

2015 年度情報メディア基盤ユニット 7 月 10 日分課題と宿題

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。問題中で使用している音ファイルは、<http://www.sato-lab.jp/imfu/sampledata> 中にあります。自己確認問題は先生や TA の人にチェックしてもらう必要はありません。後半にある【参考問題】は、演習時間中に解答出来なくてもかまいません。なお、問題は難易度順に並んでいるわけではありません。

1. 【自己確認問題】演習用資料のサンプルプログラム 11-1 を入力し、実行して見て下さい。もし何か音源ファイルをもっているようであれば、「player = minim.loadFile("schoolsong.mp3");」の読み込んでいる音源ファイルを別なファイルに変更して、プログラムを実行して見て下さい。
2. 【自己確認問題】演習用資料のサンプルプログラム 11-2 を入力し、実行して見て下さい。もし何か音源ファイルをもっているようであれば、「player = minim.loadFile("schoolsong.mp3");」の読み込んでいる音源ファイルを別なファイルに変更して、プログラムを実行して見て下さい。
3. 【自己確認問題】Processing の Examples 中にある Minim Audio 関連のサンプルを実行して見て下さい。
4. 【自己確認問題】Processing の Examples 中にある Minim Audio 関連のサンプルの中に、groove.mp3 という mp3 形式のファイルがあります。このファイルがどこにあるか見つけて下さい。
5. 次のプログラムは、groove.mp3 ファイルを読み込み、マウスをクリックしたら再生を行うものです。ただし、ファイルの最後まで再生が終了すれば、それで再生は終了となります。空欄を埋めて、プログラムを完成させて下さい。

未完成プログラム

```
import ddf.minim.*;
import ddf.minim.signals.*;
import ddf.minim.analysis.*;
import ddf.minim.effects.*;

  (a)  minim;
  (b)  player;

void setup(){
  size(100,100);
  minim = new Minim(this);
  player = minim.loadFile( (c) );
}

void draw(){
}
```

```

void mouseClicked(){
    player.__(d)__ ();
}
void stop(){
    player.__(e)__ ();
    minim.__(f)__ ();
    super.__(g)__ ();
}

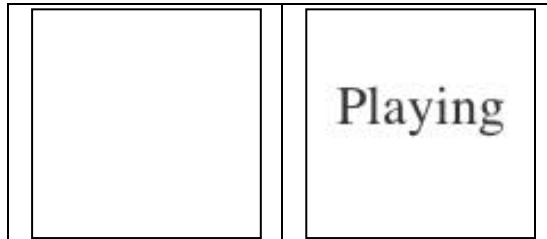
```

6. 次のプログラムは、groove.mp3 ファイルを読み込み、マウスをクリックしたら再生を行うものです。ただし、繰り返し再生を行うものとします。空欄を埋めて、プログラムを完成させて下さい。

未完成プログラム
<pre> import ddf.minim.*; import ddf.minim.signals.*; import ddf.minim.analysis.*; import ddf.minim.effects.*; __(a)__ minim; __(b)__ player; void setup(){ size(100,100); minim = new Minim(this); player = minim.loadFile(__(c)__); } void draw(){ } void mouseClicked(){ player.__(d)__ (); } void stop(){ player.__(e)__ (); minim.__(f)__ (); super.__(g)__ (); } </pre>

7. 【自己確認問題】 演習用資料のサンプルプログラム 11-3 と 11-4 を入力し、実行して見て下さい。もし何か音源ファイルをもっているようであれば、「player = minim.loadFile("schoolsong.mp3");」の読み込んでいる音源ファイルを別なファイルに変更して、プログラムを実行して見て下さい。
8. 【自己確認問題】 演習用資料サンプルプログラム 10-5 と 10-6 を入力し、実行して見て下さい。
9. 【目コピ問題】 次のプログラムは、groove.mp3 ファイルを読み込み、マウスをクリックした

