

プログラミング演習I（第6回）課題

• 基本① スケッチ名: math

- 自分の学生番号の下8ケタ(201****)の約数をすべて表示し、最後に約数の個数を表示するプログラムを作成してください。また、最後にすべての約数を足し合わせた数を表示するようにせよ。
- 数値の表示はprintln()を使おう。

例: 20130123の場合

20130123の約数は

1

3

13

39

516157

1548471

6710041

20130123

約数の数は8です

約数の総和は28904848です

[step1]

まずはwhile文(またはfor文)を使って、1から2013****までの数値を順に調べていく処理を書いてみよう。約数であるかどうかを調べるif文はどう書けばよいだろうか？

[step2]

「もし約数だったら」というif文の中に、その数値を表示する処理と、個数をカウントする処理と、合計を計算する処理を入れてみよう。

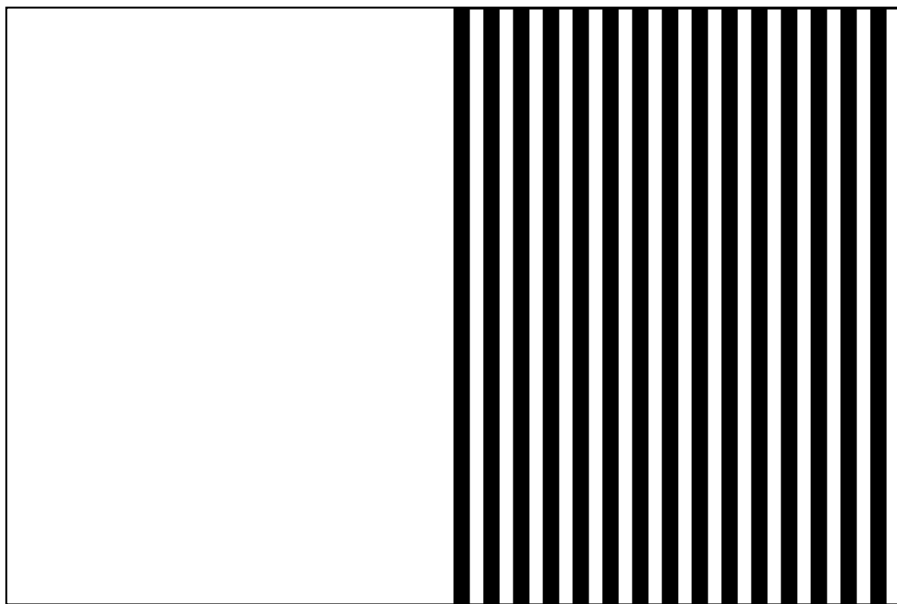
[step3]

繰り返し文の外に約数の個数と総和を表示する処理を入れよう。

プログラミング演習I（第6回）課題

• 基本② スケッチ名 : stripe

- 画面の【右半分】に黒帯と白帯を交互に並べるプログラムを作成してください。帯の幅は10にしてください。



[step1]

まずは塗り潰しなしの枠線だけの四角形を並べる処理を書いてみよう。画面の右半分だけに描かせるにはどうしたらいいだろうか？

[step2]

if文を使って塗り潰しの色が1つ目は黒、2つ目は白、3つ目は黒、4つ目は白…となるようにしてみよう。変数と条件分岐によって色指定を変えるにはどうしたらいいだろうか？

