コスチュームを変えてみよう	20
第2回 アニメーションに挑戦	23
1. こうもりを飛ばそう	24
2. ダンスを踊ろう	28
3. 運動会	31
第3回 図形を描いてみよう	35
1. 指定した座標にねこを移す	36
2. アルファベットを描く	41
3. 一筆書きできないアルファベットを描く	43
4. 多角形を描いてみよう	46
☆coffee break 湧巻き	50
第4回 プログラムのデータを管理する変数	51
1. 変数とは	52
2. マウス位置を追いかけてゴーストが移動	55
3. ねこを変数を使って移動	58
4. n角形を描く	60
5. グラフを描いてみよう	63
☆coffee break 3次元関数の3D表示	66

第5回 データをまとめて管理するリスト	67
1. リストとは	68
2. ギガの自己紹介	72
3. リストに音のデータを格納	74
4. リストに移動データを格納	78
☆coffee break 迷路	80
第6回 プログラムを制御する	81
1. 制御スクリプトの種類	82
2. ボールを変数を使って跳ね返す	84
3. ボールをラケットで跳ね返す	87
第7回 同じ処理をひとつにまとめるブロック	91
1. ブロックとは	92
2. 3角形を描くブロック triangle を作ってみよう	94
3. n角形を描くブロック polygon を作ってみよう	96
4. 指定した2点間で直線を描くブロック line を作ってみよう	99
5. 数学関数で計算	100
☆coffee break リバーシー	106
第8回 データ構造とアルゴリズム	107
1. 決定木	108
2. ハノイの塔	111
3. リカーシブ・グラフィックス	114
☆coffee break じゃんけん	118

第1回 Scratch でねこを動かしてみよう

Scratch でプログラミングする上で必要な基本事項を説明します。

1. Scratch を始めるには？

最初に Scratch を始めるためのユーザー登録の仕方やプログラムを作る方法を説明します。

2. Scratch の画面

Scratch でプログラムを作る時の主役は「ねこ」です。

3. ねこを動かしてみよう

この「ねこ」を「スクリプト」を使って動かします。

4. プログラムを作ってみよう

スクリプトを画面右のスクリプトエリアにドラッグドロップしてプログラムを作ります。

5. 何度もくりかえすには？

スクリプトの「制御」を使えば同じことを何度も繰り返すことができます。

6. 何かしらやってみよう

スクリプトの「音」を使えばしゃべることができます。

7. 大きさを変えてみよう

スクリプトの「見た目」を使えば大きさを変えることができます。

8. ねこの代わりに恐竜を使ってみよう

ライブラリから別のスプライトを選ぶことができます。

9. コスチュームを変えてみよう

コスチュームはスプライトの色々な容姿です。

1. Scratch を始めるには？

■Scratch の Web ページからユーザー登録します

<https://scratch.mit.edu/>

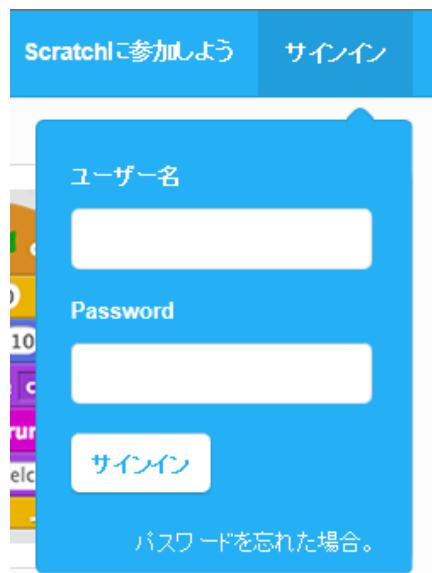
- ・「Scratch に参加しよう」を選択します



- ・登録するための質問に答えます

The screenshot shows the account creation form. It has three input fields: 'Scratchで使うユーザー名を入力' (Input your Scratch username), 'パスワードを入力' (Input your password), and 'パスワードの確認' (Confirm your password). Below the fields is a large orange Scratch cat icon. At the bottom, there is a progress bar with steps 1 through 4, where step 1 is highlighted in orange, and a '次へ' (Next) button.