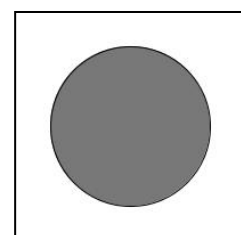
ら、必ずファイルの先頭から再生を行うものです。また、再生中には「Playing」との表示が
 できるようになっています。空欄を埋めて、プログラムを完成させて下さい。



未完成プログラム
<pre> import ddf.minim.*; import ddf.minim.signals.*; import ddf.minim.analysis.*; import ddf.minim.effects.*; (a) minim; (b) player; PFont font; void setup(){ size(100,100); minim = new Minim(this); player = minim.loadFile("groove.mp3"); font = loadFont((c)); textFont(font,24); textAlign((d)); } void draw(){ background(255); if(player. (e) ()){ fill(50); text("Playing",width/2,height/2); } } void mouseClicked(){ player. (f) (); player. (g) (); } void stop(){ player. (h) (); minim. (i) (); super. (j) (); } </pre>

10. 【目コピ問題】 次のプログラムは、円の部分をクリックすると、
 score.wav を再生するものです。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> import ddf.minim.*; import ddf.minim.signals.*; </pre>

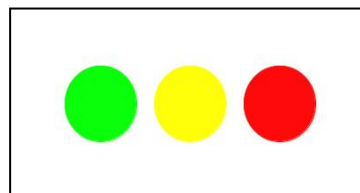


```

import ddf.minim.analysis.*;
import ddf.minim.effects.*;
Minim minim;
AudioSample sSound;
void setup(){
  minim = new Minim(this);
  sSound = __ (a) __("score.wav");
  size(200,200);
  smooth();
}
void draw(){
  background(255);
  stroke(0);
  fill(120);
  ellipse(width/2,height/2,__ (b) __, __ (b) __);
}
void mouseClicked(){
  if(dist(width/2,height/2,mouseX,mouseY) < __ (c) __){
    sSound.__ (d) __();
  }
}
void stop(){
  __ (e) __
  minim.stop();
  super.stop();
}

```

11. 【目コピ問題】 次のプログラムは、円の部分をクリックすると、音声ファイルを再生するものです。左の円をクリックした場合には score.wav を再生し、真ん中の円をクリックした場合には damage.wav を再生し、右の円をクリックした場合には totaled.wav を再生するものです。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

```

import ddf.minim.*;
import ddf.minim.signals.*;
import ddf.minim.analysis.*;
import ddf.minim.effects.*;

Minim minim;
AudioSample[] se;
int radiusBall = 40;
int[] xBall;
color[] cBall;
void setup(){
  minim = new Minim(this);
  size(400,200);
  smooth();
}

```

```

se = __ (a) __;
se[0] = minim.loadSample( __ (b) __);
se[1] = minim.loadSample( __ (c) __);
se[2] = minim.loadSample( __ (d) __);
xBall = __ (e) __;
for(int i=0;i<3;i++){
    xBall[i] = __ (f) __;
}
cBall = __ (g) __;
cBall[0] = color(10,255,10);
cBall[1] = color(255,255,10);
cBall[2] = color(255,10,10);
}

void draw(){
    background(255);
    for(int i=0;i<3;i++){
        stroke( __ (h) __);
        fill( __ (i) __);
        ellipse(xBall[i],height/2,2*radiusBall,2*radiusBall);
    }
}

void mouseClicked(){
    for(int i=0;i<3;i++){
        if(dist( __ (j) __, __ (k) __,mouseX,mouseY) < __ (l) __){
            se[i]. __ (m) __;
            break;
        }
    }
}

void stop(){
    for(int i=0;i<3;i++){
        __ (n) __;
    }
    minim.stop();
    super.stop();
}

```

12. 【自己確認問題】 演習用資料サンプルプログラム 11-10 を入力し、実行して見て下さい。

「sine = new SineWave(440, 0.5, out.sampleRate());」の 440 が周波数を決めています。そこで、この 440 という数字を別な数字に置き換えて、プログラムを実行して見て下さい。

13. 【自己確認問題】 演習用資料サンプルプログラム 11-11～11-13 を入力し、実行して見て下さい。

「sine = new SineWave(440, 0.5, out.sampleRate());」などの 440 の部分を別な数字に置き換えて、プログラムを実行して見て下さい。

