

# 体験授業：Scratchによるプログラミング

ネットワークセキュリティ研究室 村上 恭通



つなぐ知 かなえる技

大阪電気通信大学

Osaka Electro-Communication University

ITとIoTを裏から支える

通信工学科



# Scratch体験

① <http://scratch.mit.edu> にアクセス



## 注目のプロジェクト



Walk me home.проект  
Dream\_sunset



\*Secret of the Rings  
hydreana\_voice



Future Space Hover  
Dad76



Broccoli for Dessert  
Lizzy\_06



Simply Snail  
\_Kat-taK\_

☆Scratchはブラウザ上で(オンラインで)動作する

☆パソコンやAndroidで動くScratchアプリもある(<https://scratch.mit.edu/download/>)

# プログラミングの流れ

① プロジェクト(作品)名を入力

③ コードタブを選択

④ 部品の種類を選択

⑤ 部品をスクリプトエリアにドラッグしてプログラミング

⑥ ステージ上で動作

② スプライトを選択

The image shows a screenshot of the Scratch programming environment with several annotations in green boxes and red outlines. The browser address bar shows 'scratch.mit.edu'. The top navigation bar includes 'ファイル', '編集', 'チュートリアル', 'Untitled', '共有する', and 'プロジェクトページを見る'. The left sidebar has tabs for 'コード', 'コスチューム', and '音'. The main workspace is divided into a 'ブロックパレット' (left) and a 'スクリプトエリア' (center). The 'ブロックパレット' contains various code blocks categorized by '動き', '見た目', '音', 'イベント', '制御', '調べる', '演算', '変数', and 'ブロック定義'. The 'スクリプトエリア' contains a 'スクリプトエリア' label and a 'スクロール' button. The 'ステージ' (stage) area on the right shows a cat sprite. The 'スプライトリスト' (sprite list) at the bottom right shows a 'スプライト1' (Sprite1) with 'x' and 'y' coordinates set to 0, and '大きさ' (size) set to 100 and '向き' (direction) set to 90. A 'バックパック' (backpack) is visible at the bottom left. A green arrow points from the 'スクリプトエリア' to the 'ステージ'.

バックパック

バックパック

☆バックパックには部品を置いておける

# Scratch入門:マウスを追いかける猫(無限ループ)



図1



図2



図3

- ① Scratchを起動
- ② スプライトの「スプライト1」が選択されている
- ③ スクリプトの「イベント」をクリック
- ④  をスクリプトエリアにドラッグ
- ⑤ スクリプトの「制御」をクリック
- ⑥  をドラッグして結合(図1参照)
- ⑦ スクリプトの「動き」をクリック
- ⑧  を「ずっと」の中にドラッグ(図1)
- ⑨ 図1になったら実行! マウスの向きに回るはず
- ⑩  をドラッグして結合(図2)
- ⑪ 図2になったら実行! マウスに向かってくるはず
- ⑫ スクリプトの「見た目」をクリック
- ⑬  をドラッグして結合(図3)
- ⑭ 図3になったら実行! マウスに向かって歩くはず