- ・次回からユーザー名とパスワードサインインします



■プログラムを作る

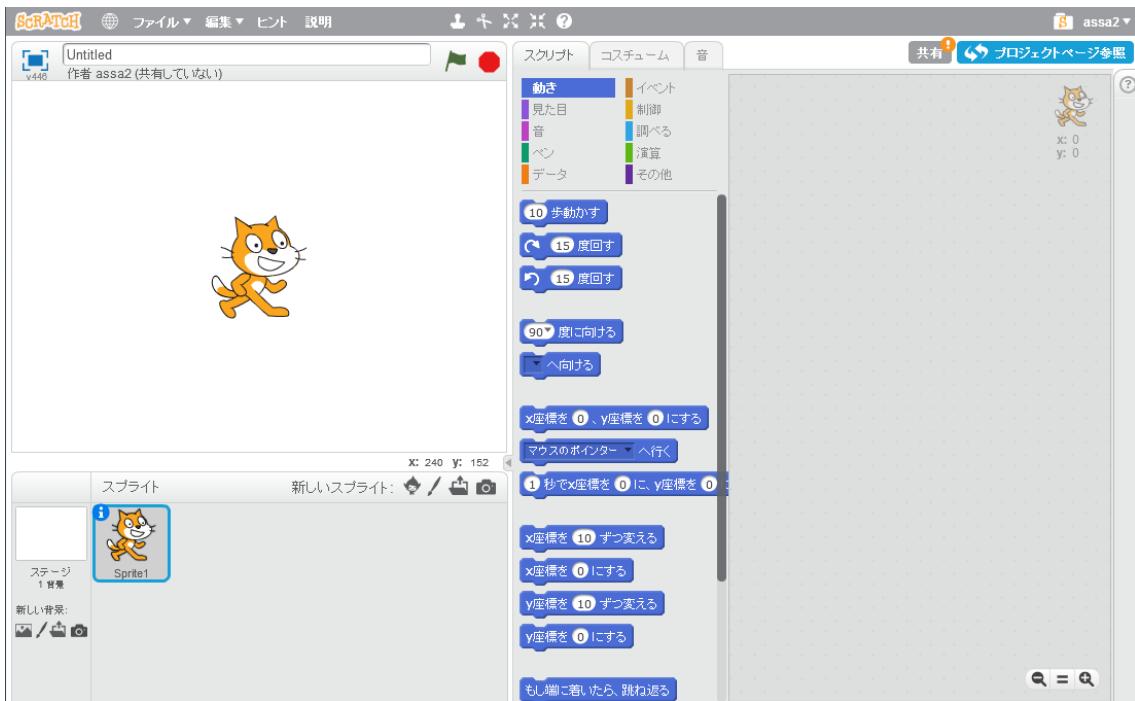
- ・「私の作品」を選択します



- ・「新しいプロジェクト」を選択します



- Scratch でプログラミングする画面が表示されます



第2回 アニメーションに挑戦

アニメーションのサンプルとして以下を作ります。

1. こうもりを飛ばそう

こうもりを画面の左から右へコスチュームを変えながら移動し、端で跳ね返します。

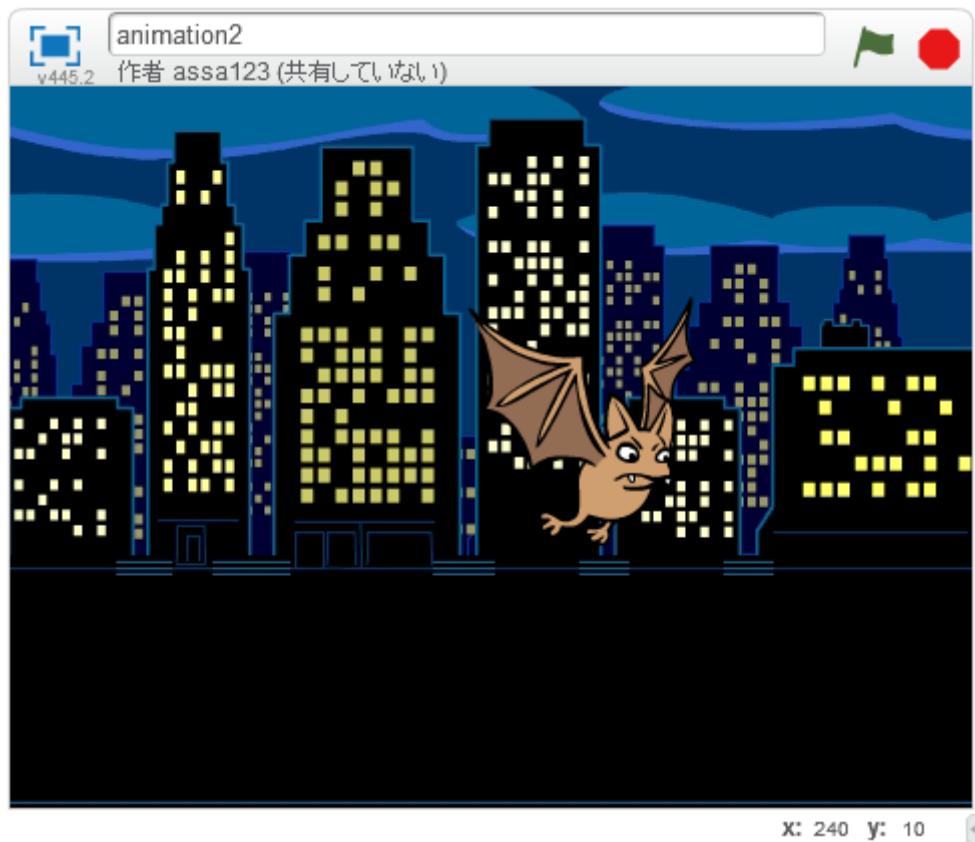
2. ダンスを踊ろう

バレリーナがリズムに合わせてステージでダンスします。

3. 運動会

カブトムシがトラックを回る運動会です。

1. こうもりを飛ばそう

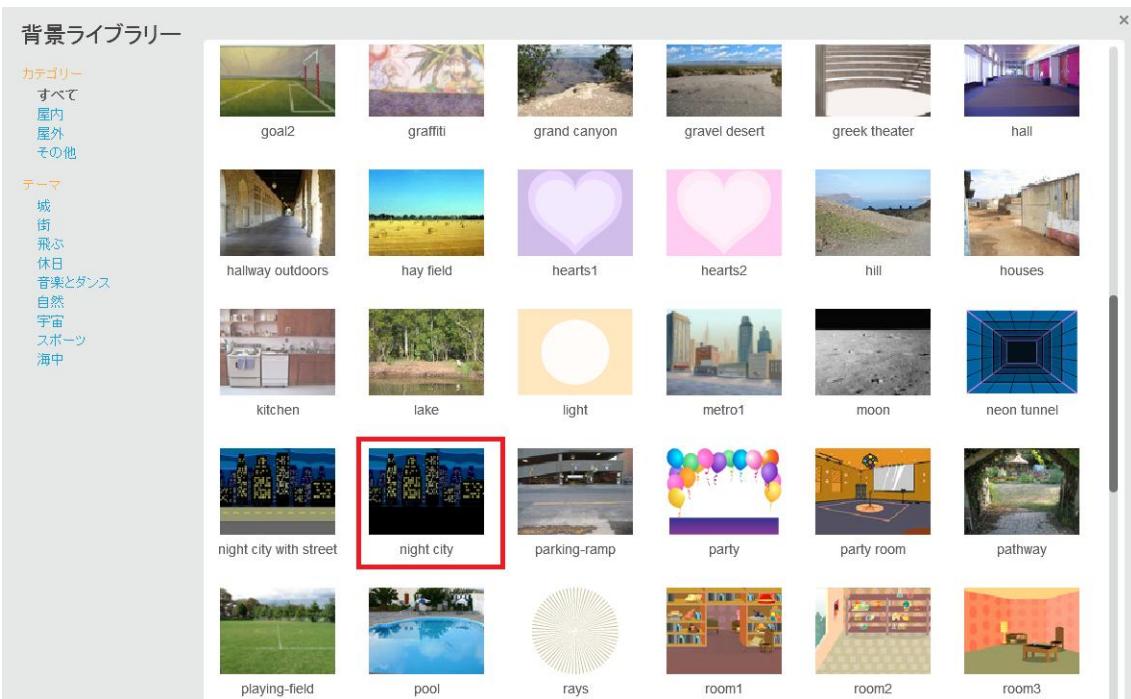


■背景を変える

- ・新しい背景を選択



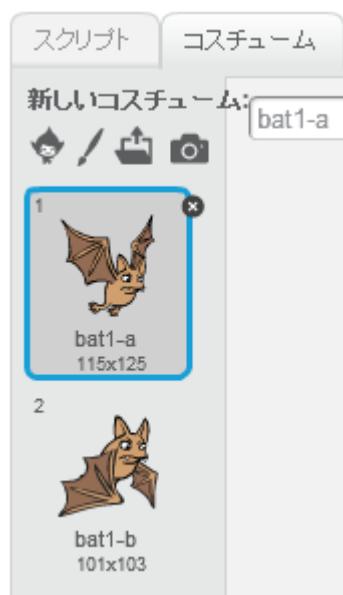
- ・「背景ライブラリー」から night city を選択



■スプライトをこうもり (Bat1) に変えます



- ・Bat1 のコスチュームは2種類

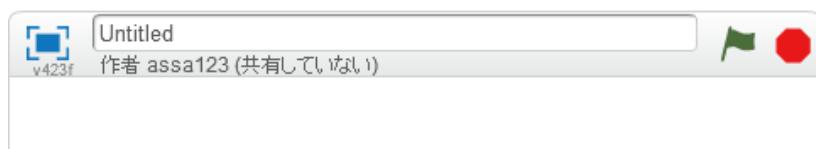


■スプライトが端に来たら跳ね返すには以下のスクリプトを使います

もし端に着いたら、跳ね返る

■プログラムの開始と停止

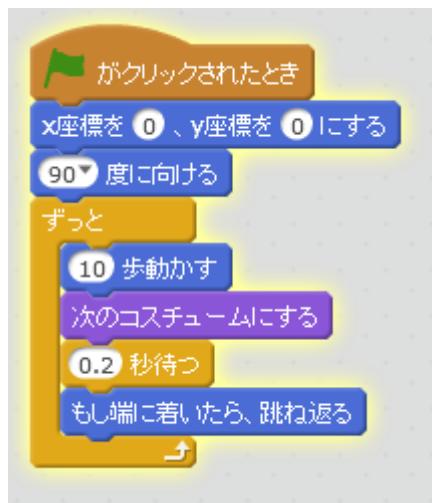
- ・プログラムの開始はスクリプトエリアにドラッグドロップしたブロックをクリックすることで行えました。
