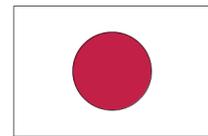


## 2019 年度情報メディア基盤ユニット 7 月 5 日分課題と事前学習シート

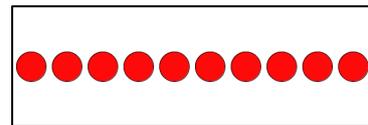
授業関連資料は <http://www.sato-lab.jp/imfu> からダウンロード出来ます。授業中に配布したプリントに誤りを見つけた際には、修正版をのせてあります。問 2 以降は、出来たら先生か TA の人に確認をしてもらって下さい。【自己確認問題】は先生や TA の人にチェックしてもらう必要はありません。出来たら先生か TA の人に確認をもらって下さい。問題の解答はキャリアポートフォリオにも提出して下さい。問題は難易度順に並んでいるわけではありません。

0. 【友達と一緒に確認問題】周りの同級生などと一緒に相談しながら、1 限の中間試験の自己採点をして下さい。マーク問題の自己採点の結果（1 問 1 点）をキャリアポートフォリオ上で記入して下さい。

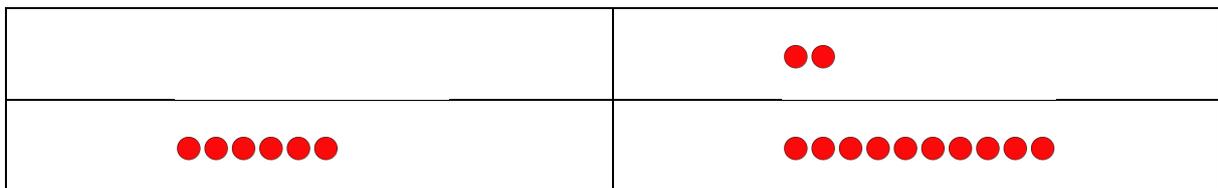
1. 【目コピ問題】日の丸のような画像を表示して下さい。日の丸の形は縦が横の 3 分の 2 の長方形で、日章の直径は縦の 5 分の 3 で中心は旗の中心となっている。



2. 円を 10 個表示するプログラムを作成せよ。円の間隔は 60、一番左の円の中心の X 座標は 30、半径は 25 とする。Y 座標は画面の高さの半分とする。



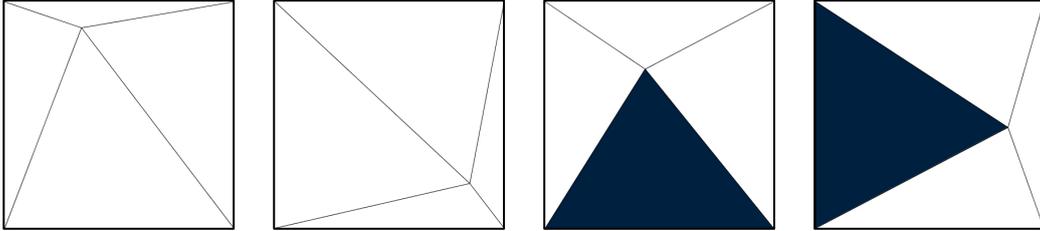
3. 円の間隔は 60、一番左の円の中心の X 座標は 30、半径は 25 とする。Y 座標は画面の高さの半分とする。円の中心の X 座標の値は mouseX より小さな範囲となるように円を描画せよ。



4. マウスボタンが押されていないときには緑色の円が、マウスボタンが押されているときには赤色の円が表示されるようなプログラムを作成して下さい。

マウスボタンが押されていないとき	マウスボタンが押されているとき
● ○	○ ●

5. 縦横 400 ドットの大きさのウィンドウを表示し、システム変数 `mouseX`, `mouseY`, `width`, `height` を使用して、マウスマウサーソルの位置でウィンドウの四隅から出ている 4 つの線分が交わるような表示となるようなプログラムを作成せよ。マウスマウサーソルのある位置とウィンドウの頂点がつくる 4 つの三角形の中で、面積が最も大きな三角形（複数ある場合にはどれか 1 つ）の内部を青色で塗りつぶすこと。



6. 【自己確認問題】 下のプログラムはトランプカードを表示するプログラムである。トランプカードの表示のために、`drawCard` 関数を定義している。この関数は 4 つの `int` 型引数をとる。