# Scratch入門:Gigaから逃げろ(オブジェクト指向・条件)



図4

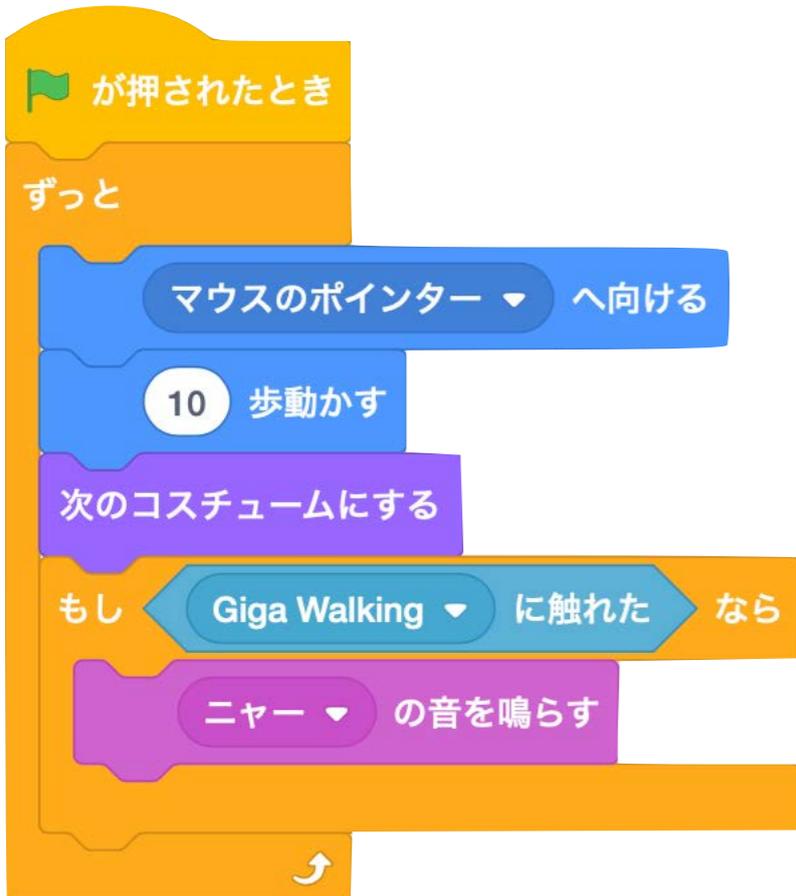
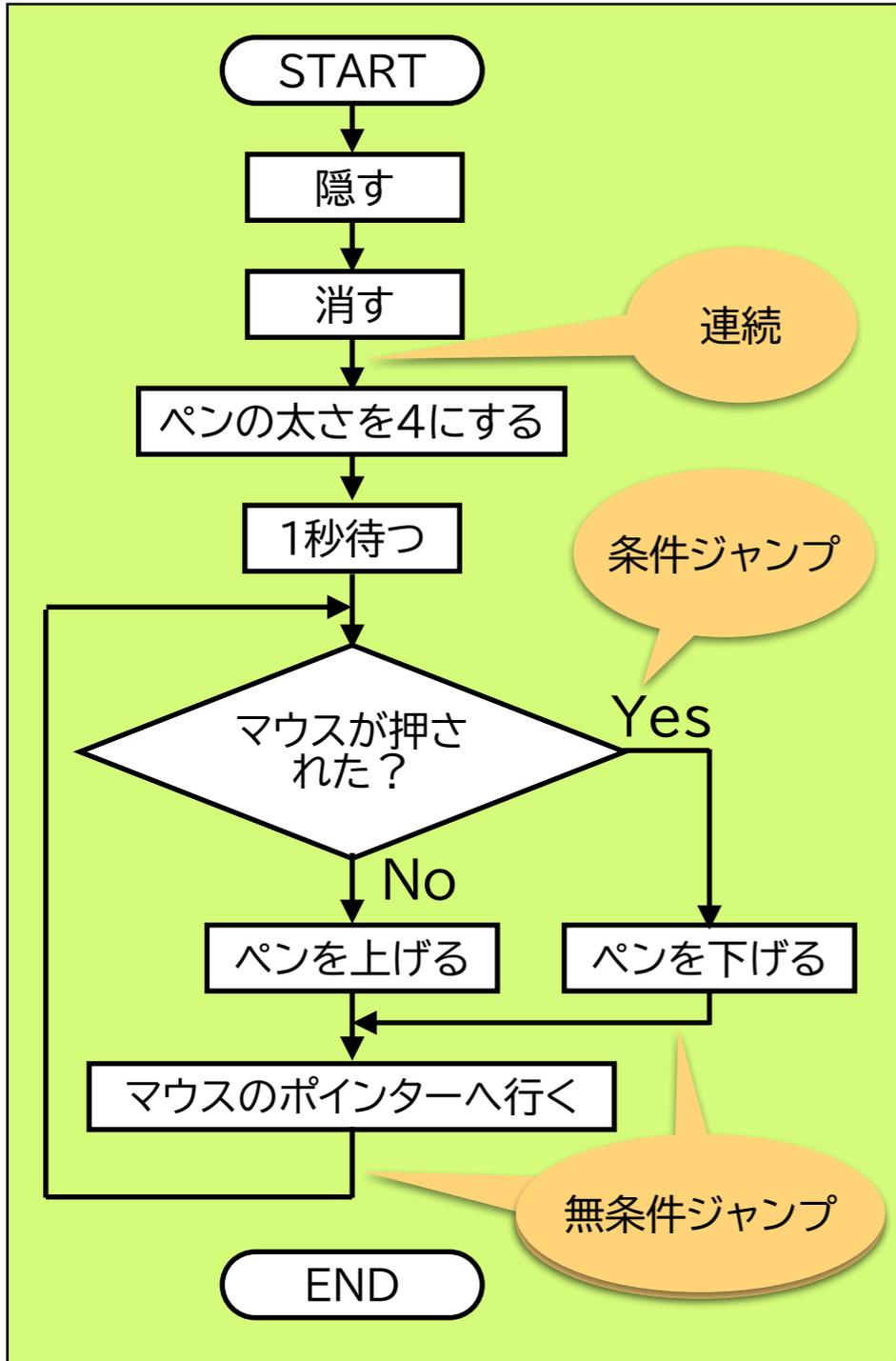