- ・しかし、一般的なプログラムの開始と停止は、画面右上の「緑の旗」と「赤色の八角形」のクリックで行います。



- ・「緑の旗」のクリックでプログラムを開始するには、「スクリプト」の「イベント」にある  をクリックでスクリプトエリアにドラッグドロップします。



■こうもりを画面の左から右へコスチュームを変えながら移動し、端で跳ね返すするスクリプト



☆やってみよう

- こうもりが跳ね返ったときスプライトの左右と上下の向きが入れ替わってしまいます。
左右だけを入れ替えるには、繰り返しに入る前に以下のスクリプトを使います。



第3回 図形を描いてみよう

ペン機能を使ってステージに図形を描画します。

1. 指定した座標にねこを移す

ねこの位置は x 座標、y 座標で指定します。

2. アルファベットを描く

N や M などの一筆書きできるアルファベットを描きます。

3. 一筆書きできないアルファベットを描く

A や B は一筆書きできないのでいったんペンを上げて移動します。

4. 多角形を描いてみよう

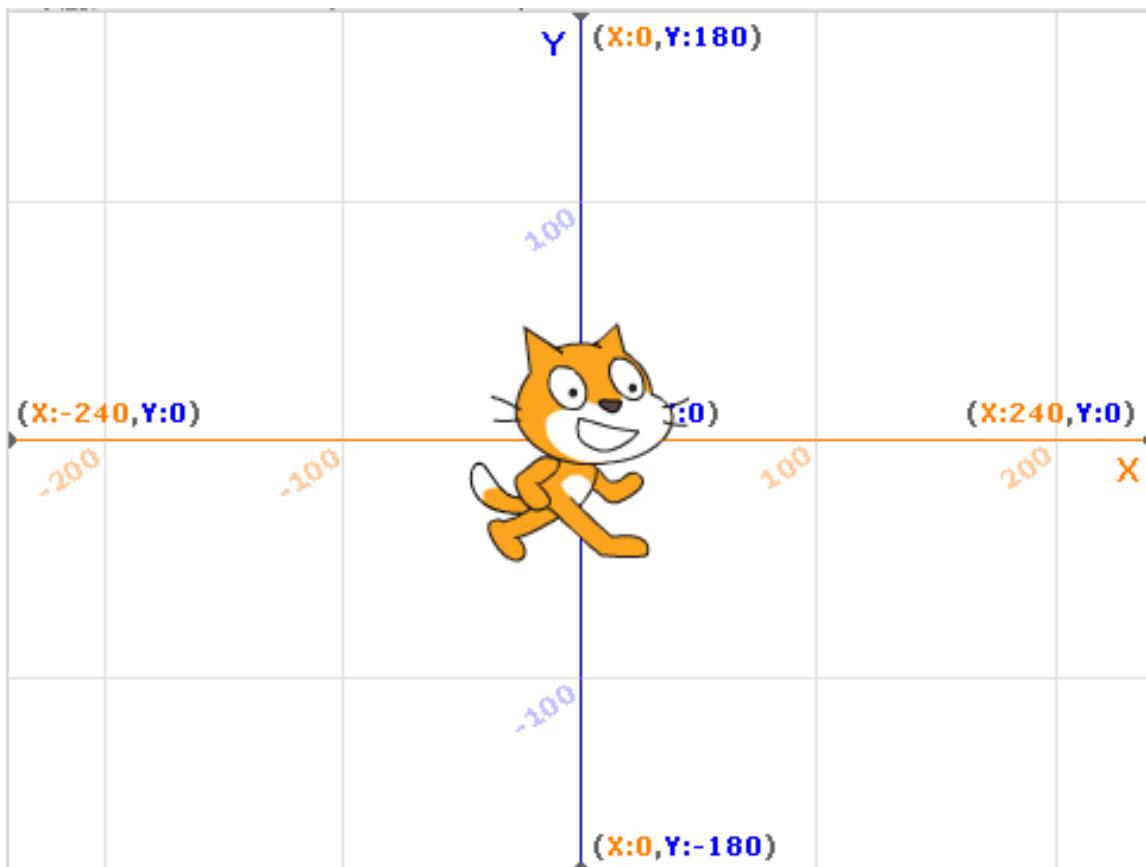
3 角形、4 角形、5 角形などを多角形と言います。

1. 指定した座標にねこを移す

■位置を示す座標とは

- ・ねこの位置は x 座標、y 座標で指定します。
- ・数学（算数）では位置を決めるのに座標を使います。画面の横方向を x 座標、縦方向を y 座標と呼びます。
- ・画面の中心が (0,0) 位置となります。
- ・右方向が x 座標の正の方向、上方向が y 座標の正の方向となります。
- ・Scratch で使う画面の座標は以下のようになっています。
- ・画面の大きさは横（x 座標）が-240～240 で、縦（y 座標）が-180～180 です。

