

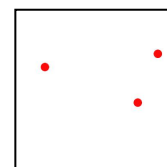
## 2018 年度情報メディア基盤ユニット 6 月 15 日分課題と宿題と事前学習問題

授業関連資料は <http://www.sato-lab.jp/imfu> からダウンロード出来ます。授業中に配布したプリントに誤りを見つけた際には、修正版をのせてあります。【自己確認問題】は先生や TA の人にチェックしてもらわなければならない必要はありません。空欄に答えを記入する問題以外は、出来たら先生か TA の人に確認をしてもらい、OK をもらったならスケッチをキャリアポートフォリオにアップロードして下さい。問題は難易度順に並んでいるわけではありません。

1. 【自己確認】本文中のサンプルプログラムを実行して見て下さい。本文中のサンプルプログラムを実行して見て下さい。特に、以下に個は実行して見て下さい。

9-5,9-6,9-7,9-9

2. 【目コピ問題】次のサンプル 9-3 や 9-4 を参考にして、3 つの赤色の円が左から右に移動するプログラムを作成して下さい。なお、円の左端がウィンドウの外側に出たら、左側から出現するようにして下さい。入力したプログラムはキャリアポートフォリオに提出して下さい。

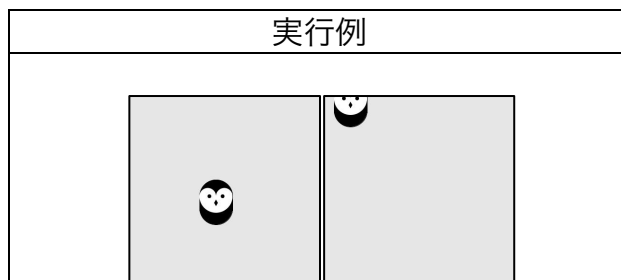


3. 【目コピ問題】もとのプログラムと同じ動作となるように、未完成プログラムの drawOwl 関数を作成して下さい。

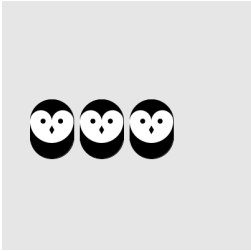
もとのプログラム	実行例
<pre>// 左目を描く void drawLeftEye(float x,float y){     fill(255);     ellipse(x-3.5,y,35,35);     fill(0);     ellipse(x,y,8,8); } // 右目を描く void drawRightEye(float x,float y){     fill(255);     ellipse(x+3.5,y,35,35);     fill(0);     ellipse(x,y,8,8); } void setup() {     size(200, 200); } void draw() {     background(230);     // 胴体     stroke(0);     fill(0);</pre>	

<pre> ellipse(110, 45, 70, 70); ellipse(110, 70, 70, 70); rect( 75, 45, 70, 30); noStroke(); fill(255); arc(110,45, 70, 70, 0, PI); // あご // 左目 drawLeftEye(96,45); // 右目 drawRightEye(124,45); quad(110, 52, 114, 59, 110, 66, 106, 59); // くちばし } </pre>	
<p>未完成のプログラム</p>	
<pre> // 左目を描く void drawLeftEye(float x,float y){   fill(255);   ellipse(x-3.5,y,35,35);   fill(0);   ellipse(x,y,8,8); } // 右目を描く void drawRightEye(float x,float y){   fill(255);   ellipse(x+3.5,y,35,35);   fill(0);   ellipse(x,y,8,8); } //フクロウを描く void drawOwl(float x,float y){ //この後に命令を追加して下さい。  } void setup() {   size(200, 200); } void draw() {   background(230);   drawOwl(110,52); } </pre>	

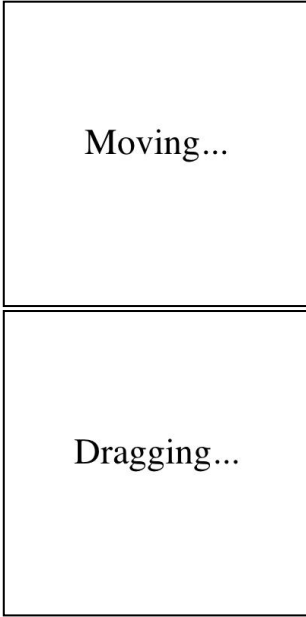
4. 【目コピ問題】 未完成プログラムは、問 4 で作成した drawOwl 関数などを利用して、マウスの動きにあわせて、フクロウが表示されるようなプログラムを作成して下さい。