図5

- ① 「新しいスプライト」をクリック
- ② 「Giga Walking」を追加
- ③ 「スプライト1(猫)」をクリック
- ④ 図3のスク립トを「Giga Walking」にドラッグ
- ⑤ 実行！猫とGigaがマウスに向かってくるはず
- ⑥ 「Giga Walking」をクリック
- ⑦ 歩数を3にして実行！Gigaが猫より遅くなるはず
- ⑧ ▼をクリックして「スプライト1」にする(図4)
- ⑨ 実行！Gigaが猫を追いかけるはず. 逃げろ！
- ⑩ 「スプライト1(猫)」をクリック
- ⑪ 「制御」の  を  の下に結合
- ⑫ 「音」の  を中にはめる
- ⑬ 「調べる」の  を条件部分にはめる
- ⑭ ▼をクリックして「Giga Walking」にする(図5)
- ⑮ 図5になったら完成！Gigaに捕まると鳴くはず

# 構造化プログラミング入門

## フローチャート



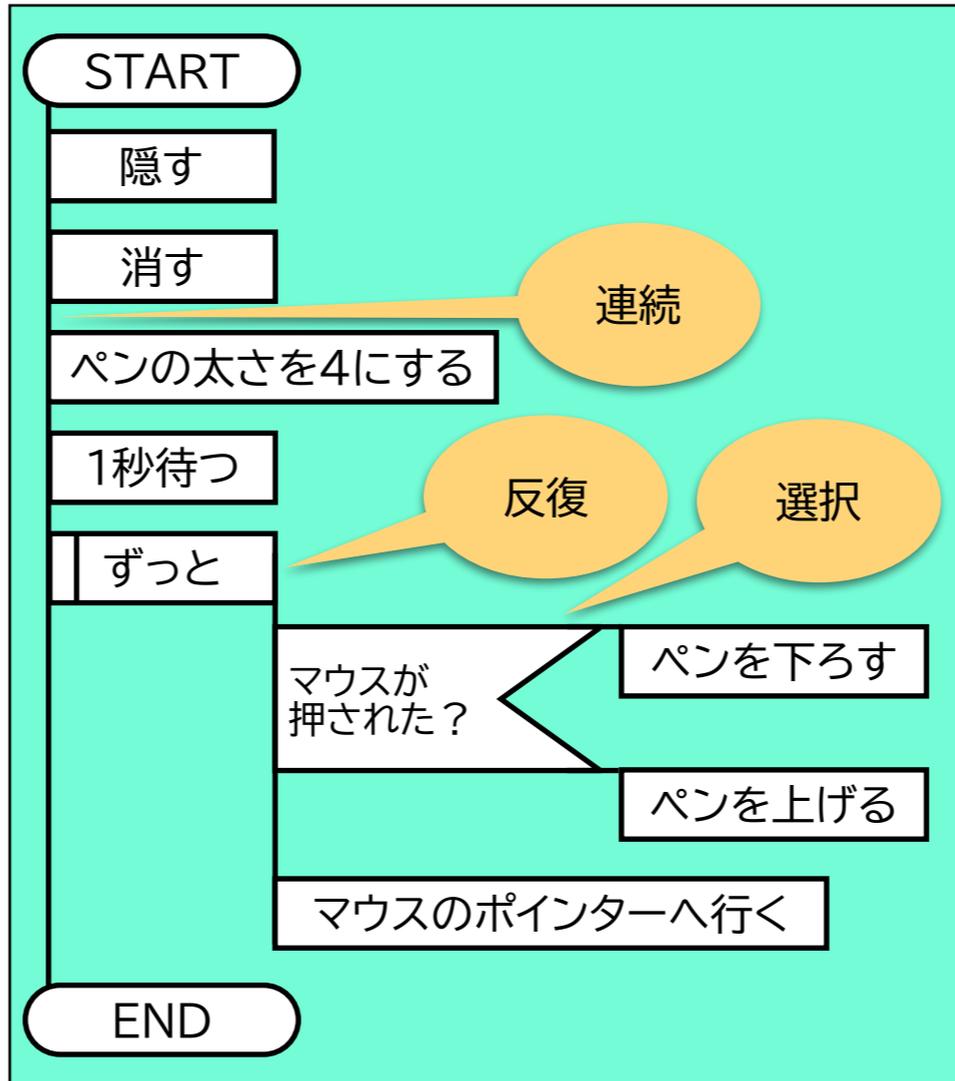
Flow Chart(流れ図)