プログラミング演習I（第6回）課題

• 基本③ スケッチ名：fizzbuzz

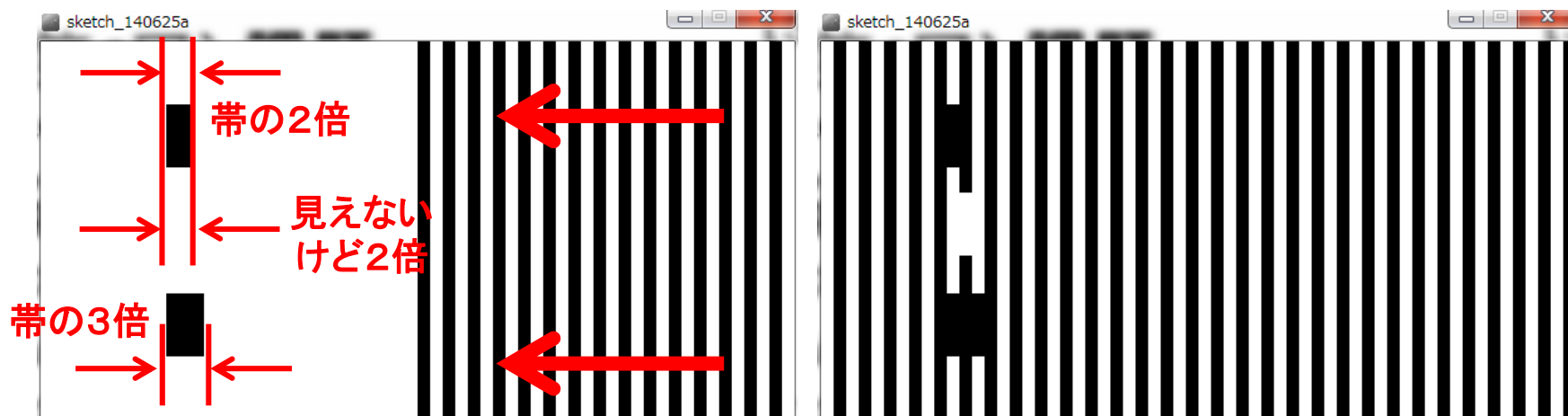
- 数字を1から100まで順に改行しながらカウントアップ表示するプログラムを作成せよ
- ただし、数字の値が3で割り切れる場合は「Fizz」と、5で割り切れる場合は「Buzz」と、3と5で割り切れる場合は「Fizz Buzz」と表示するようにせよ
- 出力例は右のとおり

```
1
2
Fizz
4
Buzz
Fizz
7
8
Fizz
Buzz
11
Fizz
13
14
Fizz Buzz
16
17
Fizz
19
Buzz
:
```

プログラミング演習I（第6回）課題

• 発展① スケッチ名 : footsteps

- 「フットステップ錯視」のプログラムを作ってください。
- 動画 : <http://www.youtube.com/watch?v=puWYJqFxPLM>
- 基本②(stripe)に、帯の2倍, 3倍の黒い四角形, 帯の2倍の白い四角形を配置し、帯が右から左に向かって等速で移動する処理を追加してください。帯は右端からどんどん溢れ出てくるようにせよ。
- また、マウスがウィンドウの左半分にある場合は帯の動きを停止させ、右半分にある場合は帯を動かすといったように停止・移動を繰り返すようにせよ。



プログラミング演習I（第6回）課題

• 発展② スケッチ名：fibonacci

- 最初の40個のフィボナッチ数のリストを表示するプログラムを作成せよ
- フィボナッチ数列とは、最初の2つの数字は0と1で、次の数は前の2つの合計の値となる（例えば最初の10個のフィボナッチ数列は 0, 1, 1, 2, 3, 5, 8, 13, 21, 34 となる）
- また、1行あたり10個ずつ表示するようにせよ（1行の中ではカンマ区切りとせよ）

（出力例）

```
0, 1, 1, 2, 3, 5, 8, 13, 21, 34
55, 89, 144, 233, 377, 610, 987, 1597, 2584, 4181
6765, 10946, 17711, 28657, 46368, 75025, 121393, 196418, 317811, 514223
832040, 1346269, 2178309, 3524578, 5702887, 9227465, 14930352, 24157817
```

プログラミングの技

ふらぐ

フラグが立った！
フラグを立てろ！
フラグが折られた！
死亡フラグ

フラグとはプログラム上での
進行にまつわる条件
フラグの値に応じて処理の内容
を変化させることを意識する
どこに制御するための分岐が
必要かを考えよう！

```
void setup() {  
}  
  
int flag = 0;  
void draw() {  
    if( flag == 1 ){  
        // フラグが立っている時の処理  
    } else {  
        // フラグが立っていない時の処理  
    }  
}  
  
void mousePressed(){  
    // マウスプレスでフラグを立てる  
    flag = 1;  
}
```

フラグはしっかり使いこなしましょう！

プログラミングの技

- 0, 1, 2, 3, 0, 1, 2, 3, 0, 1 ... と値がループするプログラムを作るにはどうしたらいいだろうか？

① if文を使う方法

カウンタの値を1ずつ増やしていき、3を越えたら強制的に0に戻す。 →

```
void setup() {  
  
}  
  
int x = 0;  
  
void draw() {  
  println(x);  
  x = x + 1;  
  if ( x>3 ) {  
    x = 0;  
  }  
}
```

プログラミングの技

②「余り」を使う方法

1ずつ増えていくカウンタの値を4で割った余り $x\%4$ を使う。

0, 1, 2, ...N でループさせたいときは、N+1で割った余りを使う。

```
void setup() {  
  
}  
  
int x = 0;  
  
void draw() {  
  println(x);  
  x = x + 1;  
  x = x % 4;  
}
```

このテクを知っておくと便利です