

# プログラム問題集(課題)

プログラミング入門2 第5回

# ■ 問題11

問題11 次のクラスMyRectangleを作成し動作を確認せよ。

## インスタンス変数

変数の型と名前	初期値	説明
int x	無し	長方形のx座標
int y	無し	長方形のy座標
int width	無し	長方形の幅
int height	無し	長方形の高さ

## インスタンスメソッド

戻り値の型	メソッド名(引数)	機能
void	showFields(int row)	インスタンス変数の内容を Spreadsheet の row行に表示する。
void	header()	Spreadsheet の先頭行に、ヘッダとして、X座標、Y座標、幅、高さという String を表示する。

## Q11DrawRectangles のメソッド

戻り値の型	メソッド名(引数)	機能
MyRectangle	create(int x, int y, int width, int height)	MyRectangleクラスのインスタンスを作成し、各インスタンス変数にそれぞれ引数の値を代入し、そのインスタンスを返す。

動作確認クラス:Q11DrawRectangles

# ■ 問題12

メソッドshowArea()を作成し動作を確認せよ。

新しく作成するインスタンスメソッド

返り値の型	メソッド名(引数)	機能
void	showArea(int row)	長方形の面積を計算し、その結果を Spreadsheet の row行4列に表示する
void	header()	Spreadsheet の先頭行に、ヘッダとして、X座標、Y座標、幅、高さに加えて、面積という String を表示する。



スプレッドシート

ファイル(E) ウィンドウ(W) ヘルプ(H)

A	B	C	D	E
X座標	Y座標	幅	高さ	面積
100	100	200	300	60000.0

(クラス名 : MyRectangle)  
動作確認クラス:Q12DrawRectangles

## ■ 問題13

長方形をキャンバスに描くメソッドdraw()を作成し動作を確認せよ。

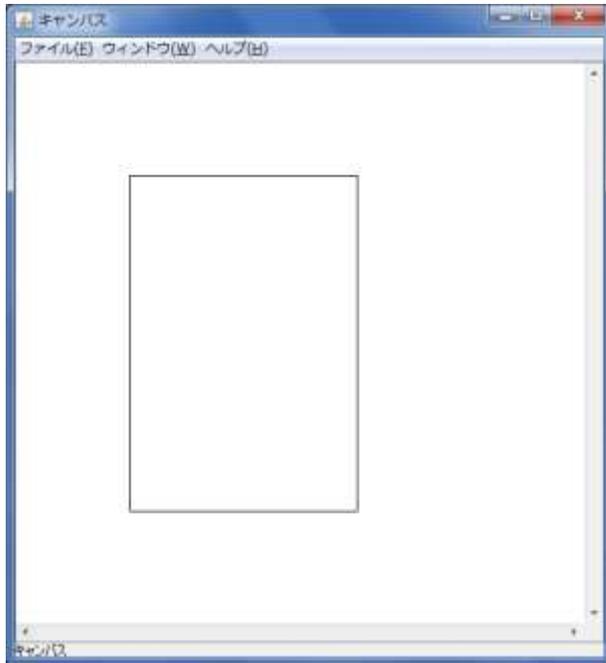
またメソッドarea() を作成し動作を確認せよ。

新しく作成するインスタンスメソッド

戻り値の型	メソッド名(引数)	機能
void	draw()	長方形をキャンバスに描く
double	area()	長方形の面積を計算し、その値を返す

(クラス名 : MyRectangle)  
動作確認クラス:Q13DrawRectangles

# 問題13実行例



A screenshot of a software window titled "スプレッドシート" (Spreadsheet). The window has a menu bar with "ファイル(F)", "ウィンドウ(W)", and "ヘルプ(H)". The main area is a spreadsheet with columns A through E. The first row contains headers: "X座標", "Y座標", "幅", "高さ", and "面積". The second row contains values: "100", "100", "200", "300", and "600000.0". The second row is highlighted with a red border.