5. 【目コピ問題】 未完成プログラムは、問 4 で作成した drawOwl 関数を利用して複数のフクロウを表示するものです。プログラムが動作するようにして下さい。

未完成のプログラム	実行例
<pre>// この部分に問 4 の関数を追加 void setup() {   size(400, 300); } void draw() {   background(230);   drawOwl(__(a)__,height/2);   drawOwl(__(b)__,height/2);   drawOwl(__(c)__,height/2); }</pre>	

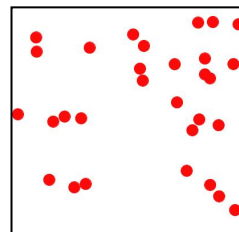
6. 【目コピ問題】 下のプログラムは、マウスをドラッグしている際には”Dragging...”と表示し、単にマウスを移動させている時には”Moving...”と表示するプログラムです。空欄を埋めて、プログラムを完成させて下さい。正しく動くプログラムをキャリアポートフォリオ上にも提出して下さい。

未完成のプログラム	実行例
<pre>PFont font; String msg; __(a) displaying = false;// 表示するかどうかを決める変数 void setup(){   size(400,400);   smooth();   font = loadFont(__(b)__);   textFont(font,48);   textAlign(CENTER); } void draw(){   background(255);   fill(0);   if(displaying){     text(msg,width/2,height/2);     displaying = false;   } } void mouseMoved(){   displaying = __(c);   msg = __(d); } void mouseDragged(){   displaying = __(e);   msg = __(f); }</pre>	

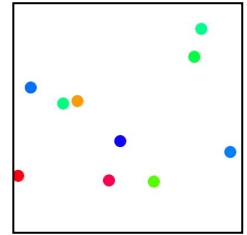
7. 次の2つのプログラムは同じ動作をするプログラムです。空欄を埋めて、プログラムを完成させて下さい。正しく動くプログラムをキャリアポートフォリオ上にも提出して下さい。

未完成のプログラムその1	未完成のプログラムその2
<pre>int[] scores; PFont font;  void setup(){   size(300,400);   scores = __ (a) __;   scores[0] = 10000;   scores[1] = 9000;   scores[2] = 5000;   scores[3] = 1000;   scores[4] = 30;   font = loadFont("Serif-48.vlw");   textFont(font,48); }  void draw(){   background(255);   fill(100);   textAlign(CENTER);   rectMode(CENTER);   float topOffset = __ (b) __;   for(int i = 0;i &lt; 5;i++){     __ (c) __;   } }  void mouseClicked(){   int newScore =     int(random(100000));   println(newScore);   for(int i = 0;i&lt;5;i++){     if(newScore &gt;= scores[i]){       for(int j=5-1;j &gt;i;j--){         scores[j] = scores[j-1];       }       scores[i] = newScore;       break;     }   } }</pre>	<pre>int[] scores = {__ (a) __}; PFont font;  void setup(){   size(300,400);   font = loadFont("Serif-48.vlw");   textFont(font,48); }  void draw(){   background(255);   fill(100);   textAlign(CENTER);   rectMode(CENTER);   float topOffset = (height - 60*5)/2     +textAscent();   for(int i = 0;i &lt; scores.__ (b) __;i++){     text(str(scores[i]),       width/2,topOffset+60*i,width,60);   } }  void mouseClicked(){   int newScore = int(random(100000));   println(newScore);   for(int i = 0;i&lt;5;i++){     if(newScore &gt;= scores[i]){       for(int j=5-1;j &gt;i;j--){         scores[j] = scores[j-1];       }       scores[i] = newScore;       break;     }   } }</pre>

8. 【目コピ問題】 問2のプログラムを変更して、30個の円が左から右に移動するようなプログラムを作成して下さい。作成したプログラムをキャリアポートフォリオ上にも提出して下さい。