以下の背景は「xy-grid」を選択します。



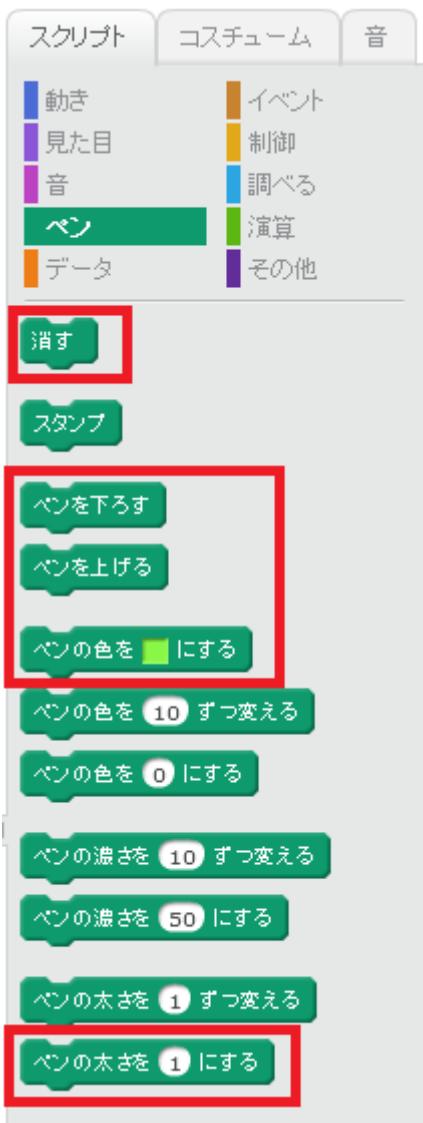
※ 「x 座標、y 座標とは」

x は横軸、y は縦軸を表しています。数字だけの場合は、比例と反比例で学習したように、x 軸の場合は中心より右、y 軸の場合は中心より上を表します。数字にマイナスがついたら、x 軸の場合は中心より左、y 軸の場合は中心より下を表します。

■ペン機能

ステージにペンを使って描画を行うことができます。

- ・「スクリプト」の「ペン」を選択します。緑色のスクリプトです。



- ・ステージに描画されているものをすべて消します。



- ・ペンの上げ、下げを指定します。



- ・ペンの色を指定した色に設定します。色を指定するには  をクリックするとマウスがハンドマークになるので、画面上のスプライトや、スクリプトなどの色をクリックします。するとその色が選ばれます。

