

## 2018 年度情報メディア基盤ユニット 7 月 6 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> の中にあります。【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。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。問2~12,16はキャリアポートフォリオにも解答をアップして下さい。また、問13~15、17はプログラムをアップロードして下さい。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、11-2、11-3、11-4、10-5、10-6、11-10、11-14 を入力し、実行して見て下さい。もし何かサウンドファイルをもっているようであれば、それを利用してかまいません。
2. Processing の“サンプル…”の中にある Minim 関連のサンプルの中には、groove.mp3 という mp3 形式のファイルを利用しているものが沢山あります。次のプログラムは、このファイル groove.mp3 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。

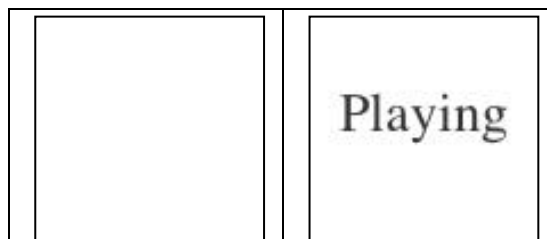
未完成プログラム
<pre>import ddf.minim.*; import ddf.minim.analysis.*; import ddf.minim.effects.*; import ddf.minim.signals.*; import ddf.minim.spi.*; import ddf.minim.ugens.*; __ (a) __ minim; __ (b) __ player; void setup(){   size(100,100);   minim = new Minim(this);   player = minim.loadFile(__ (c) __); }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

3. 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.analysis.*; import ddf.minim.effects.*; import ddf.minim.signals.*; import ddf.minim.spi.*; import ddf.minim.ugens.*; __ (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>

4. 【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。



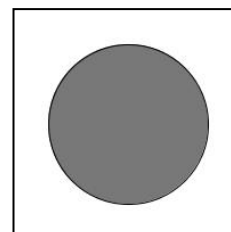
未完成プログラム
<pre> import ddf.minim.*; import ddf.minim.analysis.*; import ddf.minim.effects.*; import ddf.minim.signals.*; import ddf.minim.spi.*; import ddf.minim.ugens.*; __ (a) __ minim; __ (b) __ player; PFont font; void setup(){   size(100,100);   minim = new Minim(this);   player = minim.loadFile("groove.mp3");   font = loadFont(__ (c) __);   textFont(font,24);   textAlign(__ (d) __); } </pre>

```

}
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)();
}
}

```

5. 【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。



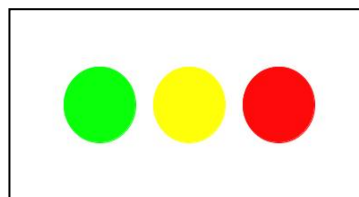
#### 未完成のプログラム

```

import ddf.minim.*;
import ddf.minim.analysis.*;
import ddf.minim.effects.*;
import ddf.minim.signals.*;
import ddf.minim.spi.*;
import ddf.minim.ugens.*;
Minim minim;
AudioSample sSound;
void setup(){
  minim = new Minim(this);
  sSound = __(a)__("score.wav");
  size(200,200);
}
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();
}
}

```

6. 【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	
<pre> import ddf.minim.*; import ddf.minim.analysis.*; import ddf.minim.effects.*; import ddf.minim.signals.*; import ddf.minim.spi.*; import ddf.minim.ugens.*; Minim minim; AudioSample[] se; int radiusBall = 40; int[] xBall; color[] cBall; void setup(){   minim = new Minim(this);   size(400,200);   se = __ (a) __;   se[0] = minim.loadSample(__ (b) __);   se[1] = minim.loadSample(__ (c) __);   se[2] = minim.loadSample(__ (d) __);   xBall = __ (e) __;   for(int i=0;i&lt;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&lt;3;i++){     stroke(__ (h) __);     fill(__ (i) __);     ellipse(xBall[i],height/2,2*radiusBall,2*radiusBall);   } } void mouseClicked(){   for(int i=0;i&lt;3;i++){     if(dist(__ (j) __, __ (k) __,mouseX,mouseY) &lt; __ (l) __){       se[i].__ (m) __;       break;     }   } } void stop(){   for(int i=0;i&lt;3;i++){ </pre>	

```

    ____(n)__;
}
minim.stop();
super.stop();
}

```

7. 【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

```

import ddf.minim.*;
import ddf.minim.analysis.*;
import ddf.minim.effects.*;
import ddf.minim.signals.*;
import ddf.minim.spi.*;
import ddf.minim.ugens.*;
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
}

