

# 「スクラッチ」で プログラミング！

コンピュータに仕事をさせるには、その手順をあらかじめコンピュータに記憶させる必要があります。その手順を記述したものを「プログラム」といい、「プログラム言語」という言語を使って記述します。

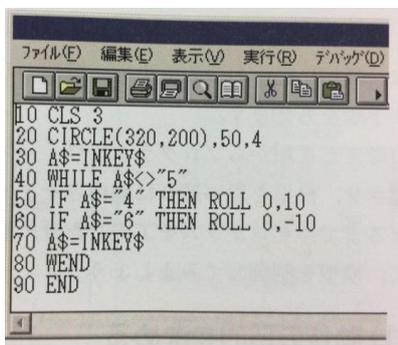
## ●プログラム言語の例

- BASIC言語 …… 教育用に開発された言語、初心者用
- C言語 …… OSやアプリケーションなどの開発に向いている言語
- FORTRAN言語 …… 科学技術計算に向いている言語
- 簡易言語 …… アイコンをつないで簡単なプログラミングができる

今回は「簡易言語」のひとつである「スクラッチ」を使ってみましょう。

「スクラッチ」とは、「BASIC言語」「C言語」などのようにキーボードからプログラムを打ち込むのではなく、画面のブロックをマウス操作で組み合わせてプログラムする「ビジュアルプログラミング言語」です。

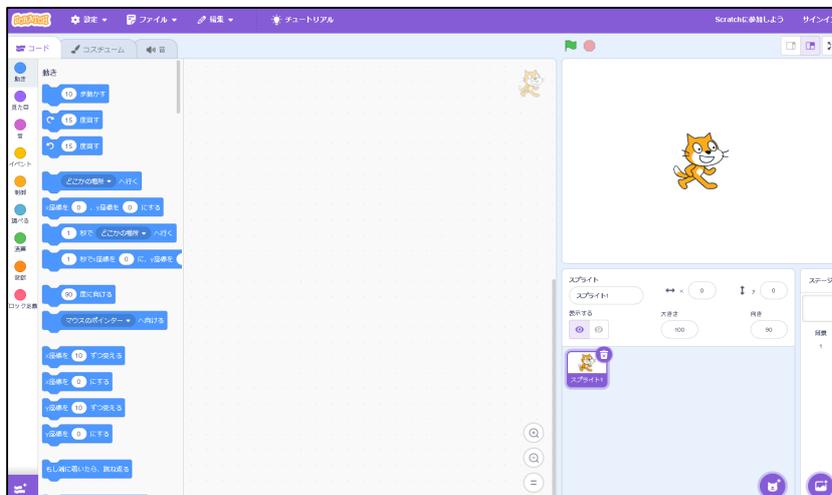
## ■BASIC言語



```
10 CLS 3
20 CIRCLE(320,200),50,4
30 A$=INKEY$
40 WHILE A$<>"5"
50 IF A$="4" THEN ROLL 0,10
60 IF A$="6" THEN ROLL 0,-10
70 A$=INKEY$
80 WEND
90 END
```