<code>void drawCard(int x,int y,int suit,int rank)</code>
<p>引数 <code>x</code> と <code>y</code> にはカードの表示位置の座標を指定する。</p> <p>引数 <code>suit</code> にはどの絵柄（クラブ、スペード、ハート、ダイヤ）を表示するかを指定する。この値が 0 の時にはクラブ、1 の時にはスペード、2 の時にはハート、3 の時にはダイヤのカードが表示される。</p> <p>引数 <code>rank</code> にはランクのカードを表示するのかを指定する。この値が 1 の時には A のカードが、2 の時には 2 のカードが、3 の時には 3 のカードが、…10 の時には 10 のカードが、11 の時には J のカードが、12 の時には Q のカードが、13 の時には K のカードが表示される。ただし、0 を指定すると <code>suit</code> の値にかかわらずカードの背面が表示される。</p>

この関数の利用するためには、2 つの `PImage` 型変数 `frontCards`、`backCard` と 2 つの `int` 型変数 `cardWidth`、`cardHeight` の値を正しく設定する必要がある。

このプログラムを入力して実行して見て下さい。このプログラムで使用している 2 つの画像ファイル `cards.png` と `card_back.png` は以下の URL からダウンロードして下さい。

<http://www.sato-lab.jp/imfu/sampledata/cards.png>

[http://www.sato-lab.jp/imfu/sampledata/card\\_back.png](http://www.sato-lab.jp/imfu/sampledata/card_back.png)

サンプルプログラム
<pre> PImage frontCards;// 以下の 4 つの変数が宣言されている必要があります。 PImage backCard; int cardWidth = 73; // 読み込んでいる画像の大きさによって、 int cardHeight = 98; // この 2 つの値となっています。  void setup(){   size(400,400);   frontCards = loadImage("cards.png");   backCard = loadImage("card_back.png"); </pre>

```

    }

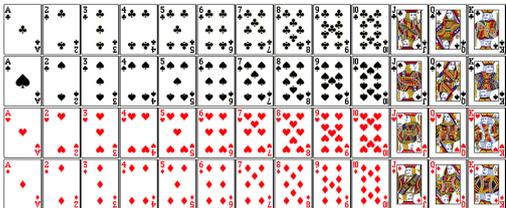
    void drawCard(int x,int y,int suit,int rank){
        if(0<=suit && suit<4 && 0 < rank && rank < 14){
            copy(frontCards,cardWidth*(rank-1),cardHeight*suit,
                cardWidth,cardHeight,x,y,cardWidth,cardHeight);
        }else if(rank == 0){
            image(backCard,x,y);
        }else{
            println("invalid parameters!!");
        }
    }

    void draw(){
        background(255);
        drawCard(mouseX,mouseY,2,1);
        drawCard(mouseX,mouseY+100,1,0);
    }
}

```

問 7~のプログラムでは問 6 で定義した drawCard 関数とこの関数を使用している 4 つの変数を利用している。

7. 下のプログラムは 52 枚の異なるトランプカードを全て表示するプログラムである。空欄を埋めてプログラムを完成させよ。

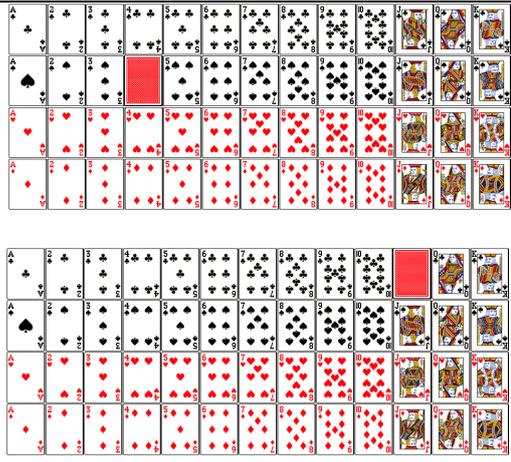
未完成のプログラム	実行例
<pre> PImage frontCards; PImage backCard; int cardWidth = 73; int cardHeight = 98;  void setup(){     size(949,392);     frontCards = loadImage("cards.png");     backCard = loadImage("card_back.png"); }  void drawCard(int x,int y,int suit,int rank){     if(0&lt;=suit &amp;&amp; suit&lt;4 &amp;&amp; 0 &lt; rank &amp;&amp; rank &lt; 14){         copy(frontCards,cardWidth*(rank-1),cardHeight*suit,             cardWidth,cardHeight,x,y,cardWidth,cardHeight);     }else if(rank == 0){         image(backCard,x,y);     }else{         println("invalid parameters!!");     } }  void draw(){     background(255);     for(int i=0;i&lt;4;i++){         for(int j=0;j&lt;13;j++){             drawCard( (a) , (b) , (c) , (d) );         }     } } </pre>	

```

}
}