14. 【自己確認問題】 演習用資料サンプルプログラム 11-14 を入力し、実行して見て下さい。
15. 【目コピ問題】 次のプログラムは、周波数 880Hz の正弦波を鳴らプログラムです。空欄を埋めて、プログラムを完成させて下さい。

未完成プログラム
<pre>import ddf.minim.*; import ddf.minim.signals.*; import ddf.minim.analysis.*; import ddf.minim.effects.*; Minim minim; AudioOutput out; (a) sine; void setup(){ minim = new Minim(this); out = minim. (b) (Minim.STEREO); sine = new (c) ((d) , (e) , (f)); out.addSignal(sine); } void draw(){ } void stop(){ (g) .close(); // ライン出力の機能を終了する (h) .stop(); // Minimの機能を停止する (i) .stop(); // 停止の際のおまじない } }</pre>

16. 【目コピ問題】 AudioOutput 型の変数 out (例えば) に対して、out.mute()を実行すると、音が止まります。逆に、out.unmute()を実行すると、音が鳴り始めます。次のプログラムは、このことを利用して、マウスボタンを押している時だけ、音が鳴るようになっています。周波数 440Hz の正弦波を鳴らプログラムです。空欄を埋めて、プログラムを完成させて下さい。つまり、マウスボタンが押された out.unmute()を実行し、マウスボタンが離されたら out.mute()を実行しています。

このように音の再生を制御するのが一番良い方法かどうかはわかりませんが、何とか要件を満たすことが出来ます。ただ、この方法だと最後のノイズがのって、ちょっと気持ち悪いですね。

未完成プログラム
<pre>import ddf.minim.*; import ddf.minim.signals.*; import ddf.minim.analysis.*; import ddf.minim.effects.*; Minim minim; AudioOutput out; SineWave sine; void setup(){ minim = new Minim(this); out = minim. (a) (Minim.STEREO);</pre>

```

sine = new ___(b)___ ( ___(c)___ , ___(d)___ , ___(e)___ );
out.addSignal(sine);
out.___(f)___(); // はじめから音が鳴っては困りますよね。
}
void draw(){
}
void ___(g)___(){
out.unmute();
}
void ___(h)___(){
out.mute();
}
void stop(){
___(i)___ .close(); // ライン出力の機能を終了する
___(j)___ .stop(); // Minimの機能を停止する
___(k)___ .stop(); // 停止の際のおまじない
}
}

```

17. 【目コピ問題】 次のプログラムは、マウスボタンを押している時だけ、音が鳴るようになっています。周波数 440Hz の正弦波を鳴らプログラムです。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、setFreq メソッドを使って、周波数を 0Hz に設定することで、音の発生を止めています。

未完成プログラム

```

import ddf.minim.*;
import ddf.minim.signals.*;
import ddf.minim.analysis.*;
import ddf.minim.effects.*;
Minim minim;
AudioOutput out;
SineWave sine;
void setup(){
minim = new Minim(this);
out = minim.___(a)___ (Minim.STEREO);
sine = new ___(b)___ ( ___(c)___ , ___(d)___ , ___(e)___ );
out.addSignal(sine);
sine.___(f)___ (0); // はじめから音が鳴っては困りますよね。
}
void draw(){
}
void ___(g)___(){
sine.___(h)___ (440);
}
void ___(i)___(){
sine.___(j)___ (0);
}
void stop(){
___(k)___ .close(); // ライン出力の機能を終了する
___(l)___ .stop(); // Minimの機能を停止する
super.stop(); // 停止の際のおまじない
}
}

```

18. 【目コピ問題】 次のプログラムは、マウスボタンをクリックすると、0.1 秒間だけ 880Hz の正弦波を発生させるプログラムです。空欄を埋めて、プログラムを完成させて下さい。空欄(f)