↑ 命令は文字ばかり

## ■簡易言語「スクラッチ」



↑ 左側にあるブロックをつなぎあわせてプログラムを作る。視覚的に命令の流れが分かりやすい！

**「スクラッチ」を使って、簡単なプログラムを作ろう！  
そして、プログラミングの基本を学ぼう！  
試行錯誤しながら「課題を解決する力」を身に付けよう！**

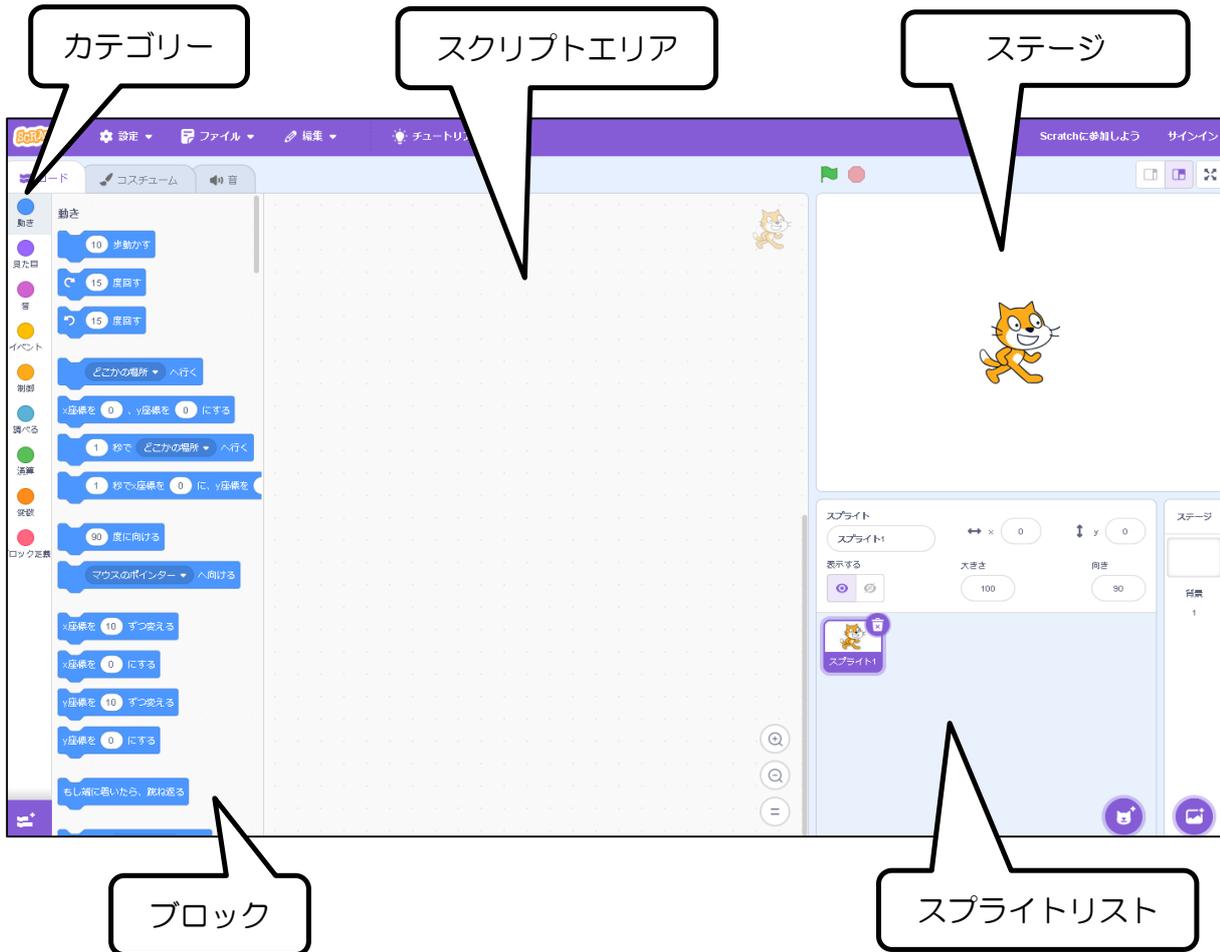
※ 「スクラッチ」は、以下のURLからジャンプできます！

URL → <https://scratch.mit.edu/projects/editor/>

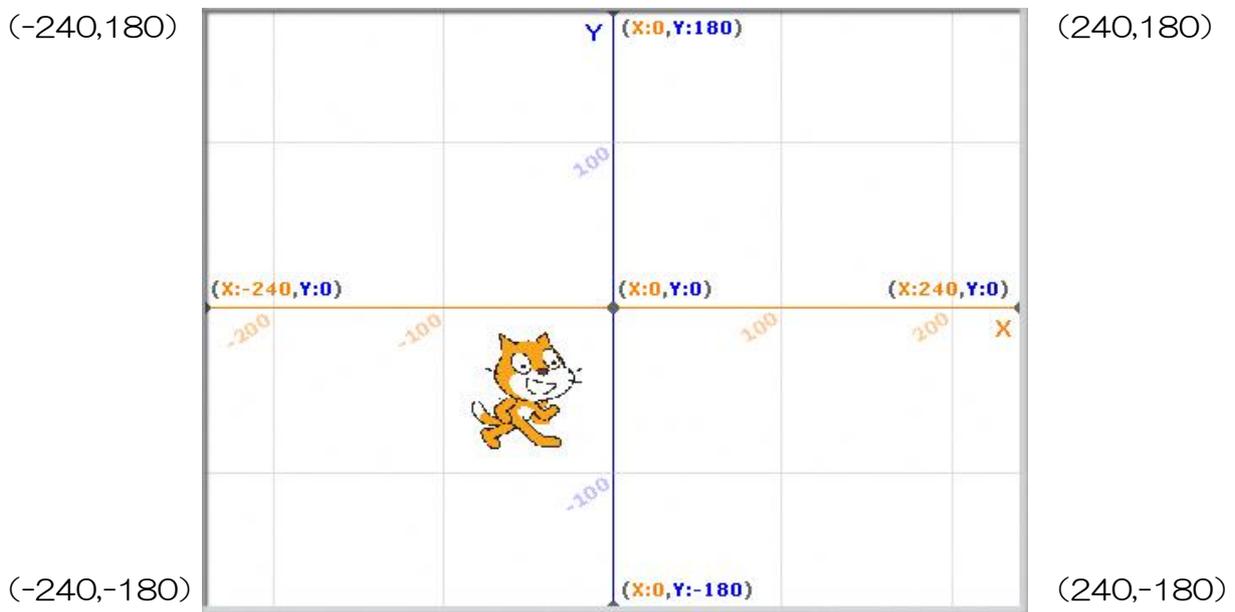
(参考) Scratchでつくる！たのしむ！プログラミング道場【株式会社ソーテック社】

## ●画面を確認しよう！

- ・「スプライト」 …… キャラクターや背景のこと
- ・「スクリプト」 …… プログラムのこと（台本という意味があります）



## ●座標を確認しよう！（縦軸がY軸、横軸がX軸）



# STEP.1 スプライトを動かしてみよう!

## ●ネコを動かしてみよう!

- ①「カテゴリー」の「イベント」から「(緑旗)が押されたとき」を、スクリプトエリアにドラッグ&ドロップする。
- ②「カテゴリー」の「動き」から「10歩動かす」を、①で置いた「(緑旗)が押されたとき」の下に近づけるようにドラッグする。