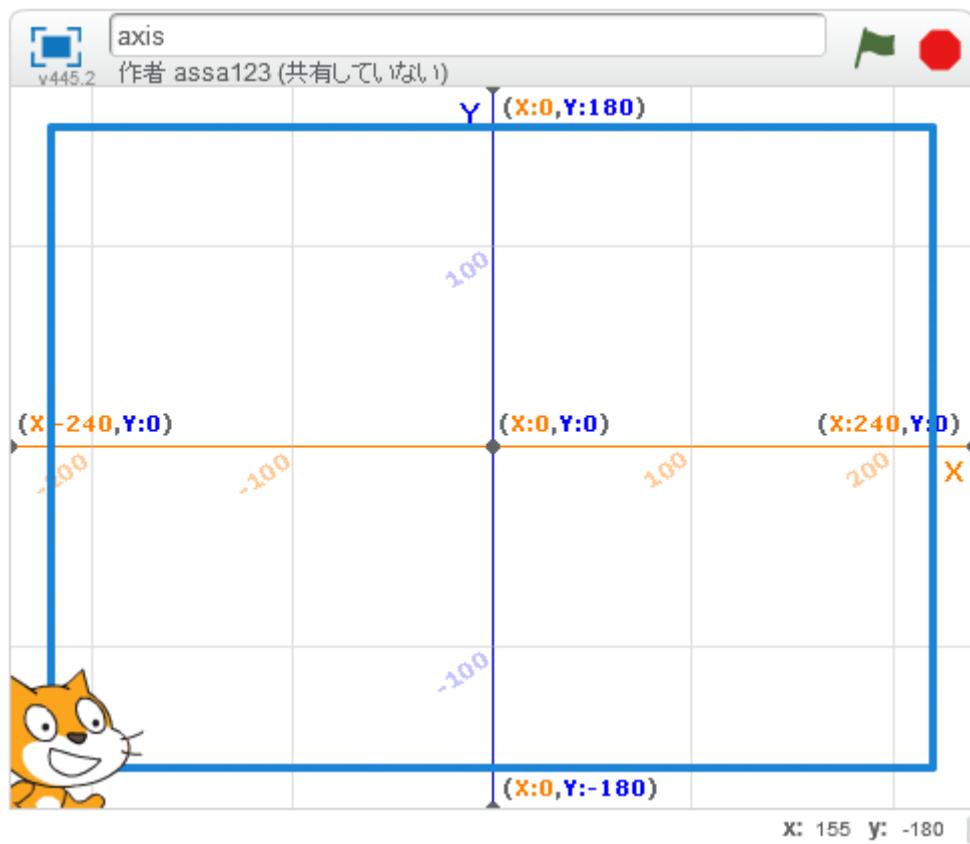
 ペンの色を  にする

- ・ペンの太さを指定した数値に変えます。

 ペンの太さを  にする

■ 4隅を移動するには

画面の左下(-220,-160)をスタートして、右下(220,-160)、右上(220,160)、左上(-220,160)、左下と移動します。

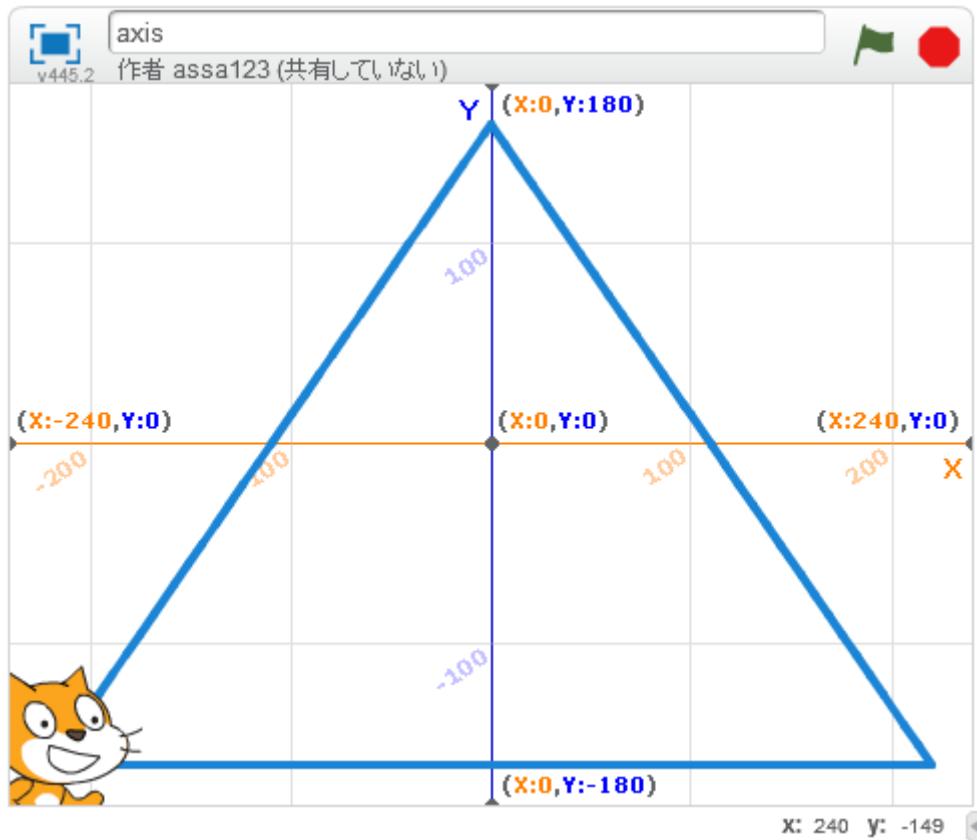


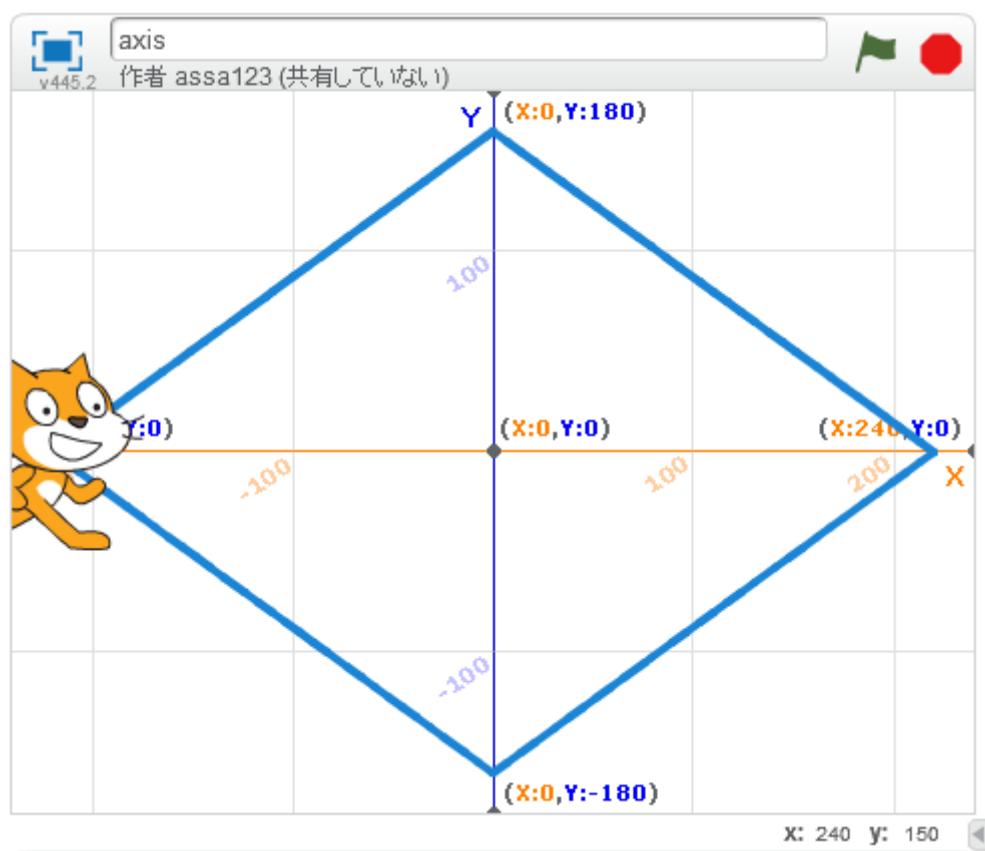
■スクリプト



☆やってみよう

以下のように移動してみよう。





第4回 プログラムのデータを管理する変数

変数とはプログラムで使用するデータを格納しておくためのものです。変数を使ってスライドの位置を管理したり、グラフを描いたりすることができます。

1. 変数とは

変数の作り方、演算式の作り方などを説明します。

2. マウス位置を追いかけてゴーストが移動

変数 x、y を使ってゴーストをマウス位置に移動します。

3. ねこを変数を使って移動

変数 x、y を使ってねこを右に移動します。

4. n 角形を描く

3 角形、4 角形、5 角形など個々の多角形を第3回で描きましたが、変数 n を使って n 角形を描きます。

5. グラフを描いてみよう

$y=ax^2+bx+c$ のような関数のグラフを描きます。

1. 変数とは

■ 変数の役割

- ・変数とはプログラムで使用するデータを格納しておくためのものです。
- ・変数には名前を付けます。これを変数名と呼びます。
- ・一般的なプログラミング言語では変数名には英字と数字だけを使い、「x」とか「x1」のような変数名を使います。
- ・Scratchにはこうした制限はありませんので、「合計」や「a&b」のような変数名も可能です。しかし本書では一般的なプログラミング言語にならい、英数字の変数名を使用します。
- ・英字の大小は区別されますので、「n」と「N」は別の変数になります。