は存在しません。

未完成プログラム
<pre> import ddf.minim.*; import ddf.minim.signals.*; import ddf.minim.analysis.*; import ddf.minim.effects.*; Minim minim; AudioOutput out; SineWave sine; int ringingStart void setup(){ minim = new Minim(this); out = minim.__(a)__(Minim.STEREO); sine = new __(b)__(__(c)__, __(d)__, __(e)__); out.addSignal(sine); __(g)__; // はじめから音が鳴っては困りますよね。 } void draw(){ if(millis()-ringingStart > __(h)++){ __(i)__; } } void mouseClicked(){ ringingStart = millis(); __(j)__; } void stop(){ __(k)__.close(); // ライン出力の機能を終了する __(l)__.stop(); // Minimの機能を停止する super.stop(); // 停止の際のおまじない } </pre>

19. 【目コピ問題】 次のプログラムは、1 のキーを押すと 440Hz の正弦波を発生させ、2 のキーを押すと 880Hz の音を発生させるプログラムです。空欄を埋めて、プログラムを完成させて下さい。空欄(f)は存在しません。

未完成プログラム
<pre> import ddf.minim.*; import ddf.minim.signals.*; import ddf.minim.analysis.*; import ddf.minim.effects.*; Minim minim; AudioOutput out; SineWave sine; void setup(){ minim = new Minim(this); out = minim.__(a)__(Minim.STEREO); //最初から音が鳴っては困りますよね。 sine = new __(b)__(__(c)__, __(e)__, __(e)__); out.addSignal(sine); } void draw(){ } void keyPressed(){ if(key == '1'){ </pre>


```

    sine.setFreq(__(g)__);
}else if(key == '2'){
    sine.setFreq(__(h)__);
}
}
void keyReleased(){
    sine.setFreq(__(i)__);
}
void stop(){
    __(j)__.close(); // ライン出力の機能を終了する
    __(k)__.stop(); // Minimの機能を停止する
    __(l)__.stop(); // 停止の際のおまじない
}

```

20. 【目コピ問題】 次のプログラムは、円が左右に動くものです。端に来ると、反対側に移動を始めます。このプログラムでは、端に来るたびに、何かの音を鳴らし、さらに色を変化させています。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
<pre> import ddf.minim.*; import ddf.minim.signals.*; import ddf.minim.analysis.*; import ddf.minim.effects.*; Minim minim; AudioSample se; int xBall; int vBall; int rBall=15; color cBall; void setup(){ minim = new Minim(this); se = minim.loadSample(__(a)__); size(400,200); smooth(); colorMode(HSB,359,99,99); xBall=width/2; vBall = 1; cBall = color(random(360),99,99); } void draw(){ background(0,0,99); xBall += vBall; if(__(b)){ xBall = __(c) ; vBall = 1; se.__(d) (); cBall = color(random(360),99,99); }else if(__(e)){ xBall = __(f) ; vBall = -1; </pre>

```

    se. (g) ();
    cBall = color(random(360),99,99);
  }
  stroke(cBall);
  fill(cBall);
  ellipse(xBall,height/2,2*rBall,2*rBall);
}
void stop(){
  (h) .close(); // ライン出力の機能を終了する
  (i) .stop(); // Minimの機能を停止する
  (j) .stop(); // 停止の際のおまじない
}

```

21. 【工夫問題】 a～m のキーを押している間、次の周波数の正弦波を発生させるようなプログラムを作成してください。キーの配置などは変更してもかまいません。

キー	周波数	キー	周波数	キー	周波数
a	440.000000	f	587.329536	k	783.990872
b	466.163762	g	622.253967	l	830.609395
c	493.883301	h	659.255114	m	880.000000
d	523.251131	i	698.456463		
e	554.36522	j	739.988845		

22. 【工夫問題】 時刻が 57 秒と 58 秒と 59 秒の時に少しの間 440Hz の正弦波を鳴らし、その後 0 秒の時に 880Hz の正弦波を少しの間鳴らすようなプログラムを作って下さい。可能なら時計仕立てに見かけにしてください。