```

8. 【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できます。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

```

import ddf.minim.*;
import ddf.minim.analysis.*;
import ddf.minim.effects.*;
import ddf.minim.signals.*;
import ddf.minim.spi.*;
import ddf.minim.ugens.*;
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.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
}

```

9. 【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、setAmp メソッドを使って、振幅を 0 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.analysis.*;
import ddf.minim.effects.*;
import ddf.minim.signals.*;
import ddf.minim.spi.*;
import ddf.minim.ugens.*;
Minim minim;
AudioOutput out;
SineWave sine;
void setup(){
  minim = new Minim(this);
  out = minim.__(a)__(Minim.STEREO);
  // はじめから音が鳴っては困りますよね。
  sine = new __(b)__(__(c), __(d), __(e));
  out.addSignal(sine);
}
void draw(){
}
void __(g)(){
  sine.__(h)__(1);
}
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
}

```

10. 【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.analysis.*; import ddf.minim.effects.*; import ddf.minim.signals.*; import ddf.minim.spi.*; import ddf.minim.ugens.*; 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 &gt; __(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>

11. 【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。

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

```

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

```

12. 【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.analysis.*; import ddf.minim.effects.*; import ddf.minim.signals.*; import ddf.minim.spi.*; import ddf.minim.ugens.*; Minim minim; AudioSample se; float xBall; float vBall; int rBall=15; color cBall; void setup(){     minim = new Minim(this);     se = minim.loadSample(__(a)__);     size(400,200);     colorMode(HSB,359,99,99);     xBall=width/2;     vBall = 1;     cBall = color(random(360),99,99); } void draw(){     background(0,0,99);     xBall += vBall; </pre>



```

if(__(b)__){
  xBall = __(c__);
  vBall = random(1,3);
  se.__(d)__();
  cBall = color(random(360),99,99);
}else if(__(e)__){
  xBall = __(f__);
  vBall = -random(1,3);
  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
}

```

13. 【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		

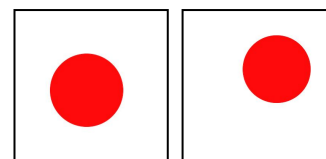
14. 【工夫問題】 時刻が 57 秒と 58 秒と 59 秒の時に少しの間（可能なら 0.1 秒） 440Hz の正弦波を鳴らし、その後 0 秒の時に 880Hz の正弦波を少しの間鳴らす（可能なら段々音を小さくしながら 2 秒間） ようなプログラムを作って下さい。可能なら時計仕立てに見かけにしてください。
15. 【工夫問題】 過去の演習問題中にあったゲームのようなプログラムにおいて、弾を発射する時や円（敵）と弾の衝突時に異なった効果音を鳴らすような改良を付け加えて下さい。
16. 【目コピ問題、おまけ】 AudioOutput型の変数out（例えば）に対して、out.disableSignal(シグナル番号)を実行すると、指定したシグナル番号の音が止まります。逆に、out.enableSignal(シグナル番号)を実行すると、指定したシグナル番号の音が鳴り始めます。シグナル番号は、out.addSigna関数を呼び出した順に0から数えて行きます。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.analysis.*; import ddf.minim.effects.*; import ddf.minim.signals.*; import ddf.minim.spi.*; import ddf.minim.ugens.*; 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)__(0);// はじめから音が鳴っては困りますよね。 } void draw(){ } void __(g)__() {   out.disableSignal(0); } void __(h)__() {   out.enableSignal(0); } 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 } </pre>

17. 【おまけ問題】七夕ですので、星のような形を表示するプログラムを作ってください。サウンドの回ですので、BGMは必修です。

### 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、円がウィ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
<pre>import ddf.minim.*;</pre>

```

import ddf.minim.analysis.*;
import ddf.minim.effects.*;
import ddf.minim.signals.*;
import ddf.minim.spi.*;
import ddf.minim.ugens.*;
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) __ ();
}
}

```

## 7月10日授業用事前学習課題

キャリアポートフォリオで解答を記入して下さい。

問1. 内分点の計算 (2つの点を結ぶ線分を指定した長さの割合で分割する点) の仕方を調べて下さい。

問2. 次のような“早撃ち”ゲームのプログラムはどのようにプログラムされているかを考えてみて下さい。“早撃ち”ゲームのおおよその内容は以下の通りです。

- 1) 最初は開始画面が表示する。
- 2) 表示されている正方形 (ターゲット) 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 (ターゲット) を表示する。
- 4) 正方形 (ターゲット) は1秒間表示されたら、別な場所に移動する。