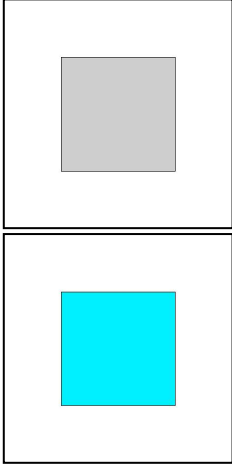
9. 【目コピ問題】 ランダムに決めた色の 10 個の円が左から右に移動するものです。空欄を埋めて、プログラムを完成させて下さい。



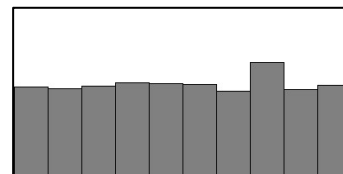
未完成のプログラム
<pre> __ (a) __ xBall; __ (a) __ yBall; __ (b) __ cBall; int numberOfBalls = __ (c) __; int radius=10; void setup(){   size(400,400);   colorMode(HSB,359,99,99);   smooth();   xBall = new float[numberOfBalls];   yBall = new float[numberOfBalls];   cBall = new color[numberOfBalls];   for(int i=0;i&lt;numberOfBalls;i++){     xBall[i] =       random(radius,width-radius);     yBall[i] =       random(radius,height-radius);     cBall[i] =       color(random(__ (d) __),99, 99);   } } void draw(){   background(0,0,99);   for(int i=0;i&lt;numberOfBalls;i++){     moveBall(i);     drawBall(i);   } } void moveBall(int idx){   xBall[__ (e) __] += 1;   if(__ (f) __ - __ (g) __ &gt; width){     xBall[__ (h) __] = -radius;   } } void drawBall(int idx){   stroke(cBall[__ (i) __]);   fill(cBall[__ (j) __]);   ellipse(xBall[__ (k) __],yBall[__ (l) __],     2*radius,2*radius); } </pre>

10. 【目コピ問題】 下のプログラムは、マウスボタンを押している間は、ランダムに決めた色の四角形を表示し、マウスボタンが押されていないときには、灰色の四角形を表示するものです。なお、描画する四角形の枠線の色は黒です。空欄を埋めて、プログラムを完成させて下さい。正しく動くプログラムをキャリアポートフォリオ上にも提出して下さい。

未完成のプログラム	実行例
-----------	-----

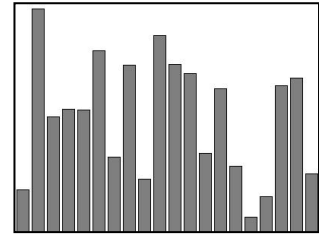
<pre> color fColor;//四角形の塗りつぶし色を保存している変数 void setup(){   size(400,400);   rectMode(CENTER);   colorMode(HSB,359,99,99);   fColor = color(0,0,80); } void draw(){   background(0,0,99);   stroke(__ (a) );   fill(__ (b) );   rect(width/2,height/2,width/2,height/2); } void __ (c) __(){   __ (d) __ = color(random(__ (e) __),99,99); } void mouseReleased(){   __ (f) __ = color(__ (g) __, __ (h) __, __ (i) __); } </pre>	
--	---

11. 【目コピ問題】 次のプログラムは、0～9 までの整数の乱数を発生させ、0～9 までの数値が何回現れたかを棒グラフとして表示するものです、空欄を埋めて、プログラムを完成させて下さい。このプログラムのある 1 箇所を書き換えると、0～19 までの整数の乱数を発生させ、0～19 までの数値が何回現れたかを棒グラフとして表示するものに変更することができます。どこを変更すればよいでしょうか？