23. 【工夫問題】 過去の演習問題中にあったゲームのようなプログラムにおいて、弾を発射する時や円（敵）と弾の衝突時に異なった効果音を鳴らすような改良を付け加えて下さい。

24. 【目コピ問題、おまけ】 AudioOutput 型の変数 out（例えば）に対して、out.noSound()を実行すると、音が止まります。逆に、out.sound()を実行すると、音が鳴り始めます。次のプログラムは、マウスボタンを押している時だけ、音が鳴るようになっています。周波数 440Hz の正弦波を鳴らすプログラムです。空欄を埋めて、プログラムを完成させて下さい。

課題中で紹介した中で、この方法が一番正解に近い気がします。

未完成プログラム	
import ddf.minim.*;	
import ddf.minim.signals.*;	
import ddf.minim.analysis.*;	
import ddf.minim.effects.*;	
Minim minim;	
AudioOutput out;	
SineWave sine;	
void setup(){	
minim = new Minim(this);	
out = minim. (a) (Minim.STEREO);	

```

sine = new (b) ( (c) , (d) , (e) );
out.addSignal(sine);
out.(f) (); // はじめから音が鳴っては困りますよね。
}
void draw(){
}
void (g) (){
out.sound();
}
void (h) (){
out.noSound();
}
void stop(){
(i) .close(); // ライン出力の機能を終了する
(j) .stop(); // Minimの機能を停止する
(k) .stop(); // 停止の際のおまじない
}

```

25. 【目コピ問題、参考問題】 数字を表示する際に、表示桁数を指定して表示を行いたい場合があります。その目的のために、Processing では nf 関数があります。この関数は、数値を指定された桁数の文字列に変換するものです。123 という値を 5 桁の文字列に変換したいときには、nf(123,5)とすると、“00123”という 5 文字の文字列となります。なお、通常この関数には、非負の数を対象としています。

この関数を使ったものがサンプル 11-1 です。このサンプルは、マウスをクリックするとストップウォッチのように経過時間を計るものです。

サンプルプログラム 11-1	実行例
<pre> PFont font; int startTime_msec; boolean counting = false; void setup(){ size(400,200); smooth(); //font は各自のものに変更してください。 font = loadFont("Serif-48.vlw"); textFont(font,48); textAlign(CENTER); fill(0); } void mouseClicked(){ counting = true; startTime_msec = millis(); } void draw(){ int t=0; background(255); if(counting){ t = millis(); } int ms = t % 1000; int s = t/1000; int m = s/60; String elapsed=nf(m,4)+":"+nf(s%60,2)+":"+nf(ms,3); text(elapsed,width/2,height/2); </pre>	<div data-bbox="1003 1205 1337 1361" style="border: 1px solid black; padding: 10px; text-align: center;">0000:00:000</div> <div data-bbox="1003 1406 1310 1563" style="border: 1px solid black; padding: 10px; text-align: center;">0001:53:063</div>

}	
---	--

この nf 関数を使って、14 時 41 分 09 秒であれば、「14:41:09」と表示されるなプログラムを作成したい。下の未完成のプログラムの空欄を埋めて、プログラムを完成させてください。

未完成プログラム	実行例
<pre> PFont font; void setup(){ size(400,200); smooth(); font = __ (a) __; textFont(font,48); } void draw(){ background(255); int h = hour(); int m = minute(); int s = second(); String time = __ (b) __ + ":" + __ (c) __ + ":" + __ (d) __; fill(0); textAlign(CENTER); text(time,width/2,height/2); } </pre>	<div style="border: 1px solid black; width: 100px; height: 100px; margin: auto; display: flex; align-items: center; justify-content: center;"> 14:41:09 </div>

nf 関数の使用方法

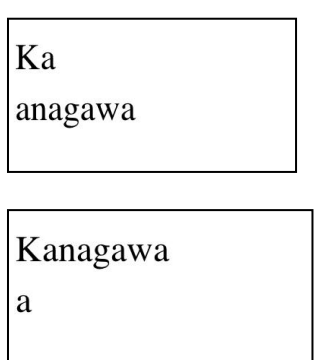
関数の呼び出し	呼び出し例	戻り値
nf(整数値,桁数)	nf(123,5)	“00123”
nf(実数値,整数部分桁数,小数部分桁数)	nf(3.1415,3,5)	“003.14150”