A	B	C	D	E
X座標	Y座標	幅	高さ	面積
100	100	200	300	600000.0

## ■ 問題14

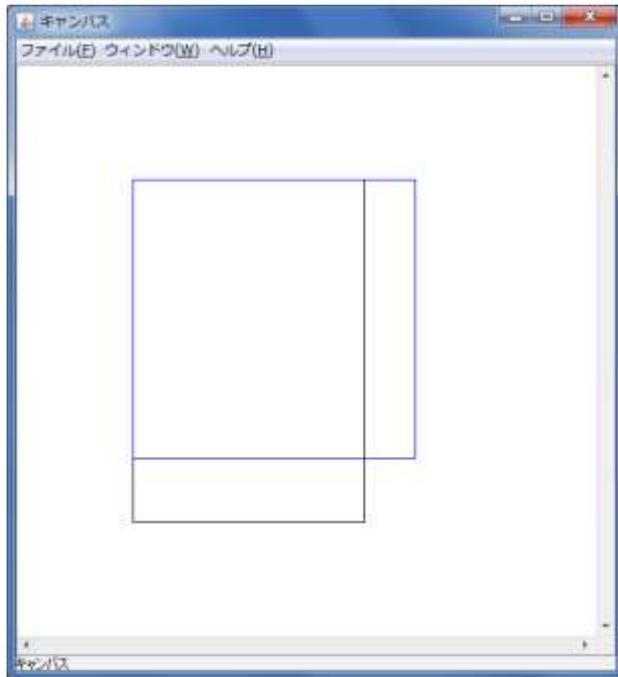
長方形を、面積を変えずに正方形に変形するメソッド  
toSquare()を作成し動作を確認せよ。

新しく作成するインスタンスメソッド

戻り値の型	メソッド名(引数)	機能
void	toSquare()	面積を計算し、その値になる正方形の一辺を求めてwidthとheightに代入する。

(クラス名 : MyRectangle)  
動作確認クラス:Q14DrawRectangles

# 問題14実行例



スプレッドシート

ファイル(E) ウィンドウ(W) ヘルプ(H)

A	B	C	D	E
X座標	Y座標	幅	高さ	面積
100	100	200	300	60000.0
100	100	244	244	59536.0

同じ面積(約)で正方形に変形  
※例では $X * X = 60000.0$ となる $X$ を求める

# ■ 問題15

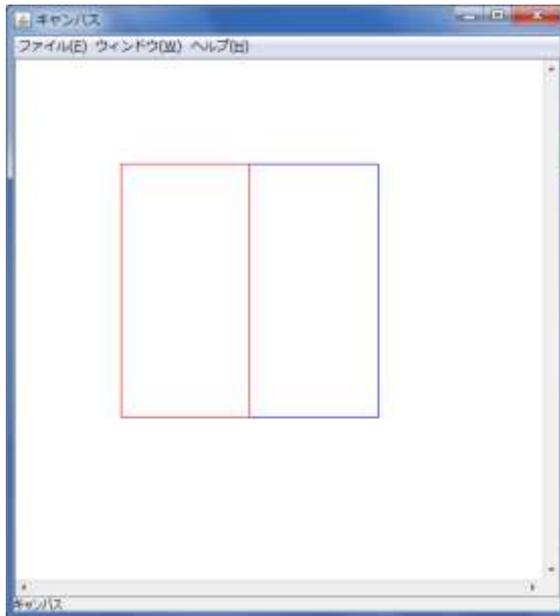
長方形の横幅の長さをfactor倍するメソッド  
expandWidthBy(double factor)を作成し動作を確認せよ。

インスタンスメソッド

戻り値の型	メソッド名(引数)	機能
void	expandWidthBy(double factor)	幅をfactor倍する

(クラス名 : MyRectangle)  
動作確認クラス: Q15DrawRectangles

# 問題15実行例



A screenshot of a spreadsheet window titled "スプレッドシート" (Spreadsheet). The spreadsheet has columns A through E. The first row contains headers: X座標 (X-coordinate), Y座標 (Y-coordinate), 幅 (Width), 高さ (Height), and 面積 (Area). The second row shows values: 100, 100, 244, 244, and 59536.0. The third row shows values: 100, 100, 122, 244, and 29768.0. A red box highlights the third row, and a blue arrow points from this box to the text below.