- ③2つのブロックの間に、灰色の影ができればドロップする。2つのブロックが1つにつながり、スクリプト(台本)ができあがる。
- ④ステージの右上の「緑色の旗」をクリックするとネコが10歩だけ動く。



■「緑の旗が押されたときに、10歩動かす」というスクリプト(台本)を作ったので、ネコ(役者)が台本のとおりステージ上で動いたのです。

## ●もっとネコを動かしてみよう!

- ①「カテゴリー」の「制御」から「ずっと」を選んで、「10歩動かす」を囲むように入れる。
- ②動かしてみよう。

■反復命令(ずっと)によって、ネコは壁にぶつかるまで動きました。

- ③「カテゴリー」の「動き」から「もし端に着いたら、跳ね返る」を入れる。
- ④動かしてみよう。
- ⑤ネコがさかさまになるので、「カテゴリー」の「動き」から「回転方法を左右のみにする」を入れる。
- ⑥動かしてみよう。



■「緑の旗」の横の「赤い丸」をクリックするとスクリプトを止められます。

# STEP.2 複数のスプライトを動かしてみよう！

## ●スプライトを増やそう！

- ①「スプライトリスト」の「スプライトを選ぶ」ボタンをクリックし、ファイルの中から「バナナ」を選ぶ。
- ②「スプライトリスト」に新しいスプライトの「バナナ」が追加される。



## ●バナナも動かしてみよう！

- ①それぞれの「スプライト」ごとに「スクリプト」を入力するので、バナナの「スクリプト」を右の図を参考に入力する。
- ②「90度に向ける」の「90」を「45」に変更し斜めに進むように直す。
- ③動かしてみよう。