```

8. 下のプログラムは問7のプログラムのように52枚の異なるトランプカードを表示するものであるが、マウスポインタが上に乗っているカードのみはカードの背面が表示されるようになっているプログラムである。空欄を埋めてプログラムを完成させよ。

未完成のプログラム	実行例
<pre> PImage frontCards; PImage backCard; int cardWidth = 73; int cardHeight = 98;  void setup(){   size(949,392);   frontCards = loadImage("cards.png");   backCard = loadImage("card_back.png"); } void drawCard(int x,int y,int suit,int rank){   if(0&lt;=suit &amp;&amp; suit&lt;4 &amp;&amp; 0 &lt; rank &amp;&amp; rank &lt; 14){     copy(frontCards,cardWidth*(rank-1),cardHeight*suit,       cardWidth,cardHeight,x,y,cardWidth,cardHeight);   }else if(rank == 0){     image(backCard,x,y);   }else{     println("invalid parameters!!");   } }  void draw(){   background(255);   for(int i=0;i&lt;4;i++){     for(int j=0;j&lt;13;j++){       drawCard( (a) , (b) , (c) , (d) );     }   }   int rank = (e);   int suit = (f);   drawCard( (g) , (h) , 0 , (i) ); } </pre>	

9. 下のプログラムは問8のプログラムの逆のような動作をするプログラムである。マスのカーソルが乗っている所のみカードの表面を表示し、それ以外は背面が表示されるようになっている。空欄を埋めてプログラムを完成させよ。面倒くさいので、問6で定義した drawCard 関数とこの関数を使用している4つの変数の宣言は除いている<sup>1</sup>。

未完成のプログラム	実行例
-----------	-----

<sup>1</sup>つまり、このプログラムを実行するためには、問6で定義した drawCard 関数とこの関数を使用している4つの変数の宣言を追加する必要がある。

<pre>//ここに追加 void setup(){   size(949,392);   frontCards = loadImage("cards.png");   backCard = loadImage("card_back.png"); }  void draw(){   background(255);   for(int i=0;i&lt;4;i++){     for(int j=0;j&lt;13;j++){       drawCard( <u>(a)</u>, <u>(b)</u>, 0, <u>(c)</u>);     }   }   int rank = <u>(d)</u>;   int suit = <u>(e)</u>;   drawCard( <u>(f)</u>, <u>(g)</u>, <u>(h)</u>, <u>(i)</u>); }</pre>	
--	--

10. 問6から作ってきたトランプカードを表示するプログラムにおいては、つぎのようになると、0から51までの52個の自然数の値と、52枚のトランプカードの間の一対一の対応をつけることができる<sup>2</sup>。

$$13 * \text{スーツの値} + (\text{カードのランク} - 1)$$

スーツの値は、クラブの時には0、スペードの時には1、ハートの時には2、ダイアの時には3とする。例えば、ダイアのエースには  $3 * 13 + (1 - 1) = 39$  という数値が対応し、クラブのキングには、  $0 * 13 + (13 - 1) = 12$  という数値が対応する。

下のプログラムは現在の時刻の秒に応じてトランプのカードを表示するものである。時刻の秒の値に対応するトランプのカードが無い場合（52秒から59秒の間）は、背面を表示することにする。空欄を埋めてプログラムを完成させて下さい。なお、空欄 a と b は1行とは限らない<sup>3</sup>。

プログラム中の getRank 関数は引数の値に対応するカードのランクを返し、getSuit 関数は引数の値に対応するカードのスーツの値を返すものである。面倒くさいので、問6で定義した drawCard 関数とこの関数を使用している4つの変数の宣言は除いている<sup>4</sup>。

<b>未完成のプログラム</b>
<pre>int getRank(int cardNo){   <u>(a)</u> }  int getSuit(int cardNo){   <u>(b)</u> }</pre>

<sup>2</sup> つまり、0から51の間の自然数を一つ決めると、それに対して一枚のカードが決まります。

<sup>3</sup> 1行でも書けます。

<sup>4</sup> もう、このプログラムを動かすためには、何が必要かはわかりますね。

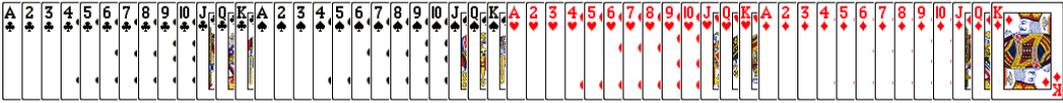
```

void setup(){
  size(300,300);
  frontCards = loadImage("cards.png");
  backCard = loadImage("card_back.png");
}

void draw(){
  background(255);
  int x = width/2-cardWidth/2;
  int y = height/2-cardHeight/2;
  int sec = second();
  int r = getRank(sec);
  int s = getSuit(sec);
  if(sec < 52){
    drawCard(_(c),_(d),_(e),_(f));
  }else{
    drawCard(_(c),_(d),_(g),_(h));
  }
}

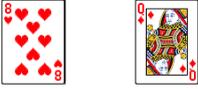
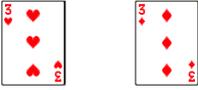
```