A	B	C	D	E
X座標	Y座標	幅	高さ	面積
100	100	244	244	59536.0
100	100	122	244	29768.0

高さは同じで幅をfactor倍(例では0.5倍)

## ■ 問題16

長方形の面積をareaに変えるメソッドsetArea(double area)を作成し動作を確認せよ。

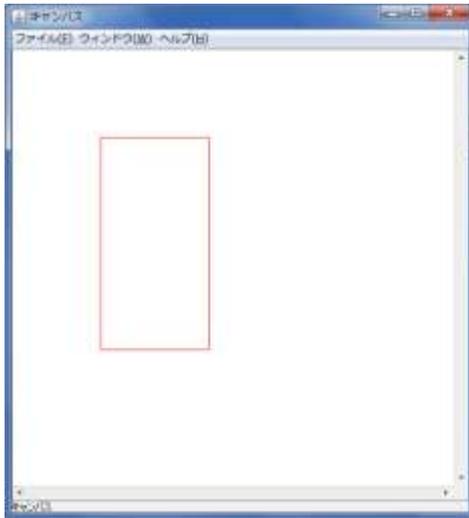
面積を変える際には、元の長方形の縦と横の長さの比を保つようにすること。

### インスタンスメソッド

戻り値の型	メソッド名(引数)	機能
void	setArea(double area)	変更前の辺の比率を変えずに幅と高さを変え、面積を引数areaにする。

(クラス名 : MyRectangle)  
動作確認クラス:Q16DrawRectangles

# 問題16実行例

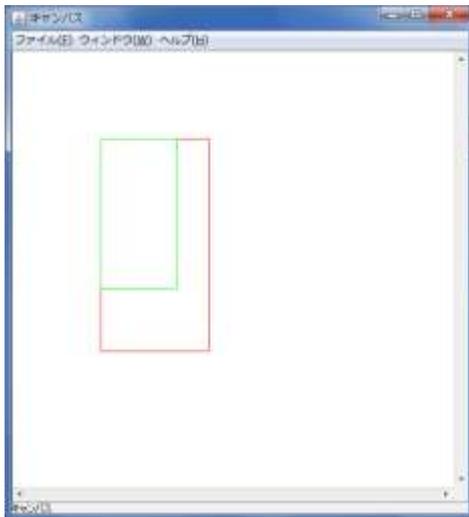


スプレッドシート

ファイル(E) ウィンドウ(W) ヘルプ(H)

A	B	C	D	E
X座標	Y座標	幅	高さ	面積
100	100	200	300	60000.0
100	100	244	244	59536.0
100	100	122	244	29768.0
100	100	86	172	14792.0

例では、面積を0.5倍にしている。(引数area/現在のarea)の平方根をそれぞれの辺にかけて比率を保っている



# ■ 問題21

問題21 次のクラスMyCarを作成し動作を確認せよ。

## インスタンス変数

変数の型と名前	初期値	説明
int x	無し	長方形のx座標
int y	無し	長方形のy座標
int bodyWidth	無し	車体の幅
int bodyHeight	無し	車体の高さ
int wheelDiameter	無し	車輪の直径

## インスタンスメソッド

戻り値の型	メソッド名(引数)	機能
void	showFields(int row)	インスタンス変数の内容を Spreadsheet の row 行に表示する。
void	header	Spreadsheet の先頭行にヘッダを表示する。