## ●スプライトの大きさを調節しよう！

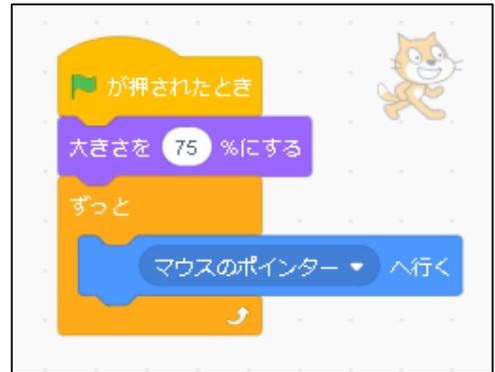
- ①「カテゴリー」の「見た目」から「大きさを100%にする」を入れ、バランスを考えて「100%」を「50%」に直す。
- ②ネコとバナナの両方のスプライトの大きさをそれぞれ変更する。
- ③動かしてみよう。



## STEP.3 ネコがバナナを捕まえる！

### ●ネコをマウスで自由に動かそう！

- ①ネコの「スクリプト」から「10歩動かす」と「もし端に着いたら…」を削除する。
- ②「カテゴリー」の「動き」から「どこかの場所へ行く」を入れ、「マウスのポインターへ行く」に変更する。
- ③動かしてみよう。



■ネコがマウスポインターを追いかけて動くようになります。すなわち、マウスでネコを自由に動かせるようになります。

### ●バナナを捕まえたら「バナナだ！」とセリフを言わせよう！

- ①「カテゴリー」の「制御」から「もし（ ）なら」を入れる。
- ②「カテゴリー」の「調べる」から「マウスのポインターに触れた」を（ ）に入れ、「bananasに触れた」に変更する。
- ③「カテゴリー」の「見た目」から「こんにちは！と2秒言う」を入れ、「バナナだ！を0.5秒言う」に変更する。
- ④動かしてみよう。



■「もし（ ）なら」のスク립トを使うことで、条件を満たせば別の動きをさせるプログラムを作ることができます。ネコがバナナを追いかけて、バナナに触れたら「バナナだ！」と表示するプログラムができます。

## STEP.4 簡単なゲームをつくろう！

■今までのプログラムをさらに改良して、「ネコがバナナやボールをつかまえるゲーム（タイム・得点つき）」を作ってみよう。



### ●「ボール」のスプライトを増やそう！

- ①「スプライトリスト」の「スプライトを選ぶ」ボタンをクリックし、ファイルの中から「ボール」を選ぶ。
- ②「スプライトリスト」に新しいスプライトの「ボール」が追加される。
- ③「ボール」のスク립トを右の図を参考に入力する。（角度や歩数を工夫してみよう）
- ④動かしてみよう。



### ●「得点」を表示しよう！

- ①「カテゴリー」の「変数」から「変数を作る」を選び、新しい変数名を「得点」とする。
- ②「カテゴリー」の「変数」に新しく追加されたブロックから、右の図を参考に「得点を0にする」を入力する。
- ③「カテゴリー」の「変数」に新しく追加されたブロックから、右の図を参考に「得点を1ずつ変える」を入力する。
- ④「バナナ」と「ボール」の得点を変える。右の図を参考に「ボール」の得点を2点にする。
- ⑤動かしてみよう。

■これで「ゲームが始まったら得点を0にして、ネコがバナナに触ったら1点が入り、ボールに触ったら2点が入る」プログラムができます。



## ● 「タイム」を表示しよう！

- ①「カテゴリー」の「変数」から「変数を作る」を選び、新しい変数名を「タイム」とする。
- ②右下の図を参考に、ネコを動かすスクリプトの横にタイムのスクリプトを入力する。
- ③「カテゴリー」の「変数」に新しく追加されたブロックから、「タイムを0にする」「タイムを1ずつ変える」のスクリプトを使う。
- ④「カテゴリー」の「演算」から「( ) = ( )」を選び、「(タイム) = (0)」と入力する。
- ⑤動かしてみよう。