非構造化プログラミング(BASIC等)

条件ジャンプ: IF THEN

無条件ジャンプ: GOTO

## PAD



PAD: Problem Analysis Diagram

構造化プログラミングの表記用に  
日立製作所が発明[1979]

構造化プログラミング(C,Java等)

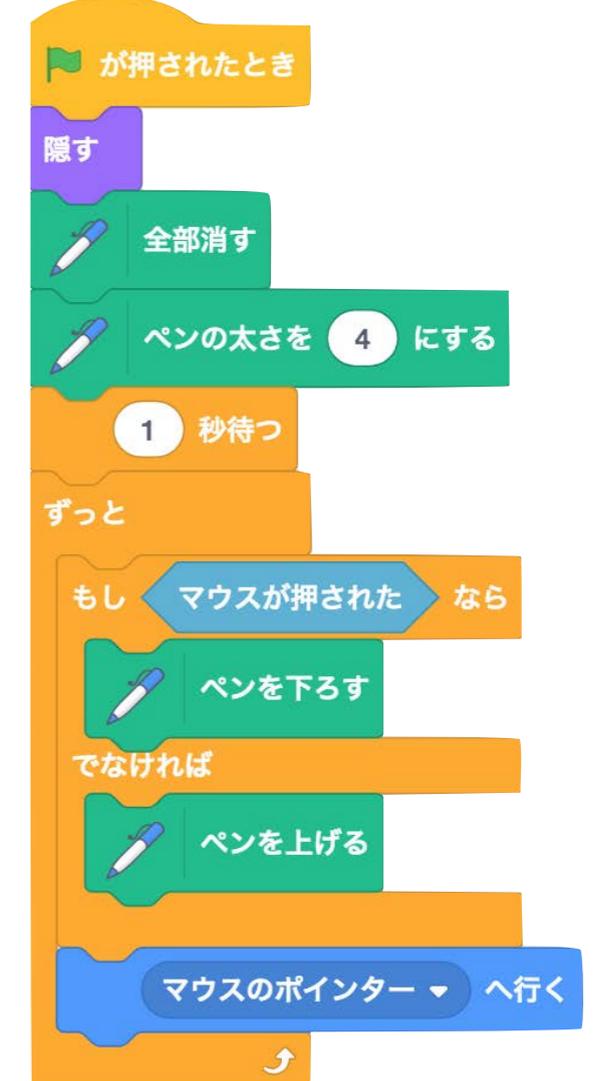
ダイクストラが提唱[1960年代後半]

連続(Sequence)

選択(Selection): if then else

反復(Iteration): for, while, do while

## SCRATCH



PADのような図をマウスで  
描くだけで実行できる

プログラミング入門に最適

- ・アルゴリズム学習
- ・構造化プログラミング入門
- ・オブジェクト指向入門 etc.