■ 変数の作り方

- ・「データ」ブロックの「変数を作る」で、オープンされたボックスに変数名を入力します。

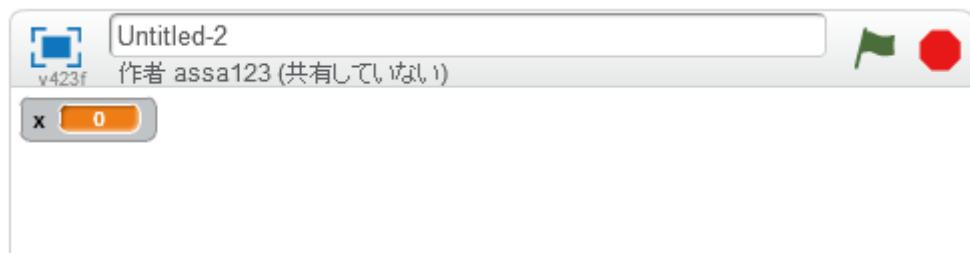


「注」 変数が使用できる範囲を「すべてのスプライト用」か「このスプライトのみ」で選択できます。通常は「すべてのスプライト用」を選択します。

- ・これで、以下のように変数と、変数を操作するスクリプトブロックが作成されます。



- 変数の内容はステージの左上に表示されます。



- ステージへの表示をしない場合は変数の前に付いているチェックを外します。



- 作成した変数を削除するにはマウスの右クリックで削除を選択します。



■変数の操作方法

変数の内容を変更するスクリプトブロックは以下の2つです。

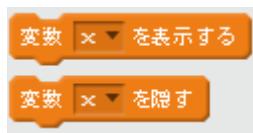
- 変数の内容を指定した内容にします。



- 変数の内容に指定した内容を加えた（減じた）値を新しい内容とします。



- ・プログラム実行時に変数の内容をステージに表示するかしないかを以下のスクリプトブロックで制御することができます。



■演算式の作り方

- ・「スクリプト」の「演算」を選択します。黄緑色のスクリプトです。

演算の基本は加減乗除を行うスクリプトブロックです。



- ・○の中に数値や文字をいれる場合は直接入力します。変数を入れる場合は をドラッグドロップします。

- ・たとえば、 x^*x+x^*x のような複雑な式は の各○の中に を入れて以下のように作ります。



第5回 データをまとめて管理するリスト

変数には1つのデータしか入りませんが、リストには多くのデータを格納することができます。リストのデータを変数を使って取り出すことでプログラミングがし易くなります。

1. リストとは

リストの作り方、リストへのデータの初期設定、リストの各要素の操作などについて説明します。

2. ギガの自己紹介

メッセージをリストに格納しておきギガが自己紹介をします。

3. リストに音のデータを格納

大きな栗の木の下での1小節の音符データをリストに格納しておき演奏します。

4. リストに移動データを格納

Mの文字を、「歩く」と「回す」で描く際の歩数と回転角度をリストに格納しておくことで、柔軟な処理をします。

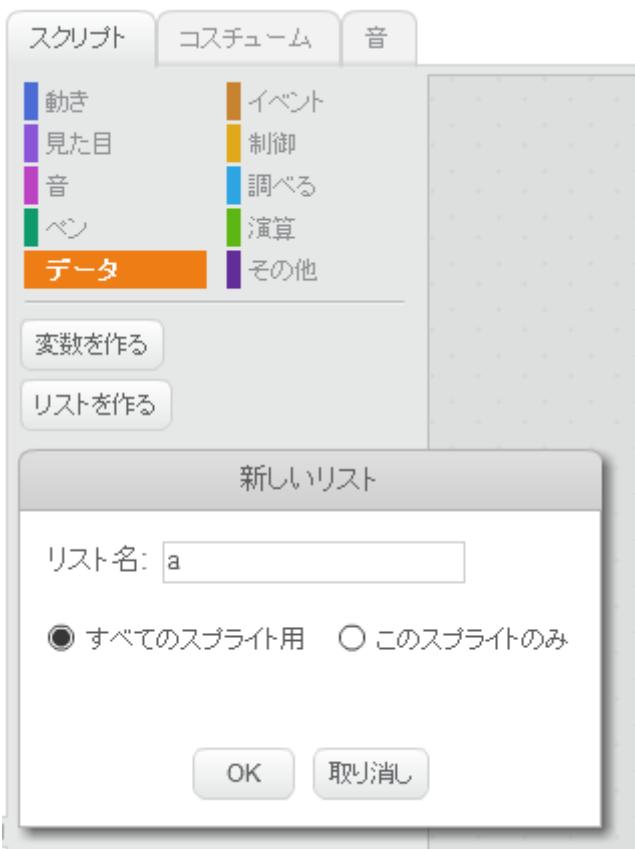
1. リストとは

■リストの役割

- ・変数には1つのデータしか入りませんが、リストには多くのデータを格納することができます。
- ・リストのデータを変数を使って取り出すことでプログラミングがし易くなります。
- ・一般的なプログラミング言語ではリストのことを配列と呼んでいます。

■リストの作り方

- ・「データ」ブロックの「リストを作る」で、オープンされたボックスにリスト名を入力します。

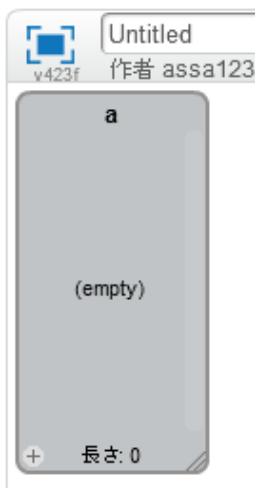


「注」リストが使用できる範囲を「すべてのスプライト用」か「このスプライトのみ」で選択できます。通常は「すべてのスプライト用」を選択します。

- ・これで、以下のようにリストと、リストを操作するスクリプトブロックが作成されます。



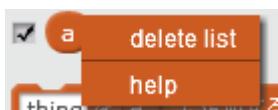
- リストの内容はステージの左上に表示されます。初期状態はデータが入っていない空(empty)です。



- ステージへの表示をしない場合はリストの前に付いているチェックを外します。



- 作成したリストを削除するにはマウスの右クリックで「delete list」を選択します。

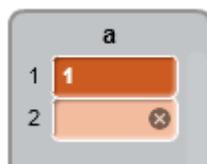


■リストへのデータの初期設定

- リストにデータを初期設定するには  ブロックを使います。データを入力しブロックをクリックするとリストにデータが格納されます。このブロックを使って次々にデータを追加します。



- 2番目以後のデータは  を使わずに、ステージに表示されているリストから入力することもできます。リストの要素にカーソルを移し「Enter」を入力すると次の要素が入力できます。



■リストの各要素の操作

- リストの各要素は何番目の要素という形で表します。要素の番号は1番から始まります。

- リストの1番目の要素は  で取り出せ、i番目の要素は  で取り出せます。要素番号に変数を指定することで柔軟な処理ができます。

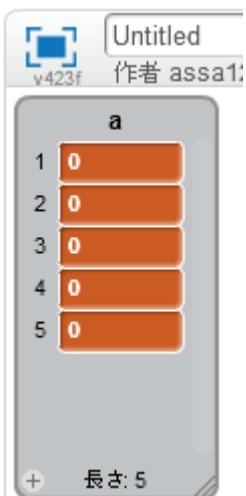
- リストのi番目の要素の内容を変更するには  を使います。

■制御スクリプトブロックを使った配列操作

- 空の配列に対し



を行えば

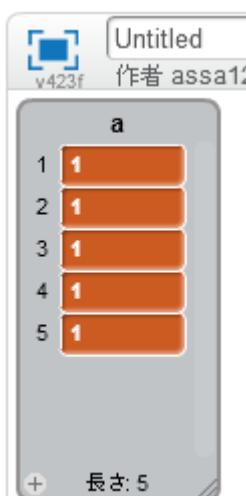


となります。これで長さ 5 のリストができます。

- ・このリストに対し



を行えば



となります。リストの内容はすべて 1 に変更されました。

第6回 プログラムを制御する

プログラムは「繰り返し」、「条件判定」などの制御スクリプトを使ってプログラムの流れを制御します。

1. 制御スクリプトの種類

制御スクリプトとして、繰り返し、条件判定があります。条件を判定するには条件式を使います。

2. ボールを変数を使って跳ね返す

ボールの位置を変数 x 、 y で管理し、移動量を変数 dx 、 dy で管理します。左右の壁に当たるたびに dx の符号を反転し、上下の壁に当たるたびに dy の符号を反転することでボールを跳ね返すことができます