■ ゲームが始まったら、タイムを15秒にセットします。「1秒待ってタイムを1秒減らす」を15回繰り返し、タイムが0になったらすべてを止める」プログラムができあがります。

■ 右のようにネコのスクリプトの中に「緑旗が押されたとき」から始まる命令が、2つ同時に実行されるようになります。



## ● 「背景」や「音」を設定しよう！

■ 背景を変えるには、「スプライトリスト」から「背景を選ぶ」ボタンをクリックし、ファイルの中から好きな「背景」を選ぶ。背景は自分でペイントで描くこともできます。



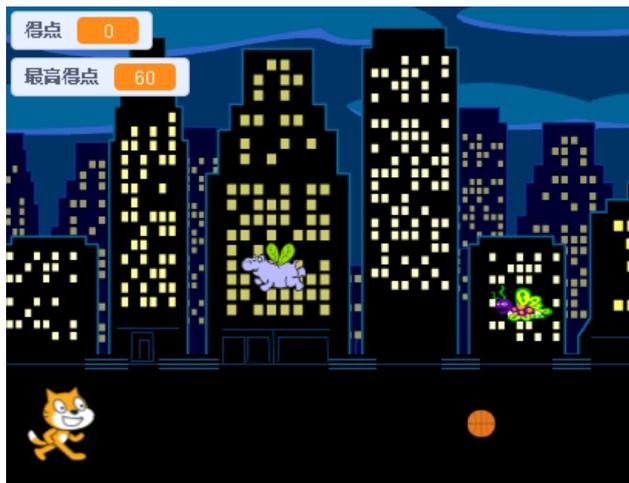
■ 音を入れるには、「カテゴリー」の「音」から「ニャーの音を鳴らす」を入れます。バナナに触ると「ニャー」と鳴くようになります。音は、録音することもできます。



# STEP.5 フログラミングの流れを考えよう!

■今までの学習をもとに、右のようなシューティングゲームを作ってみよう!

※このゲームは、右からどんどん出てくる「オバケ」に、ボールを当てて倒すゲームです。



## ●プログラムを組んでいく流れを考えよう!

下のA~Fの作業をどんな順番で行えばいいでしょう?

後からでもプログラムは修正できますが、効率的な順番を考えましょう。

例：オバケ1のプログラムを作って、オバケ2にプログラムをコピーする。

( ) → ( ) → ( ) → ( ) → ( ) → ( )

<b>A</b>   スプライトを作る
• ネコ、ボール、オバケ • 背景 • イラストの向き • スプライトの名前

<b>B</b>   オバケが出現する
• 右から現れ、左へ飛ぶ • ボールが当たると消える • 左端まで飛ぶと消える • ネコに当たると終了

<b>C</b>   ネコを上下移動
• 左端にネコを表示 • 「↑」キーで上に移動 • 「↓」キーで下に移動

<b>D</b>   音楽を鳴らす
• BGM • オバケを倒した音 • ネコがやられた音 • 音を止めるタイミング

<b>E</b>   ボールを投げる
• スペースキーで投げる • ネコから投げる • 右端まで飛ぶと消える

<b>F</b>   得点を入れる
• 得点を表示する • オバケの得点を決める • 最高得点を表示する

# STEP.6 シューティングゲームを作ろう！

## ●「スプライト」を準備しよう！

- ①右の図を参考に、4種類の「スプライト」を準備する。
- ②「スプライト」の名前を、分かりやすい名前に変更することができる。
- ③向きを変える必要がある場合は「コスチューム」で、図の左右を反転させる。