(クラス名 : MyCar)  
動作確認クラス:Q21DrawCar

続く

# ■ 問題21(続き)

問題21 次のクラスMyCarを作成し動作を確認せよ。

A	B	C	D	E	F
X座標	Y座標	車体の幅	車体の高さ	車輪の直径	
100	100	100	50	20	

(クラス名 : MyCar)  
動作確認クラス:Q21DrawCar

## ■ 問題22

車の高さ(車高+車輪)を返すメソッドgetHeight()を作成せよ。

### インスタンスメソッド

戻り値の型	メソッド名(引数)	機能
int	getHeight()	車の高さを計算し返す。



The screenshot shows a spreadsheet window titled 'スプレッドシート' (Spreadsheet). The menu bar includes 'ファイル(E)', 'ウィンドウ(W)', and 'ヘルプ(H)'. The spreadsheet contains the following data:

A	B	C	D	E	F
X座標	Y座標	車体の幅	車体の高さ	車輪の直径	車高
100	100	100	50	20	70

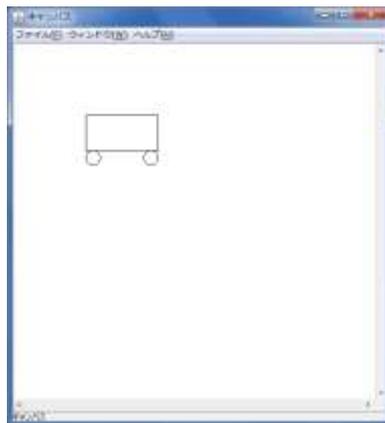
(クラス名: MyCar)  
動作確認クラス: Q22DrawCar

## ■ 問題23

車をキャンバスに描くメソッドdraw()を作成し動作を確認せよ。車を描く線の色は黒とする。

### インスタンスメソッド

戻り値の型	メソッド名(引数)	機能
void	draw()	車をキャンバスに描く



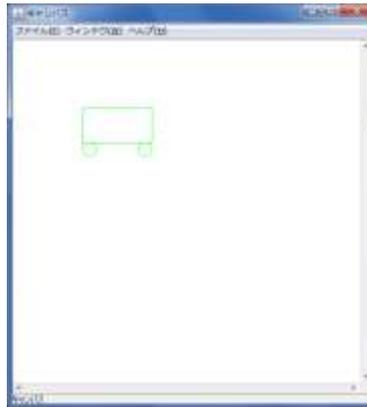
(クラス名 : MyCar)  
動作確認クラス:Q23DrawCar

## ■ 問題24

引数に車の色RGB値を渡し車を色付きで描くメソッド  
draw(int red, int green, int blue)を作成し動作を確認  
せよ。

### インスタンスメソッド

戻り値の型	メソッド名(引数)	機能
void	draw(int red, int green, int blue)	引数で与えられた色の線で車をキャンバスに 描く