26. 【目コピ問題、参考問題】 String 型(String クラス)は、いつかの便利にメソッドを持っています。下の呼び出し例では、「String msg=”Makise Riho” ;」となっています。

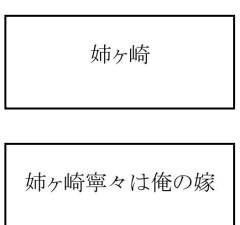
メソッド	機能	呼び出し例	戻り値
length()	文字列の長さを求める。	msg.length()	11
substring(start)	Start 番目から最後まで 文字取り出します。配列と同じように 先頭の文字が 0 番目です。	msg.substring(3);	“ise Riho”
substring(start,pos)	Start 番目から pos-1 番目のまでの文字列 を取り出します。	msg.substring(2,4)	“ki”
		msg.substring(0,3)	“Mak”

これらのメソッドを利用すると、次のようなプログラムを作成することが出来ます。このプログラムは、0.5 秒ごとに表示される文字数が増えたり減ったりするというものです。

サンプルプログラム 13-2	実行例
----------------	-----

<pre>String msg = "Kanagawa"; PFont font; void setup(){ size(400,200); //font は各自のものに変更してください。 font = loadFont("Serif-48.vlw"); textFont(font,48); fill(0); } void draw(){ background(255); fill(0); int pos = ((millis()/1000) % msg.length()); text(msg.substring(0,pos+1),10,height/3); text(msg.substring(pos),10,2*height/3); }</pre>	
--	--

次のプログラムはマウスの X 座標(mouseX)値を利用して、文字列の表示範囲を決めるものです。マウスが左端にいるときは無表示、右端にいるときには全てを表示となっています。空欄を埋めて、プログラムを完成させてください。

未完成プログラム	実行例
<pre>//msg は各自のものに変更してください。 String msg = "姉ヶ崎寧々は俺の嫁"; PFont font; void setup(){ size(500,200); //font は各自のものに変更してください。 font = loadFont("MS-PMincho-48.vlw"); textFont(font,48); textAlign(CENTER); fill(0); } void draw(){ background(255); fill(0); float dx = width/msg.length(); int pos = round(mouseX/dx); text(msg.__(a)__(b)__,__(c)__, width/2,height/2); }</pre>	

日本語フォントを指定すると、日本語の表示を行うことが出来るようになります。また、直接日本語の入力が出来ない（特に、Processing1.5 系列）、メモ帳などで入力した文字列をコピー&ペーストで貼り付けることで、日本語の文字の入力ができます。