未完成のプログラム
<pre> // rs は int 型の配列、rs[i]には、値 i の出現回数を保存しておく __ (a) __ rs; void setup(){   size(400,200);   rs = __ (b) __ __ (c) __ [10]; //10 個の要素を確保   for(int i=0;i&lt;rs.length;i++){ // 配列 rs の各要素の値を 0 にする     rs[__ (d) __] = __ (e) __;   } } void draw(){   background(255);   stroke(0);   fill(128);   //0～9 までの整数の乱数を発生させる   int r = __ (f) __ (__ (g) __ (rs.length));   __ (h) __; // 値 r の出現回数を 1 回増やす   int barWidth = int(width/rs.length);   for(int i=0;i&lt;rs.length;i++){     rect(barWidth*i,height-rs[i],__ (j) __,__ (k) __);   } } </pre>


12. 【目コピ問題】 次のプログラムは、float 型の配列変数 tops に長方形の上辺の Y 座標の値を保存し、それを用いて長方形を表示するものです。空欄を埋めて、プログラムを完成させて下さい。





未完成のプログラム
<pre> __ (a) __ tops; void setup(){   size(400,300);   tops = __ (b) __ __ (c) __ [20];   for(int i=0;i&lt;tops.__ (d) __;i++){     tops[i] = random(height);   } } void draw(){   background(255);   stroke(0);   fill(127);   for(int i=0;i&lt;tops.__ (e) __;i++){     rect(20*i+2,__ (f) __,16,height);   } } </pre>

13. 【目コピ問題】 下のプログラムは、ストップウォッチのような機能をもったプログラムです。マウスボタンをクリックすると時間の計測を始め、もう一度マウスをクリックすると計測を中止します。マウスのクリック時に、その時間を startTime に保存し、その時間と現在の時間との差を計算し、経過時間を数字で表示している。なお、秒未満の部分は数字でなく、扇型の表示で表している（映画などのカウントダウンで出てくるようなやつ）。この扇型は、時計の 12 時の場所から描画スタートします。そのスタートの場所の角度は  $3*PI/2(=3*HALF\_PI)$  となり、時間に応じた描画角度は map 関数を用いて計算しています。正しく動くプログラムをキャリアポートフォリオ上にも提出して下さい。

未完成のプログラム	実行例
<pre> int startTime; __ (a) __ doing = false; // 時間計測中は true にする PFont font; void setup(){   size(400,400);   font = __ (b) __;   textFont(font,48);   textAlign(CENTER); } void draw(){   background(255);   if(doing){ </pre>	<div style="border: 1px solid black; width: 150px; height: 100px; margin: 0 auto; display: flex; align-items: center; justify-content: center;"> <div style="text-align: center;"> <p>0m0s</p> </div> </div>

<pre> int ms = millis()- (c) ; fill(0); text(millisToMS( (d) ),width/2,height/5); float theta = map( (e) ,0,1000,0,2*PI); stroke(0); fill(50); arc(width/2,0.6*height, width/2,height/2,  (f) , (g) + (h) ); } } (i) millisToMS(int ms){ int ts = ms / (j) ;// ミリ秒を秒に変換 int s = (k) ; // ts から秒の部分を求める int m = (l) ; // ts から分の部分を求める (m) str(m) + "m" + str(s) + "s"; } void (n) (){ if(doing == false){ doing = (o) ; startTime = (p) ; }else{ doing = (q) ; } } } </pre>	<div style="border: 1px solid black; padding: 5px; width: fit-content; margin: 0 auto;"> <p>0m2s</p>  </div>
---	---

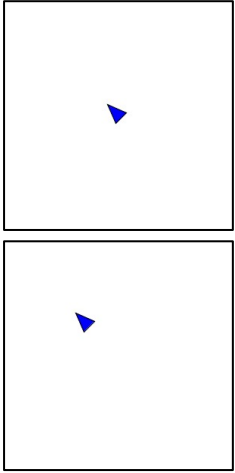
14. 【工夫問題】 ランダムな位置に四角形か円を表示し、その図形をクリックしたら、別の位置に図形を表示するという動作を繰り返すプログラムを作成してください。また、図形をクリックした回数を表示するような機能も付け加えてください。
15. 【目コピ問題】 次のプログラムは、左右のカーソルキーが押されている時のみ、ウインドウの中央に表示されている三角形が回転するプログラムです。左カーソルキー (←) を押すと反時計回り、右カーソルキー (→) を押すと時計回りに回転します。空欄を埋めて、プログラムを完成させて下さい。