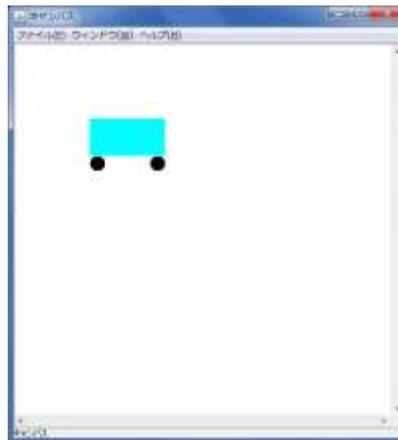
(クラス名 : MyCar)  
動作確認クラス:Q24DrawCar

## ■ 問題25

引数に車の色RGB値を渡し車体を描くメソッドpaint(int red, int green, int blue)を作成し動作を確認せよ。

### インスタンスメソッド

戻り値の型	メソッド名(引数)	機能
void	paint(int red, int green, int blue)	引数で与えられた色の車をキャンバスに描く

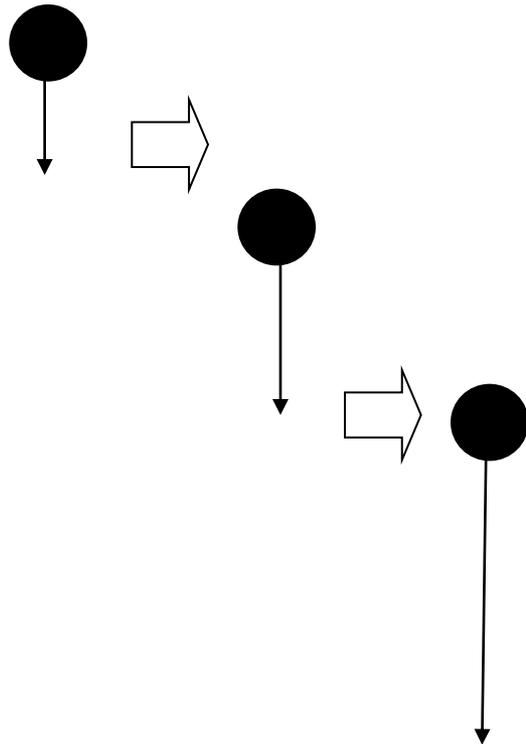


(クラス名 : MyCar)  
動作確認クラス:Q25DrawCar

## ■ 問題31(opt)

第4回で作成したMovingBallクラスのvx, vyをdouble型に変更し、drawBallメソッドは、インスタンスメソッドに変更して、落下運動をアニメーションで表現せよ。

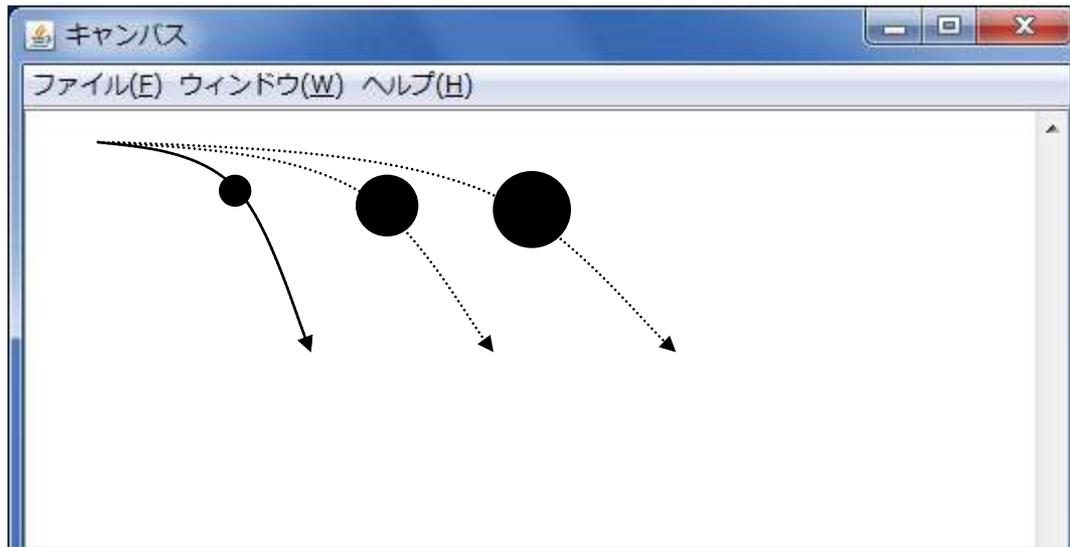
ヒント:まず、ボールの初速を0にし、加速度(vyの変化)を1.70に変えてみよ。



(クラス名Q31FallingBall)

## ■ 問題32(opt)

問題: 複数のボールを用いた落下運動のアニメーションを作成せよ。



ヒント: 複数のボールを保持させるための配列を用意する。  
(第4回, 課題41 参照)

(クラス名 Q32FallingBalls)

## ■ 問題41,42(opt)

- 問題41

ボールが拡張と収縮を繰り返すようなアニメーションを作成せよ。

ヒント: ボールが収縮している(-1)か、拡張している(1)かを示すインスタンス変数(int sOrE)を用意する。(やり方の一例)

- 問題42

拡張と収縮を繰り返すボールが移動するアニメーションを作成せよ。