27. 【工夫問題、参考問題】 次のプログラムは、

- 1) 最初は開始画面が表示する。
- 2) 表示されている正方形をクリックしたら、Hit 表示がされる。

3) Hit 表示は少し経ったら(0.5 秒)、再び正方形を表示する。

4) 正方形は 1 秒間表示されたら、別な場所に移動する。

といった感じのゲームのようなプログラムです。

サンプルプログラム 13-3

```

PFont font;
float xTarget;
float yTarget;
int targetWidth;
int targetHeight;
color targetColor;

int GAME_READY = 0;
int GAME_UPDATING = 1;
int GAME_RUNNING = 2;
int GAME_HIT = 3;

int currentState;
int time0_msec;

void startElapsedTime(){
  time0_msec = millis();
}

int elapsedTime_msec(){
  return millis()-time0_msec;
}

void setup(){
  size(400,400);
  smooth();
  font = createFont("Serif",48);
  currentState = GAME_READY;
}

void updateTarget(){
  targetWidth = 50;
  targetHeight = 50;
  xTarget = random(targetWidth,
    width-targetWidth);
  yTarget = random(targetHeight,
    height-targetHeight);
  targetColor = color(10,10,255);
}

boolean isOnTarget(int x,int y){
  if((xTarget <= x &&
    x < (xTarget+targetWidth)) &&
    (yTarget <= y &&
    y < (yTarget+targetHeight))){
    return true;
  }else{
    return false;
  }
}

void showReadyMessage(){
  textAlign(CENTER);
  textFont(font,48);
  fill(0);
  text("Start to click",width/2,height/2);
}

void showHitMessage(){
  textAlign(CENTER);
  textFont(font,48);
  fill(255,10,10);
  text("Hit!!",width/2,height/2);
}

void showTarget(){
  stroke(targetColor);
  fill(targetColor);
  rectMode(CORNER);
  rect(xTarget,yTarget,
    targetWidth,targetHeight);
}

void draw(){
  background(255);
  if(currentState == GAME_READY){
    showReadyMessage();
  }else if(currentState == GAME_UPDATING){
    updateTarget();
    currentState = GAME_RUNNING;
    startElapsedTime();
  }else if(currentState == GAME_RUNNING){
    showTarget();
    if(elapsedTime_msec() >= 1000){
      currentState = GAME_UPDATING;
    }
  }else if(currentState == GAME_HIT){
    showHitMessage();
    if(elapsedTime_msec() >= 500){
      currentState = GAME_UPDATING;
    }
  }
}

void mouseClicked(){
  if(currentState == GAME_READY){
    currentState = GAME_UPDATING;
  }else if(currentState == GAME_RUNNING){
    if(isOnTarget(mouseX,mouseY)){
      currentState = GAME_HIT;
      startElapsedTime();
    }
  }
}

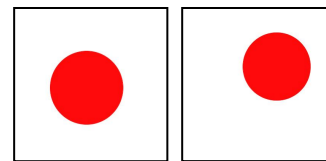
```

このプログラムに以下のような改良を加えて下さい。

- (1) 正方形の表示時間を乱数で変更するようにして下さい。
- (2) Hit 回数を表示するようにして下さい。
- (3) 5 回ミス hit したら、ゲームが終了するようにして下さい。
- (4) hit 回数に応じて、正方形の表示時間が短くなるようにして下さい。
- (5) 単純な正方形ではなく、画像や別な形となるようにして下さい。
- (6) 効果音を付け加えて下さい。
- (7) 繰り返しゲームが出来るようにし、開始画面にそれまでの最大 hit 回数が表示されるようにして下さい。

宿題

1. 次のプログラムは、マウスをクリックすると schoolsong.mp3 の再生を始めます。音の再生が行われている間は、円がランダムに移動し、少し半径を変更しています。ただし、円がウインドウの外に出て行ってしまったり、円が小さくなりすぎたり、大きくなるすぎると困るので、これらの値はある範囲からずれないようにしています。なお、再度マウスをクリックした場合には、最初から再生を開始するものとします。なお、空欄 (d) には、下の表のうちのどれかが入ります。解答はポートフォリオで！！



constrain	lerp	map	min	max
-----------	------	-----	-----	-----

未完成のプログラム

```
import ddf.minim.*;
import ddf.minim.signals.*;
import ddf.minim.analysis.*;
import ddf.minim.effects.*;
Minim minim;
AudioPlayer player;
float xBall,yBall,rBall;;
void setup(){
  minim = new Minim(this);
  player = __ (a) __ .loadFile( __ (b) __ );
  size(400,400);
  smooth();
  xBall = width/2;
  yBall = height/2;
  rBall = width/4;
}
void draw(){
  background(255);
  if(player.__ (c) __==true){
    xBall = __ (d) __ (xBall+random(-5,5),50,width-50);
    yBall = __ (d) __ (yBall+random(-5,5),50,height-50);
    rBall = __ (d) __ (rBall+random(-1,1),3,0.5*width);
  }
  stroke(255,10,10);
  fill(255,10,10);
  ellipse(xBall,yBall,2*rBall,2*rBall);
}
void mouseClicked(){
  player.__ (e) __ ();
  player.__ (f) __ ();
}
void stop(){
  __ (g) __ ();
  __ (h) __ ();
  __ (i) __ ();
}
```