未完成のプログラム	実行例
<pre> float angle; // 回転角度を表す変数 float angleVel; // 回転角度の増分を指定する変数 void setup(){ size(300,300); angle = 0; angleVel = 0; } void draw(){ background(255); translate(width/2,height/2); (a) ( (b) (angle)); fill(0,0,255); triangle(0,-20,-10,5,10,5); </pre>	<div style="border: 1px solid black; width: 100%; height: 100%; position: relative;"> <div style="position: absolute; top: 50%; left: 50%; transform: translate(-50%, -50%);">  </div> </div> <div style="border: 1px solid black; width: 100%; height: 100%; position: relative;"> <div style="position: absolute; top: 50%; left: 50%; transform: translate(-50%, -50%);">  </div> </div>



<pre> angle += __(c)__; } void keyPressed(){   if(key == __(d)){     if(keyCode == __(e)){       angleVel = -1;     }else if(keyCode == __(f)){       angleVel = 1;     }   } } void __(g)(){   if(key == __(d)){     if(keyCode == __(e)    keyCode == __(f)){       angleVel = __(h);     }   } } } </pre>	
--	--

16. 【目コピ問題】 次のプログラムは、問 15 のプログラムに変更を加え、左右のカーソルキーが押されている時のみ、ウインドウの中央に表示されている三角形が回転し、上カーソルキー（↑）が押されている間は、前に進むようなプログラムです。また、r キーまたは R キーを押すと、三角形の位置などが初期状態となります。空欄を埋めて、プログラムを完成させて下さい。

未完成のプログラム	実行例
<pre> float angle; // 回転角度を表す変数 float angleVel; // 回転角度の増分を指定する変数 float xPos; // 三角形の位置座標 float yPos; float vx; // 三角形の移動ベクトル float vy; float speed; // 三角形の速度 __(a) initTri(){   angle = 0;   angleVel = 0;   xPos = width/2;   yPos = height/2;   vx = vy = 0;   speed = 0; } void setup(){   size(300,300);   initTri(); } void drawShip(){   pushMatrix();   translate(xPos,yPos); </pre>	

```

    (b) ;
    fill(0,0,255);
    triangle(0,-20,-10,5,10,5);
    popMatrix();
}
void draw(){
    background(255);
    drawShip();
    angle += angleVel;
    xPos += (c) ;
    yPos += (d) ;
}
void keyPressed(){
    if(key == (e) || key == (f) ){
        (g) ;
    }else if(key == CODED){
        if(keyCode == LEFT){
            angleVel = -1;
        }else if(keyCode == RIGHT){
            angleVel = 1;
        }else if(keyCode == UP){
            float theta = radians((h) );
            speed = 2;
            vx = speed * cos(theta);
            vy = speed * sin(theta);
        }
    }
}
void keyReleased(){
    if(key == CODED){
        if(keyCode == LEFT || keyCode == RIGHT){
            angleVel = 0;
        }else if(keyCode == UP){
            vx = (i) ;
            vy = (j) ;
            speed = 0;
        }
    }
}
}
}

```

17. 【目コピ問題】 次の2つプログラムは、ともに String 型の msgs に入っている文字列をランダムに選び、右から左に動くようなものです。空欄を埋めて、プログラムを完成させて下さい。