3. ボールをラケットで跳ね返す

ラケットをマウスに追従して動かし、ボールに当てて跳ね返します。

1. 制御スクリプトの種類

プログラムは「繰り返し」、「条件判定」などの制御スクリプトを使ってプログラムの流れを制御します。

■繰り返し

- ・繰り返しを行うスクリプトブロックとして以下があります。「ずっと繰り返す」と「指定回数繰り返す」はすでに説明しました。
- ・「条件になるまで繰り返す」は「7. 同じ処理をひとつにまとめるブロック」で説明します。
- ・ずっと繰り返します



- ・指定した回数だけ繰り返します



- ・指定した条件になるまで繰り返します



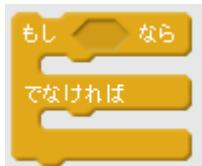
■条件判定

条件判定を行うスクリプトブロックとして以下があります。

- ・もし条件を満たしたらブロック内を実行します



- ・もし条件を満たしたら上のブロックを、そうでなければ下のブロックを実行します



■条件式



に入る条件式として以下があります。

- ・左辺は右辺より小さい



- ・左辺は右辺より大きい



- ・左辺と右辺は等しい



条件が2つ以上ある場合は以下を使います

- ・左辺と右辺の条件がともに満たすとき



- ・左辺または右辺の条件式のどちらかを満たすとき



第7回 同じ処理をひとつにまとめるブロック

同じ処理を何度もする場合に、この処理をブロックとして定義し、必要に応じて呼び出すようにするとプログラムを効率的に書くことができます。

1. ブロックとは

ブロックの作り方と呼び出し方について説明します。

2. 3角形を描くブロック triangle を作ってみよう

3角形を描くブロック triangle と5角形を描くブロック pentagon を作ります。

3. n角形を描くブロック polygon を作ってみよう

ブロックを呼び出す際に、データを渡すことができます。これを引数と言います。n角形を描くブロック polygon に n を引数として渡します。

4. 指定した2点間で直線を描くブロック line を作ってみよう

始点 x1,y1 と終点 x2,y2 を結ぶ直線を描くブロック line を作ります。

5. 数学関数で計算

数学関数を使って円周上の点の位置を計算し、ブロック line を使ってその円周上の点を結ぶことで様々な図形を描きます。

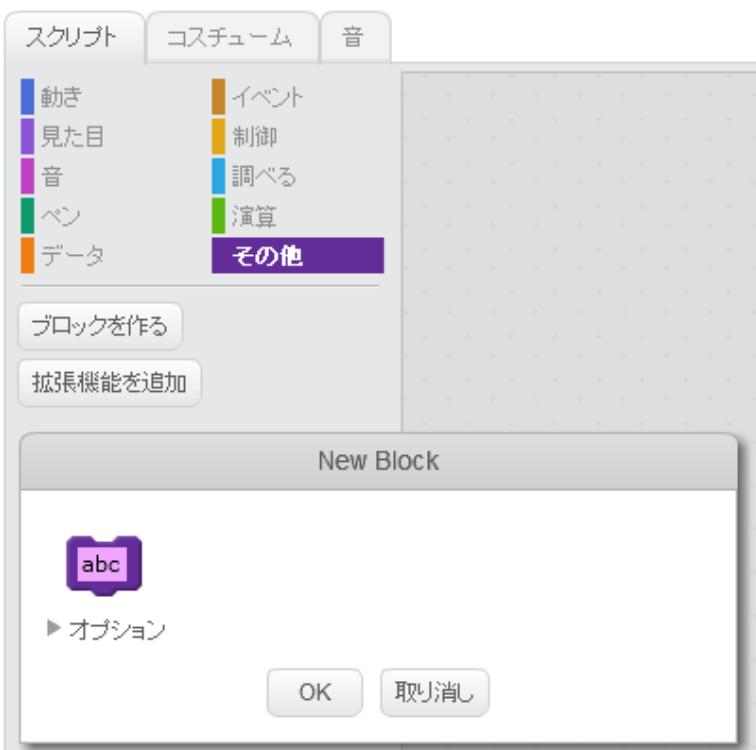
1. ブロックとは

■ ブロックの役割

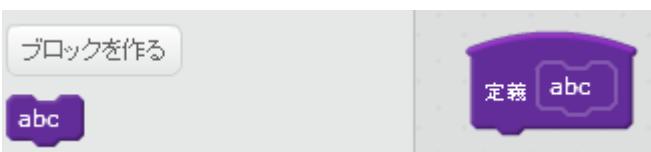
- 同じ処理を何度もする場合に、この処理をブロックとして定義し、必要に応じて呼び出すようにするとプログラムを効率的に書くことができます。
- 自分で定義したブロックは紫色です。
- 新しく定義したブロックには呼び出す方からデータを与えることができます。これを引数と言います。
- 新しく定義したブロックは一般的のプログラミング言語では関数とかプロシージャとかメソッドなどと呼ばれているものです。

■ ブロックの作り方

- 「その他」ブロックの「ブロックを作る」をクリックします。ブロックの名前を入力します。ここでは「abc」としました。



- 以下のように abc という名前の新しいブロックが作られます。



- この abc ブロックにやらせたい内容を並べます。この例ではスプライトを 10 歩いて 15 度回転させます。



-  をクリックすれば abc ブロックが実行されます。

第8回 データ構造とアルゴリズム

プログラムを使って問題を解くための論理または手順をアルゴリズム (algorithms: 算法) といいます。プログラム処理では多量のデータを扱うことが多く、この場合、取り扱うデータをどのようなデータ構造 (data structure) にするかで、問題解決のアルゴリズムが異なってきます。データ構造とアルゴリズムは密接な関係にあり、良いデータ構造を選ぶことが良いプログラムを作ることにつながります。

