

プログラミング入門 補足資料

変数を使った図形の移動・変形

理工学部 情報科学科 隅田 康明

sumida@ip.kyusan-u.ac.jp

変数を使って図形を動かす

- 変数を使った図形の動かし方、動く図形の増やし方について、講義資料やスライドとは少し違うやり方を紹介しています
- どこを変えたら動くか？
を確認してから変数を宣言していく方法です
- 慣れれば、いきなり変数宣言からで問題ないですが、どこを変数に変えたらいいか分からない、という場合には、このやり方も参考にしてみましょう

動く図形を作る手順

- まずは図形を変化させながら、
「変数」の宣言、代入になれていこう
1. まず動かない図形を描く
 2. どう動かしたいか決める
 3. 引数の何番目の数字を変えればよいかを特定する
 4. その引数にあった名前の変数を宣言する
 5. 引数の値と変数を入れ替える
 6. 変数に更新後の値を代入する

動く図形を作るときの手順（１） （右に動く図形を作る場合）

- プログラムの原型

```
void setup(){  
  size(400,400);  
}  
void draw(){  
  background(255);  
}
```

- まず、動かない図形を作る（drawの中に書く）

```
void setup(){  
  size(400,400);  
}  
void draw(){  
  background(255);  
  rect(50, 50, 20, 20); //まだ動かない  
}
```

sketch_200528a



動く図形を作るときの手順 (2)

(右に動く図形を作る場合)

- 動かしたい値の場所がどこかを考える
 - 右に動かしたい ⇨ 横の座標を変える ⇨ x 座標
 - 図形描画の命令の数値を変えて試しておく

```
void setup(){  
  size(400,400);  
}  
void draw(){  
  background(255);  
  rect(50, 50, 20, 20); //まだ動かない  
}
```

ここを変えると横に動く → ここを変数にする

まずは、どの値を変えればいいか？ をはっきりさせる

動く図形を作るときの手順（３） （右に動く図形を作る場合）

- 動かすための変数を宣言する
 - 右に動かしたい ⇨ 横の座標を変える ⇨ x 座標
 - 変数名：x1

```
int x1 = 50; //動かしたい図形の元の数値と同じ値を代入
```

```
void setup(){  
  size(400,400);  
}
```

```
void draw(){  
  background(255);  
  rect(50, 50, 20, 20); //まだ動かない  
}
```

同じ値にする（動き始める場所になる）

動く図形を作るときの手順（４） （右に動く図形を作る場合）

- 動かしたい値の場所を変数名と入れ替える

```
int x1 = 50; //動かしたい図形の元の数値と同じ値を代入
void setup(){
  size(400,400);
}
void draw(){
  background(255);
  rect(x1, 50, 20, 20); //まだ動かない
}
```

同じする（図形のx座標が変数になる）

動く図形を作るときの手順 (5)

(右に動く図形を作る場合)

- 変数の値を変えて、座標を変える

```
int x1 = 50; //動かしたい図形の元の数値と同じ値を代入
void setup(){
  size(400,400);
}
void draw(){
  background(255);
  rect(x1, 50, 20, 20); //x1 の値が変わることで動き始める
  x1 = x1 + 1;
}
```

