Processingで作成した作品について

　　　　　　　　　　　　　　　　　　　　　　　　情報科学化27015450中島圭祐

今回作品を作るにあたって、元にしたものはスカッシュです。しかし、実際に製作にとりかかると、ボールの当たり判定が難しくx,yの両方向に動かすのではなく、片方を固定し、動かすことでブロック崩しのようなものを作ろうと決めました。

参考にした作品と違う点は、まず背景の色が変化するようにしたかったので、背景とラケットの色をマウスの位置によって、変更したことです。次に、キーを押すことでボールの数を２つに増やせるようにしました。スコアを付けたり、マウスをクリックすることでゲームオーバー後にリスタートできるようにしたかったのですが、丸写しをするのも嫌ですし、参考先の式を見てもあまり、あまり理解できなかったのでやめました。なので、ボールを増やす機能を追加したり、背景を変化させる事で楽しめるようにしました。

参考文献　<http://detail.chiebukuro.yahoo.co.jp/qa/question_detail/q1380003888>

表１

|  |  |
| --- | --- |
| 自分が作成した作品 | 参考にした作品 |
| 1: **int** radius=10**;**  2: **float** xpos**,**ypos**;**  3: **float** x2pos**,**y2pos**;**  4: **float** xspeed=7**;**  5: **float** yspeed=5**;**  6: **float** x2speed=4**;**  7: **float** y2speed=8**;**  8: **int** xdirection=1**;**  9: **int** ydirection=1**;**  10: **int** x2direction=1**;**  11: **int** y2direction=1**;**  12: **int** z**;**  13: **int** z2**;**  14: **int** z3**;**  15: **void** setup**(){**  16: size**(**640**,**360**);**  17: noStroke**();**  18: ellipseMode**(**RADIUS**);**  19: xpos = width/2**;**  20: ypos = height/2**;**  21: fill**(**255**);**  22: strokeWeight**(**2**);**  23: ellipseMode**(**RADIUS**);**  24: xpos=width/2**;**  25: ypos=width/2**;**  26: x2pos=width/2**;**  27: y2pos=width/2**;**  28: **}**  29: **void** draw**(){**  30: xpos = xpos + **(** xspeed \* xdirection **);**  31: ypos = ypos + **(** yspeed \* ydirection **);**  32: background**(**0**,**mouseX**,**mouseY**);**  33: stroke**(**mouseX**,**0**,**0**);**  34: rect**(**20**,**mouseY**,**8**,**80**);**  35: ellipse**(**xpos**,** ypos**,** radius**,** radius**);**  36: **if** **(**xpos > width-radius || xpos < radius**)** **{**  37: xdirection \*= -1**;**  38: **}**  39: **if** **(**ypos > height-radius || ypos < radius**)** **{**  40: ydirection \*= -1**;**  41: **}**  42: **if** **(**xpos<= 34**)** **{**  43: **if** **(**ypos >= mouseY && ypos <= mouseY + 80**)** **{**  44: xdirection \*= -1**;**  45: **}**  46: **}**  47: **if(**keyPressed**){**  48: z =1**;**  49: **}**  50:  51: **if(**z==1**){**  52: x2pos = x2pos + **(** x2speed \* x2direction **);**  53: y2pos = y2pos + **(** y2speed \* y2direction **);**  54: ellipse**(**x2pos**,** y2pos**,** radius**,** radius**);**  55: **}**  56:  57: **if** **(**x2pos > width-radius || x2pos < radius**)** **{**  58: x2direction \*= -1**;**  59: **}**  60: **if** **(**y2pos > height-radius || y2pos < radius**)** **{**  61: y2direction \*= -1**;**  62: **}**  63: **if** **(**x2pos<= 34**)** **{**  64: **if** **(**y2pos >= mouseY && y2pos <= mouseY + 80**)** **{**  65: x2direction \*= -1**;**  66: **}**  67: **}** **if((**xpos<20**)**||**(**x2pos<20**)){**  68: textSize**(**30**);**  69: fill**(**255**);**  