1. 決定木

データ構造としては、決定木を用いた食文化判定や性格判定プログラムを紹介します。

2. ハノイの塔

プログラムの世界に特有なアルゴリズムとして再帰という考え方があります。再帰の代表的な例としてハノイの塔を紹介します。

3. リカーシブ・グラフィックス

再帰を用いたリカーシブ・グラフィックス（再帰図形）を紹介します。

1. 決定木

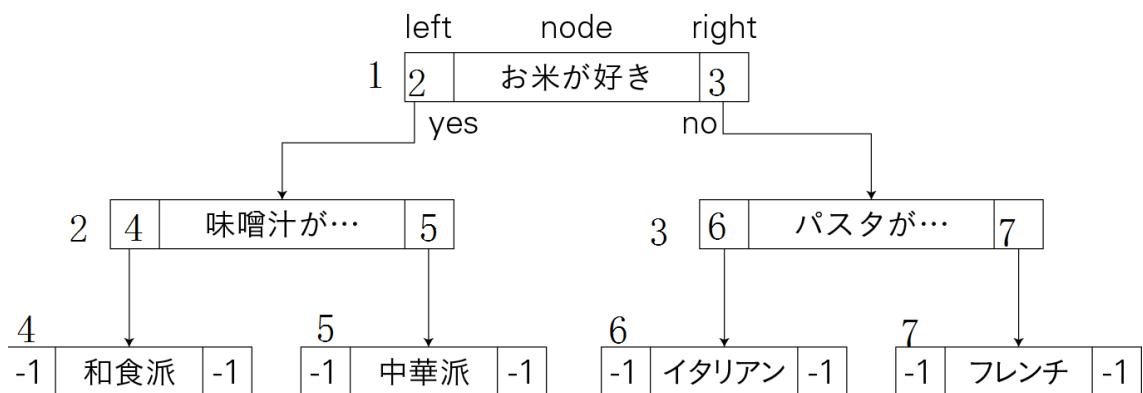
■決定木とは

木はいくつかのノードとそれを結ぶ枝から構成されます。ノードはデータに対応し、枝はデータとデータを結ぶ親子関係に対応します。あるノードから下方に分岐する枝の先にあるノードを子と呼び、分岐元のノードを親と呼びます。

木の一番始めのノードをとくに根（ルート）と呼び、子を持たないノードを葉と呼びます。木の中のあるノードを相対的な根と考え、そこから枝分かれしている枝とノードの集合を部分木と呼びます。

木のうちで、各ノードから出る枝が 2 本以下のものを特に 2 分木と呼び、データ構造の木の中心はこの 2 分木です。2 分木の中で、ノードの内容の意味の持たせ方で、決定木、2 分探索木などに分かれます。質問項目に yes、no で枝分かれするような木を決定木と呼びます。

食文化判定の決定木



left が左のノードへのポインタ、right が右のノードへのポインタ、node が質問内容になります。ポインタは連結する配列要素の添字となります。子を持たない葉ノードのポインタ部には「-1」を置くことにします。

添字	左へのポインタ left	質問内容 node	右へのポインタ right
1	2	お米が好き	3
2	4	味噌汁が好き	5
3	6	パスタが好き	7
4	-1	和食派	-1
5	-1	中華派	-1
6	-1	イタリアン	-1
7	-1	フレンチ	-1

■決定木をたどる

Giga が node リストに格納されている質問を話します。

- ・質問に対し Yes なら「y」キー、No なら「n」キーを押します。
- ・「y」キーが押されたら、変数 p を left の p 番目の内容にします。決定木で左へ進むことになります。
- ・「n」キーが押されたら、変数 p を right の p 番目の内容にします。決定木で右へ進むことになります。



■変数として p を作ります



■リストとして node, left, right を作ります

node	
1	お米が好きですか
2	味噌汁が好きですか
3	パスタが好きですか
4	あなたは和食派ですね
5	あなたは中華派ですね
6	あなたはイタリアンですね
7	あなたはフレンチですね

left	
1	2
2	4
3	6
4	-1
5	-1
6	-1
7	-1

right	
1	3
2	5
3	7
4	-1
5	-1
6	-1
7	-1

■スクリプト



☆やってみよう

- ・質問内容を自分で考えてみよう

著者略歴

河西 朝雄（かさいあさお）

山梨大学工学部電子工学科卒（1974年）。長野県岡谷工業高等学校情報技術科教諭、長野県松本工業高等学校電子工業科教諭を経て、現在は「カサイ. ソフトウェアラボ」代表。

「主な著書」

「入門ソフトウェアシリーズ C 言語」、「同シリーズ Java 言語」、「同シリーズ C++」、「入門新世代言語シリーズ VisualBasic4.0」、「同シリーズ Delphi2.0」、「やさしいホームページの作り方シリーズ HTML」、「同シリーズ JavaScript」、「同シリーズ HTML 機能引きテクニック編」、「同シリーズホームページのすべてが分かる事典」、「同シリーズ i モード対応 HTML と CGI」、「同シリーズ i モード対応 Java で作る i アプリ」、「同シリーズ VRML2.0」、「チュートリアル式言語入門 VisualBasic.NET」、「はじめての VisualC#. NET」、「C 言語用語辞典」ほか（以上ナツメ社）

「構造化 BASIC」、「Microsoft Language シリーズ Microsoft VISUAL C++ 初級プログラミング入門上、下」、「同シリーズ VisualBasic 初級プログラミング入門上、下」、「C 言語によるはじめてのアルゴリズム入門」、「Java によるはじめてのアルゴリズム入門」、「VisualBasic によるはじめてのアルゴリズム入門」、「VisualBasic6.0 入門編、中級テクニック編、上級編」、「Internet Language 改訂新版シリーズ ホームページの制作」、「同シリーズ JavaScript 入門」、「同シリーズ Java 入門」、「New Language シリーズ標準 VisualC++ プログラミングブック」、「同シリーズ標準 Java プログラミングブック」、「VB.NET 基礎学習 Bible」、「原理がわかるプログラムの法則」、「プログラムの最初の壁」、「河西メソッド：C 言語プログラム学習の方程式」、「基礎から学べる VisualBasic2005 標準コースウェア」、「基礎から学べる JavaScript 標準コースウェア」、「基礎から学べる C 言語標準コースウェア」、「基礎から学べる PHP 標準コースウェア」、「なぞりがき C 言語学習ドリル」、「C 言語標準ライブラリ関数ポケットリファレンス[ANSI C,ISO C99 対応]」、「C 言語 標準文法ポケットリファレンス[ANSI C,ISOC99 対応]」、「[標準] C 言語重要用語解説 ANSI C / ISO C99 対応」ほか（以上技術評論社）

「電子書籍：カサイ. ソフトウェアラボ」

「Android プログラミング Bible 初級 基礎編」、「Android プログラミング Bible 中級 Android 的プログラミング法」、「Android プログラミング Bible 上級 各種処理」、「Android プログラミング完全入門」、「iPhone&iPad プログラミング Bible[上]」、「iPhone&iPad プログラミング Bible[下]」、「JavaScript によるはじめてのアルゴリズム入門」、「Web アプリ入門（HTML5+JavaScript）」、「HTML5 を使った JavaScript 完全入門」、「Scratch プログラミング入門」



小・中学生のための
Scratch プログラミング入門

2016年7月1日 初版 第1刷

著者＝河西 朝雄

発行者＝河西 朝雄

発行所＝カサイ. ソフトウェアラボ

長野県茅野市ちの 813 TEL.0266-72-4778

表紙デザイン＝河西 朝樹

本書の一部または全部を著作権法の定める範囲を超えて、無断で複写、複製、転載、あるいはファイルに落とすことを禁じます。

本書に記載された内容は、情報の提供のみを目的としています。したがって、本書を用いた運用は、必ずお客様自身の責任と判断によって行ってください。これらの情報の運用の結果について、発行者および著者はいかなる責任も負いません。

定価＝500円＋税

©2016 河西 朝雄