11. 下のプログラムは 52 枚の異なるトランプカードを全て表示するプログラムである。空欄を埋めてプログラムを完成させよ。なお、面倒くさいので問 10 で定義した 2 つの `getSuit` 関数と `getRank` 関数の定義は除いている。

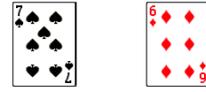
未完成のプログラム
<pre> void setup(){   size(1042,128);   frontCards = loadImage("cards.png");   backCard = loadImage("card_back.png"); } void draw(){   background(255);   int y= height/2-cardHeight/2;   for(int i=0;i&lt;52;i++){     int s = getSuit(_(a));     int r = getRank(_(a));     drawCard(i*19,y,_(b),_(c));   } } </pre>
実行例


12. 下のプログラムは 52 枚の異なるトランプカードの中から異なる 2 つのカードを選び、どちらのカードのランクの値が大きいかをあてるプログラムである。setup 関数内で異なる 2 枚のカードを選び (変数 `cardID0` と `cardID1` に 0~51 までの異なる 2 つの自然数の値を代入する)、最初はカードの裏面を表示する。並んだ 2 枚のカードの中でランクの値が大きいと思われるカードを、カーソルキーの左キーと右キーを使って指定す

る。つまり、大きいと思われるカードの向きのカーソルキーを押す。このとき、予測が当たれば”You Win”と表示し、反対側のカードのランクの値が大きければ”You Lose”と表示する。ただし、2枚のカードのランクの値が等しい場合には”Draw”を表示するものとする。カードの表面を表示するか背面を表示するかは boolean 型変数 openCards の値を利用して決める。openCards の値が true の時は表面を表示し、false の時はカードの背面を表示する。空欄を埋めてプログラムを完成させよ。空欄(b)は1行とは限らない。なお、面倒くさいので問10で定義した2つの getSuit 関数と getRank 関数の定義は除いている。

<pre> PFont font;  int cardID0; int cardID1; boolean openCards = false;  String msg;  void setup(){   size(300,400);   font = loadFont( (a) );   textFont(font,48);   frontCards = loadImage("cards.png");   backCard = loadImage("card_back.png");   // 空欄 b で cardID0 と cardID1 の値を決める   (b) }  void draw(){   background(255);   int x0 = width/4 - cardWidth/2;   int x1 = 3*width/4 - cardWidth/2;   int y = height/5;   if(openCards){     drawCard(x0,y, (b), (c));     drawCard(x1,y, (d), (e));     fill(0);     textAlign(CENTER,CENTER);     text(msg,0,width/2,width,height/2);   }else{     drawCard(x0,y, (f), (g));     drawCard(x1,y, (h), (i));   } }  void keyPressed(){   if(openCards    key != CODED){     return;   } </pre>	   <p><b>YOU LOSE</b></p>   <p><b>DRAW</b></p>
--	---

```
}
int rank0 = (j);
int rank1 = (k);
msg = "Draw";
if(keyCode == (l)){
    if(rank0 > rank1){
        msg = "You Win";
    }else if(rank0 < rank1){
        msg = "You Lose";
    }
}
openCards = (m);
}else if(keyCode == (n)){
    if(rank0 < rank1){
        msg = "You Win";
    }else if(rank0 > rank1){
        msg = "You Lose";
    }
}
openCards = (m);
}
}
```



**YOU WIN**

今週の宿題はありません。

2019 年度情報メディア基盤ユニット  
事前学習シート（7 月 9 日分）  
解答はキャリアポートフォリオで。

1. 次の URL にアクセスして指定されたビデオの視聴とアンケートに回答して下さい。学籍番号によって見ってもらうビデオが異なります<sup>5</sup>。最後に出てくるキーワードは授業内容とは無関係に設定されています。アンケート結果は授業用 e-learning 教材作成の際など参考にします。ビデオは youtube 上にあるので、家などからアクセスして下さい。



URL: <http://bit.ly/2J7L6o8>

2. 次の英単語の意味を調べてみよう。

maximum	
minimum	
tentative	
so far	

<sup>5</sup> 前回と同じグループを選んで下さい。Web ページにグループ分けの表をのせてあります。