Kanagawa

未完成のプログラムその 1	未完成のプログラムその 2
<pre> (a) msgs = {"Kanagawa",             "Institute",             "of",             "Technology"};  PFont font; int msgId; </pre>	<pre> (a) msgs = {"Kanagawa",             "Institute",             "of",             "Technology"};  PFont font; (b) msg; </pre>

```

int x,y;
void setup(){
  size(400,200);
  smooth();
  font = loadFont(___ (b) ___);
  textFont(font,36);
  msgId = int(random(msgs.___(c)___));
  y = int(random(height));
  x = width;
}
void draw(){
  background(255);
  x--;
  if(x+textWidth(___ (d) ___) < 0){
    x = width;
    y = int(random(textAscent(),
      height-textDescent()));
    msgId= ___ (e) ___(random(msgs.
  (f) ___));
  }
  fill(0);
  text(___ (g) ___,x,y);
}

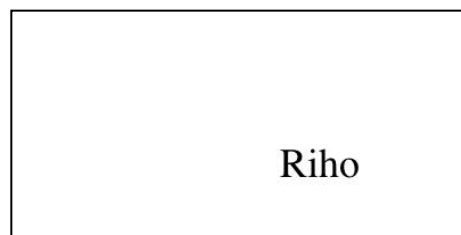
```

```

int x,y;
void setup(){
  size(400,200);
  smooth();
  font = loadFont(___ (c) ___);
  textFont(font,36);
  msg =msgs[int(random(msgs.___(d)___))];
  y = int(random(height));
  x = width;
}
void draw(){
  background(255);
  x--;
  if(x+textWidth(___ (e) ___) < 0){
    x = width;
    y =int(random(textAscent(),
      height-textDescent()));
    msg=msgs[___ (f) ___(random(msgs.
  (g) ___))];
  }
  fill(0);
  text(___ (h) ___,x,y);
}

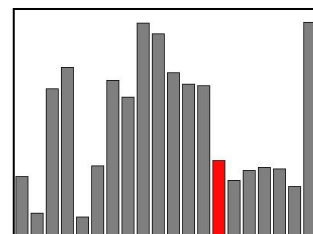
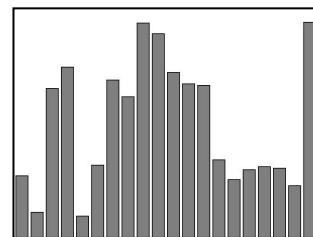
```

18. 【目コピ問題】 次のプログラムは、問 17 のプログラムを、6 つのメッセージをランダムに選択して、表示する動作となるように変更してください。表示されるメッセージの個数にこだわりはありませんので、6 個以上ならなんでもかまいません。



19. 【目コピ問題】 次のプログラムは、問 12 のプログラムに、マウスカーソルを長方形上に移動し、その場所でマウスボタンを押した際に、その長方形を赤色で表す機能を追加したものです。空欄を埋めて、プログラムを完成させて下さい。

int 型変数 choiceId は、どの長方形が選ばれているかを示す変数 (tops 配列の要素番号) です。長方形が選択されていないときには、-1 となっています。boolean 型の値を返す関数 inRect は点 (x,y) が左上の頂点座標が (xTopLeft, yTopLeft) で、横幅 w、縦幅 h の長方形の内部に含まれているときには true、そうでない時には false を返すものです。



#### 未完成のプログラム

```

___ (a) ___ tops;
int choiceId;
void setup(){

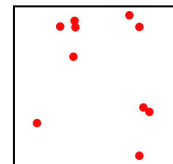
```

```

size(400,300);
tops = __ (b) __ __ (c) __[20];
for(int i=0;i<tops.__ (d) __;i++){
  tops[i] = random(height);
}
choiceId = __ (e) __;
}
void draw(){
  background(255);
  stroke(0);
  for(int i=0;i<tops.__ (f) __;i++){
    if(i == choiceId){
      fill(__ (g) __);
    }else{
      fill(__ (h) __);
    }
    rect(20*i+2,__ (i) __,16,height);
  }
}
__ (j) __ inRect(float x,float y,float xTopLeft,float yTopLeft,float
w,float h){
  if((__ (k) __ <= x && x <= __ (l) __) &&
    (__ (m) __ <= y && y <= __ (n) __)){
    return true;
  }else{
    return false;
  }
}
}
void __ (o) __(){
  choiceId = -1;
  for(int i=0;i< tops.length;i++){
    if(inRect(mouseX,mouseY,20*i+2,tops[i],16,height)){
      choiceId = __ (p) __;
      break;
    }
  }
}
}
void __ (q) __(){
  choiceId = -1;
}
}

```

20. 【目コピ問題】 次のプログラムは 10 個の円がランダムな位置から左右に移動し、ウインドウの左右で跳ね返るものです。空欄を埋めて、プログラムを完成させて下さい。



