

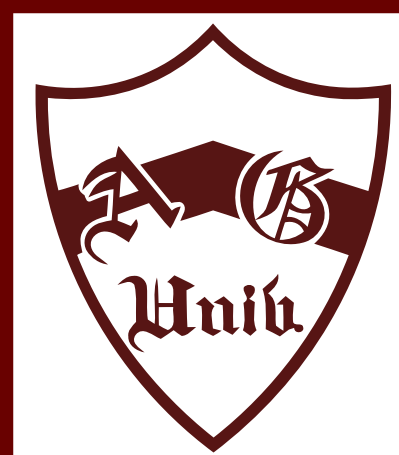
静止画の作成1

担当: 佐藤

計算機実習III

第1回: 演習課題解説





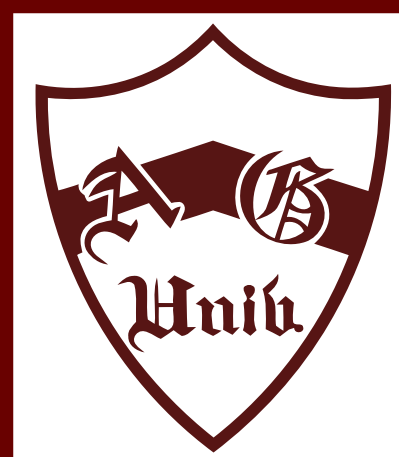
課題1-1

解答例

システム変数を利用→汎用性UP

```
size(600, 150) ;  
for (int x = 10 ; x < width ; x += 10) {  
    if (x % 50 == 0)  
        if (x % 100 == 0) line(x, 0, x, 100) ;  
        else line(x, 0, x, 75) ;  
    else line(x, 0, x, 50) ;  
}
```

- ▶ 基本的に短い線を描く
- ▶ 5本ごとに長めの線を描く
- ▶ 10本ごとに長い線を描く
 - 重複する場合→より長い線を優先



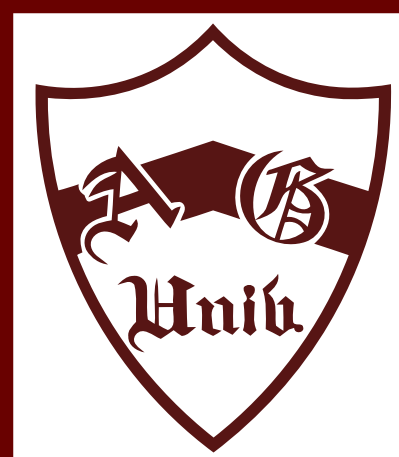
課題1-2

解答例

```
size(500, 500) ;
int length = width / 2 ; // height / 2
int centerX = width / 2, centerY = height / 2 ;
for (int i = 0 ; i < 5 ; i++) {
    // diamond
    quad(centerX, centerY - length,
        centerX + length, centerY,
        centerX, centerY + length,
        centerX - length, centerY) ;
    length /= 2 ;
    // square
    quad(centerX - length, centerY - length,
        centerX + length, centerY - length,
        centerX + length, centerY + length,
        centerX - length, centerY + length) ;
}
```

- ▶ 繰り返し利用する式は変数に代入して用いる
- ▶ 変数名は式の意味を推測しやすいものにする
 - キャメル記法
 - スネーク記法

- ▶ 菱形と四角形をこの順に描くことを大きさを小さくしながら5回繰り返す
- ▶ 菱形と四角形のいずれの頂点もウィンドウ中心(centerX, centerY)とそこからの長さ(length)で定式化できる



課題1-3

解答例

```
size(500, 500) ;
int centerX = width / 2, centerY = height / 2 ;
float r1 = 100, r2 = 250 ;
beginShape(QUAD_STRIP) ;
for (int i = 0 ; i <= 360 ; i += 10) {
  float theta = radians(i) ;
  float x1 = centerX + r1 * cos(theta) ;
  float y1 = centerY + r1 * sin(theta) ;
  float x2 = centerX + r2 * cos(theta) ;
  float y2 = centerY + r2 * sin(theta) ;
  vertex(x1, y1) ;
  vertex(x2, y2) ;
}
endShape() ;
```

QUAD_STRIPの性質上, 「=」がないと最後の区画が描かれない

極座標と直交座標の関係式を用いる際には, 極座標の原点の位置に注意

- ▶ 角度をdegree(度)で考える場合, radians()でradian(ラジアン)に変換して用いる
- ▶ ラジアンは実数値なので, 代入する変数の型はfloatにする