70: text**(**"GAME OVER"**,**240**,**180**);**  71: noLoop**();**  72: **}**  73: **if(**mousePressed**){**  74: z2=1**;**  75: **}**  76: **if(**z2==1**){**  77: loop**();**  78: **}**  79: **}** | 1: **void** setup**()**  2: **{**  3: size**(**400**,** 400**);** *// ウィンドウの大きさを決める*  4: **}**  5: **float** x = 10**;**  6: **float** y = 10**;**  7: **float** dx = 2**;**  8: **float** dy = 2**;**  9: **float** dd = 1.2**;** *//ラケットで打ちかえしたときの速度上昇 1.2で20%上昇*  10: **int** ball\_stock = 2**;**  11: **int** count = 0**;**  12:  13: **void** draw**()** **{**  14:  15: **if** **(**x > width**)** **{** *// x座標をチェックしている*  16: dx = -dx**;** *// 右端を超えていたらボールの速度を左向きにする*  17: **}** **else** **if** **(**x < 0**)** **{**  18: dx = -dx**;** *// 左端を超えていたらボールの速度を右向きにする*  19: **}**  20: **if** **(**y > height**)** **{** *// y座標のチェック*  21: dy = -dy**;** *// 下端を超えていたらボールの速度を上方向にする*  22: **}** **else** **if** **(**y < 0**)** **{**  23: dy = -dy**;** *// 上端を超えていたらボールの速度を下方向にする*  24: **}**  25: x = x + dx**;** *// ボールのx座標を速度の分だけ変化させる*  26: y = y + dy**;** *// ボールのy座標を速度の分だけ変化させる*  27: noStroke**();***// 矩形に枠をつけない*  28: background**(**255**);** *// 背景を塗りつぶす*  29: smooth**();**  30: fill**(**0**,**0**,**255**,**160**);**  31: ellipse**(**x**,** y**,** 20**,** 20**);***// ballを3×3の大きさの正方形として描く*  32: smooth**();**  33: fill**(**0**,**0**,**255**,**160**);**  34: rect**(**mouseX**,** 380**,** 50**,** 10**);** *// padを50×3の大きさの正方形として描く*  35: fill**(**0**,**0**,**255**,**160**);**  36: textSize**(**18**);**  37: text**(**"SCORE: "+nf**(**count**,**4**),**20**,**20**);** *// カウントの表示*  38: fill**(**0**,**0**,**255**,**160**);**  39: textSize**(**18**);**  40: text**(**"BALL LEFT:"+ball\_stock**,**280**,**20**);**  41: **if** **(**y >= 380**)** **{** *// ボールがラケットの位置より低いときだけ処理をする*  42: **if** **(**x >= mouseX && x <= mouseX + 50**)** **{** *// ボールがラケットにあたった場合*  43: dy = -dd\*dy**;** *// ボールの速度を上向きにする + 速度上昇*  44: dx = dd\*dx**;** *// 速度上昇*  45: count = count + 1**;** *// カウントを増やす*  46: **}** **else** **if(**ball\_stock<1**){**  47: x = 0**;** *// ボールの位置を再設定(x座標を0にした)*  48: y = 0**;** *// ボールの位置を再設定(y座標を0にした)*  49: dx = 2**;** *// ボールの速度 x軸方向*  50: dy = 2**;** *// ボールの速度 y軸方向*  51: textSize**(**50**);**  52: ball\_stock = 0**;**  53: fill**(**0**,**0**,**255**,**160**);**  54: text**(**"GAME OVER"**,** 55**,** 200**);** *// カウントの表示//count = 0; // カウントリセット*  55: ball\_stock = 2**;**  56: noLoop**();**  57: **}else{**  58: y = 2**;**  59: ball\_stock--**;**  60: **if(** dx < 0 **)** dx=-2**;** **else** dx=2**;** *//x速度 初期値に戻す*  61: **if(** dy < 0 **)** dy=-2**;** **else** dy=2**;** *//y速度 初期値に戻す*  62: **}**  63: **}**  64: **}**  65: **void** mousePressed**()** **{**  66: loop**();**  67: count = 0**;**  68: **}** |