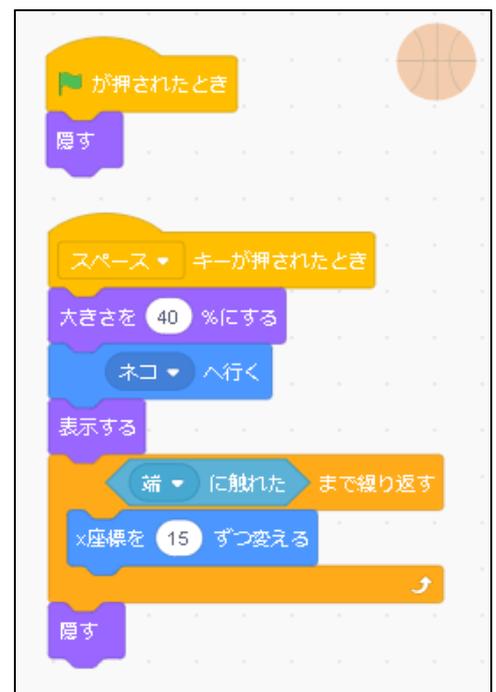
## ●「ネコ」を上下に移動させよう！

- ①「ネコ」を表示する大きさを決める。
- ②「ネコ」を座標「X=-200、Y=0」に表示する。
- ③「上向き矢印キー」が押されたら、Y座標の値を増やし、「ネコ」を上を移動させる。
- ④「下向き矢印キー」が押されたら、Y座標の値を減らし、「ネコ」を下を移動させる。
- ⑤動かしてみよう。



## ●「ネコ」から「ボール」を発射させよう！

- ①「スペースキー」が押されたら、「ネコ」の手元に「ボール」が表示される。
- ②「ボール」を表示する大きさを決める。
- ③「ボール」が画面の端に触れるまで、X座標の値を増やし続け、「ボール」を飛ばす。
- ④画面の端に到達すると「ボール」を「隠す」。
- ⑤動かしてみよう。



■ここまでの命令で、「ネコ」を「↑↓キー」で移動させ、「スペースキー」で「ボール」を発射するプログラムになります。

## ● 「背景・音・変数」を準備しよう！

- ①使用する「背景」「音」「変数」を準備する。  
「音」は使用する「スプライト」に登録する。
- ②「ステージ」に、スプライトの動きとは関係ない「得点」「最高得点」「BGM」のプログラムを書き込む。
- ③ゲームが始まると「得点」の表示を0にして、「Dance Energetic」という音を繰り返し鳴らしてBGMにする。
- ④「r」キーを押すと「最高得点」をリセットし、0点に戻す。



■スプライトの動きと関係ない命令は、「ステージ」に書き込むことで「スクリプト」がはっきりします。

## ● 「オバケ」を飛ばそう！

- ①「オバケ」を「乱数」を使って、ランダムに出現するように設定する。  
Y座標を「-150~150」に設定。
- ②「オバケ」が端に触れるまで、X座標を「-5」ずつ減らす。「-5」は「オバケ」のスピード。
- ③もし「オバケ」が「ボール」に触れたら、「得点」を増やして「Bite」の音を鳴らして「オバケ」を隠す。
- ④もし「オバケ」が「ネコ」に触れたら、「ニャー」の音を鳴らしてゲームを終了する。
- ⑤もしゲーム終了時に「最高得点」だった場合は「最高得点」の欄に今の「得点」を表示する。

■この「オバケ」のスクリプトはかなり複雑ですが「プログラムの流れや意味」が分かれば、「スクラッチ」のほとんどの機能を使いこなせるはず。がんばってプログラムを読み解いてみましょう。



## ●「オバケ2」を飛ばそう！

- ①「オバケ2」の「スクリプト」は基本的に「オバケ1」と同じ設定なので、「オバケ1」の「スクリプト」をドラッグして、スプライトリストの「オバケ2」のアイコンの上でドロップすると、「オバケ1」の「スクリプト」がコピーされる。「オバケ2」にも必要な「音」を登録しないと音が鳴らない。
- ②同じ命令だと、2種類の「オバケ」が同じタイミングで出現するので、右図を参考にして「1秒待つ」の命令を入れ、出現するタイミングをずらす。
- ③2種類の「オバケ」の「進む速さ」と「得点」を変更する。前のページを参照して、自分なりに工夫してみよう。
- ④動かしてみよう。



## ●ゲームを改良してみよう！

さらにこのゲームを面白くするにはどこをどう改良すればよいかを考えてみよう！

ここをこうしたい！（改良点）	プログラムの変更点