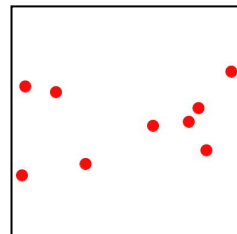
未完成のプログラム
<pre> float[] xBall; float[] yBall; float[] vBall; int numberOfBalls=10; float radius=10; void setup(){   size(400,400); </pre>

```

smooth();
xBall =
  (a) (b)[numberOfBalls];
yBall =
  (c) (d)[numberOfBalls];
vBall =
  (e) (f)[numberOfBalls];
for(int i=0;i<(g);i++){
  xBall[i] = width/2;
  yBall[i] = random(radius,
                    height-radius);
  vBall[i] = random(1,2);
  if(random(10) <= 5){
    vBall[i] = -vBall[i];
  }
}
}
void draw(){
  background(255);
  for(int i=0;i<numberOfBalls;i++){
    moveBall(i);
    drawBall(i);
  }
}
void drawBall(int idx){
  stroke(255,10,10);
  fill(255,10,10);
  ellipse(xBall[(h)],yBall[(i)],
          2*radius,2*radius);
}
void moveBall(int idx){
  xBall[idx] += vBall[idx];
  if((j)){
    vBall[idx] = -vBall[idx];
    xBall[idx] = width-radius;
  }else if((k)){
    vBall[idx] = -vBall[idx];
    xBall[idx] = radius;
  }
}
}

```

21. 【目コピ問題】 次のプログラムは、マウスをクリックした場所に円が描かれ、左右に移動するようなプログラムです。空欄を埋めて、プログラムを完成させて下さい。このプログラムでは、何回かクリックを繰り返していると、ある問題が発生します。どのような問題が発生するのでしょうか？



#### 未完成のプログラム

```

float[] xBall;
float[] yBall;
float[] vBall;
int numberOfBalls=0;
int maxBalls = 1000;
float radius=10;
void setup(){

```

```

size(400,400);
smooth();
xBall = new float[maxBalls];
yBall = new float[maxBalls];
vBall = new float[maxBalls];
}
void draw(){
background(255);
for(int i=0;i<__(a);i++){
moveBall(i);
drawBall(i);
}
}
void drawBall(int idx){
stroke(255,10,10);
fill(255,10,10);
ellipse(xBall[__(b)],yBall[__(c)],2*radius,2*radius);
}
void moveBall(int idx){
xBall[idx] += vBall[idx];
if(__(d)){
vBall[idx] = -vBall[idx];
xBall[idx] = width-radius;
}else if(__(e)){
vBall[idx] = -vBall[idx];
xBall[idx] = radius;
}
}
}
void __(j)(){
xBall[numberOfBalls] = __(f);
yBall[numberOfBalls] = __(h);
vBall[numberOfBalls] = random(1,2);
if(random(10) <= 5){
vBall[numberOfBalls] = -vBall[numberOfBalls];
}
__(i)++;
}
}

```

22. 【工夫問題】 問 21 のプログラムを変更し、maxBalls 以上の円を表示する場合には、一番古い円の表示をやめ、新たにクリックした場所から移動を開始する円を付け加えるものです。つまり、最大で maxBalls 個の円しか表示されないようになっています。空欄を埋めて、プログラムを完成させて下さい。

#### 未完成のプログラム

```

float[] xBall;
float[] yBall;
float[] vBall;
int numberOfBalls=0;
int maxBalls = 10;
float radius=10;
void setup(){
size(400,400);
smooth();