連続は順次・逐次・接続ともいう

選択は条件分岐ともいう

反復は繰り返し・ループともいう

# おまけ:Pen Pineapple Apple Pen! ⇨ PPAP



- ① コードの「変数」をクリック
- ② 「変数を作る」をクリック
- ③ 新しい変数名:「文字列」、OKをクリック
- ④ 「変数を作る」をクリック
- ⑤ 新しい変数名:「i」、OKをクリック
- ⑥ 図6のプログラムを作成
- ⑦ 文字列のi番目の文字=「スペース(空白)」があることに注意!
- ⑧ 実行!
- ⑨ 「Pen Pineapple Apple Pen!」と入力
- ⑩ どうなりましたか?

スペース(空白)がある

図6

他にも「Social Networking Service」「Compact Disc」「Play Station」…  
思いつくものを入力してみよう!

# 部品一覧

## 動き

10 歩動かす

15 度回す

15 度回す

どこかの場所へ行く

x座標を 0、y座標を 0 にする

1 秒で どこかの場所へ行く

1 秒でx座標を 0 に、y座標を

90 度に向ける

マウスのポインターへ向ける

x座標を 10 ずつ変える

x座標を 0 にする

y座標を 10 ずつ変える

y座標を 0 にする

もし端に着いたら、跳ね返る

回転方法を 左右のみ にする

x座標

y座標

向き

スプライトを動かす・回す

## 見た目

こんにちは! と 2 秒言う

こんにちは! と言う

うーん... と 2 秒考える

うーん... と考える

コスチュームを コスチューム1 にする

次のコスチュームにする

背景を 背景1 にする

次の背景にする

大きさを 10 ずつ変える

大きさを 100 %にする

色 の効果を 25 ずつ変える

色 の効果を 0 にする

画像効果をなくす

最前面へ移動する

1 層 手前に出す

コスチュームの 番号

背景の 番号

大きさ

値の出力／  
スプライトの  
見た目を変更

## 音

終わるまで ニャー の音を鳴らす

ニャー の音を鳴らす

すべての音を止める

ピッチ の効果を 10 ずつ変える

ピッチ の効果を 100 にする

音の効果をなくす

音量を -10 ずつ変える

音量を 100 %にする

音量

音の制御

## 変数

変数を作る

変数 を 0 にする

変数 を 1 ずつ変える

変数 変数 を表示する

変数 変数 を隠す

リストを作る

変数とリストの  
作成・操作

## イベント

が押されたとき

スペース キーが押されたとき

このスプライトが押されたとき

背景が 背景1 になったとき

音量 > 10 のとき

メッセージ1 を受け取ったとき

メッセージ1 を送る

メッセージ1 を送って待つ

プログラム  
スタート／  
メッセージ  
送受信

## ブロック定義

ブロックを作る

ブロック名

ブロックの  
作成・定義

## 制御

1 秒待つ

10 回繰り返す

ずっと

もし なら

もし なら

でなければ

まで待つ

まで繰り返す

すべてを止める

クローンされたとき

自分自身 のクローンを作る

このクローンを削除する

繰り返し／  
条件分岐

## 調べる

マウスのポインター に触れた

色に触れた

色が 色に触れた

マウスのポインター までの距離

What's your name? と聞いて待つ

答え

スペース キーが押された

マウスが押された

マウスのx座標

マウスのy座標

ドラッグ できる ようにする

音量

タイマー

タイマーをリセット

ステージ の 背景 #

現在の 年

2000年からの日数

ユーザー名

値の入力／  
キー入力／  
マウス入力

## 演算

+ /

1 から 10 までの乱数

> 50

< 50

= 50

>

<

=

かつ

または

ではない

apple と banana

apple の 1 番目の文字

apple の長さ

apple に a が含まれる

を で割った余り

を四捨五入

の 絶対値

+-x÷  
数学関数

IT IoT