四角形のx座標 (x1) を1ずつ増やす = 右に移動する

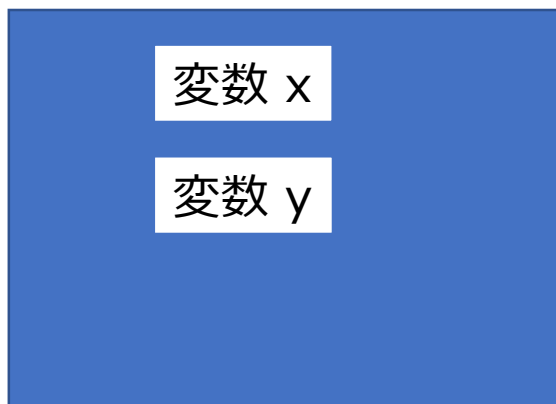
慣れるまでは、まず動かない図形を描いてから、どう動かすのかを考えよう

自分のやりやすい方法で図形を動かせるなら、それでも良い

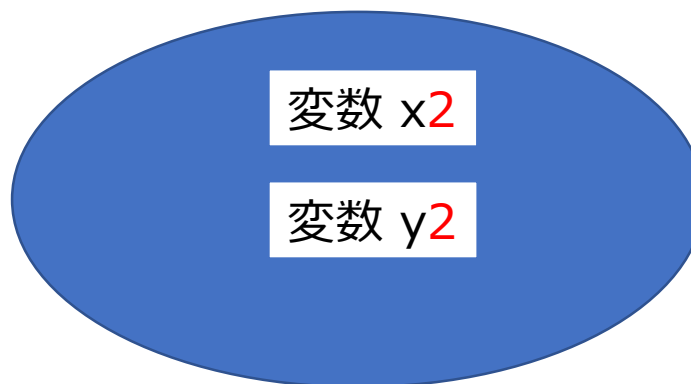
動く図形を増やす

- 図形と変数は 1 セット

四角の図形の変数



楕円の図形の変数



それぞれの図形用の変数を作らないといけない
同じ名前の変数は使わない

同じ速さで動く場合などで、変数を使いまわすことも出来るが、
変数の扱いに慣れるまでは、それぞれの図形用の変数とおこう

動く図形を増やす手順（１）

- 例：下に伸びる線を追加する場合
- まずは、動かない線を描く（動き始める前の状態）
 - どこを変えたらどう動くか、数値を変えて確かめておく

```
int x1 = 50; //動かしたい図形の元の数値と同じ値を代入
void setup(){
  size(400,400);
}
void draw(){
  background(255);
  rect(x1, 50, 20, 20); //x1 の値が変わることで動き始める
  x1 = x1 + 1;
  line(10, 10, 100, 10); //(10,10) から (100,10)に線を引く
}
```

sketch_200530a



動く図形を増やす手順（２）

- 動かすための変数を宣言する
 - 下に伸ばしたい ⇨ 終点のY座標を増やせばいい
 - 変数名：y2（2つ目の図形なので2とつけておく）

```
int x1 = 50; //動かしたい図形の元の数値と同じ値を代入
```

```
int y2 = 10;
```

```
void setup(){  
  size(400,400);  
}
```

```
void draw(){  
  background(255);  
  rect(x1, 50, 20, 20); //x1 の値が変わることで動き始める  
  x1 = x1 + 1;  
  line(10, 10, 100, 10); //(10,10) から (100,10)に線を引く  
}
```

同じする（終点のy座標が変数になる）

ここを変えると下に伸びる → ここを変数にする

動く図形を増やす手順（3）

- 動かしたい値の場所を変数名と入れ替える

```
int x1 = 50; //動かしたい図形の元の数値と同じ値を代入
```

```
int y2 = 10;
```

```
void setup(){  
  size(400,400);  
}
```

```
void draw(){  
  background(255);  
  rect(x1, 50, 20, 20); //x1 の値が変わることで動き始める  
  x1 = x1 + 1;  
  line(10, 10, 100, y2); //(10,10) から (100,10)に線を引く  
}
```

同じする（終点のy座標を作った変数に置き換え）

動く図形を増やす手順 (4)

- 変数の値を変えて、座標を変える

```
int x1 = 50; //動かしたい図形の元の数値と同じ値を代入
int y2 = 10;
void setup(){
  size(400,400);
}
void draw(){
  background(255);
  rect(x1, 50, 20, 20); //x1 の値が変わることで動き始める
  x1 = x1 + 1;
  line(10, 10, 100, y2); //(10,10) から (100,10)に線を引く
  y2 = y2 + 1;
}
```

線の終点のY座標を1ずつ増やす → 線が下に伸びていく

変数を使って、図形を変化させよう

- 色を変える
 - `stroke( 255, 0, 0, 255);`
 - 透明度を少しずつ減らすと・・・？
- 大きさを変える
 - `ellipse(200, 200, 10, 10);`
 - 幅と高さを増やしていくと・・・？
 - `ellipse(200, 200, s3, s3);`
のように同じ変数を使うこともできる