```

```

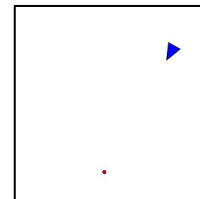
xBall = new float[maxBalls];
yBall = new float[maxBalls];
vBall = new float[maxBalls];
}
void draw(){
  background(255);
  for(int i=0;i<__(a__);i++){
    moveBall(i);
    drawBall(i);
  }
}
void drawBall(int idx){
  stroke(255,10,10);
  fill(255,10,10);
  ellipse(xBall[__(b__)],yBall[__(c__)],2*radius,2*radius);
}
void moveBall(int idx){
  xBall[idx] += vBall[idx];
  if(__(d__)){
    vBall[idx] = -vBall[idx];
    xBall[idx] = width-radius;
  }else if(__(e__)){
    vBall[idx] = -vBall[idx];
    xBall[idx] = radius;
  }
}
void __(f__)(){
  __(g__) last = numberOfBalls;
  if(numberOfBalls == __(h__)){
    last = numberOfBalls-1;
    for(int i=0;i < last;i++){
      xBall[__(i__)] = xBall[__(j__)];
      yBall[__(k__)] = yBall[__(l__)];
      vBall[__(m__)] = vBall[__(n__)];
    }
  }else{
    __(o__)++;
  }
  xBall[last] = __(p__);
  yBall[last] = __(q__);
  vBall[last] = random(1,2);
  if(random(10) <= 5){
    vBall[last] = -vBall[last];
  }
}
}

```

これ以降の問題は余裕のある人向けです。

23. 【目コピ問題】 問 18 のプログラムを変更して、メッセージ毎に異なった色で表示するようにしてください。作成したプログラムをキャリアポートフォリオ上にも提出して下さい。

24. 【工夫問題】 問 16 のプログラムを変更して、z キーまたは Z キーを押したら、三角形の頂点から赤い円が発射されるようなプログラムを作成して下さい。作成したプログラムをキャリアポートフォリオ上にも提出して下さい。



25. 【工夫問題】 配列を利用して、5 つの長方形（もしくは画像）を表示し、その中から正解の長方形（画像）を選ぶようなゲームを作成して下さい。作成したプログラムをキャリアポートフォリオ上にも提出して下さい。
26. 【工夫問題】 問 13 のプログラムでは、時間の計測を始めてから、再びマウスをクリックすると表示が消えてしまいます。そこで、表示が消えないようなプログラムを作成して下さい。つまり、「マウスクリック→計測開始→マウスクリック→表示したまま計測終了→マウスクリック→経過時間を 0 として、計測開始…」を繰り返すようなプログラムとしてください。
27. 【工夫問題】 問 13 のプログラムを変更し、秒未満の部分を表している扇型の部分は、「最初の 1 秒は黒で扇を描画→次の 1 秒は白で扇を描画（見かけ上は黒の部分が減っていく）」となるようにプログラムを変更してください。
28. 【工夫問題】 次のプログラムを大幅に修正し、「マウスをクリックすると、三角形の頂点から、真上に円盤状の球を発射する」というように変更をしたものを作成して下さい。もし可能なら、発射された弾が消えるまで、次の弾が発射出来ないようにするなどの機能を追加してください。

もとにするプログラム

```
void setup(){
  size(400,400);
}
void draw(){
  background(0);
  noStroke();
  fill(10,10,255);
  triangle(mouseX,mouseY-20,mouseX-14 ,mouseY+20, mouseX+14, mouseY+20);
}
```

29. 【工夫問題】 問 1 に、ウインドウの上方から赤色の円が落ちてくるような処理を加えて下さい。さらに、赤色の円と弾との衝突判定を行う処理を付け加え、衝突していると判定した場合には、再び赤色の円の位置をウインドウ最上部から再び下方向に移動するようにすること。また、衝突回数をカウントし、その回数を表示するようにして下さい。



## 宿題

宿題はキャリアポートフォリオ上に作成したプログラムをアップロードして下さい。

【目コピ問題】ランダムに決めた色の 10 個の円が右から左に移動するもようなプログラムを作成して下さい。作成したプログラムは、Processing の Tools メニュー内の Archive Sketch で保存したファイルを、キャリアポートフォリオ上にアップロードして下さい。

## 中間試験のお知らせ

**試験日：6月29日(金)1限の講義時間**

**試験範囲：6月15日(金)までの授業範**

形式：マークシート

持ち込み：プリント、ノート、ノート PC、本は可。人は不可。

注意事項：持ち込んだノート PC をネットに接続することは認めません。カンニング、相談などもダメです。

2018 年度情報メディア基盤ユニット

事前学習シート (6月19日分)

解答はキャリアポートフォリオで解答して下さい。

1. 配列がないと、どんなプログラムを作るときに不便でしょうか？