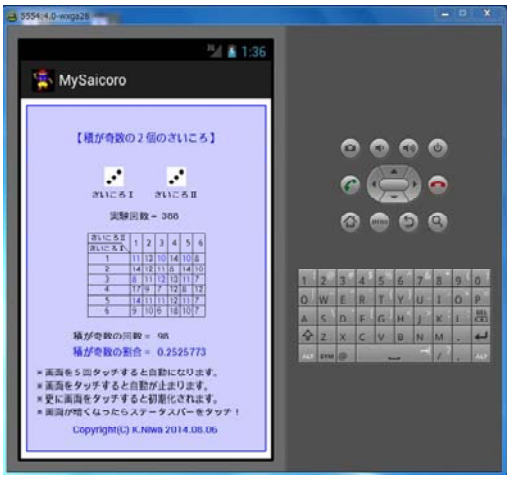
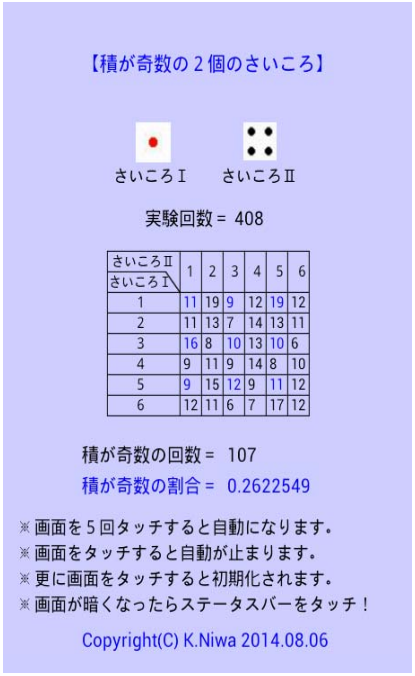
```
//--------------------------------------
//
//        積が奇数の２個のさいころ
//                Ver6
//    Copyright(C) K.Niwa 2014.08.06
//
//--------------------------------------
```



【エミュレータ画面例】



さいころⅡ / さいころⅠ	1	2	3	4	5	6
1	11	19	9	12	19	12
2	11	13	7	14	13	11
3	16	8	10	13	10	6
4	9	11	9	14	8	10
5	9	15	12	9	11	12
6	12	11	6	7	17	12

【スマートフォン画面例】
（Android 4.0）

［アプリの概要］

２個のさいころを同時に投げたときの目の出方が観察できます。

２個のさいころの目の積が奇数になる割合はどれだけでしょうか。

２個のさいころを同時に投げる実験を何度もおこなった場合、実験回数を多くすると、２個のさいころの目の積が奇数になる割合は、実験回数が少ないときに比べてどのようなことが言えるのでしょうか。

また、２個のさいころの目の積が奇数になる確率を数学的に求めてみましょう。

［１］Saicoro.java

```
package jp.seitoku.saicoro;

import android.content.Context;
import android.content.res.Resources;      //画像用
import android.graphics.*;
import android.util.AttributeSet;
import android.view.*;    //view.View → view.* に変更

public class Saicoro extends View {

        private Bitmap bitmap1 = null;
        private Bitmap bitmap2 = null;
        private Bitmap bitmap3 = null;
        private Bitmap bitmap4 = null;
        private Bitmap bitmap5 = null;
        private Bitmap bitmap6 = null;

        int flag=0;                                                    //自動識別子
        int syoki=0;                                                   //初期化識別子
        int r1,r2;                                                     //さいころ１、
さいころ２の目の識別子（乱数）
        int ct1=0;                                                     //実験回数
        int d11=0,d21=0,d31=0,d41=0,d51=0,d61=0;   //度数 d31とはさいころ１の目が３で、さいころ２の目が１
```

```
の度数
        int d12=0,d22=0,d32=0,d42=0,d52=0,d62=0;
        int d13=0,d23=0,d33=0,d43=0,d53=0,d63=0;
        int d14=0,d24=0,d34=0,d44=0,d54=0,d64=0;
        int d15=0,d25=0,d35=0,d45=0,d55=0,d65=0;
        int d16=0,d26=0,d36=0,d46=0,d56=0,d66=0;
        int d=0;                                                    //積が奇数の度数
        float ritu;                                                 //積が奇数の
割合
        int yy,xx;                                                  //枠に使用し
たループカウンター

        int width;
        int height;

        public Saicoro(Context context, AttributeSet attrs, int defStyle) {
                super(context, attrs, defStyle);
                // TODO 自動生成されたコンストラクター・スタブ
                init(context);
        }

        public Saicoro(Context context, AttributeSet attrs) {
                super(context, attrs);
                // TODO 自動生成されたコンストラクター・スタブ
                init(context);
        }

        public Saicoro(Context context) {
                super(context);
                // TODO 自動生成されたコンストラクター・スタブ
                init(context);
        }

        private void init(Context context) {
                Resources res = context.getResources();
                bitmap1 = BitmapFactory.decodeResource(res, R.drawable.sai1);
                bitmap2 = BitmapFactory.decodeResource(res, R.drawable.sai2);
                bitmap3 = BitmapFactory.decodeResource(res, R.drawable.sai3);
                bitmap4 = BitmapFactory.decodeResource(res, R.drawable.sai4);
                bitmap5 = BitmapFactory.decodeResource(res, R.drawable.sai5);
                bitmap6 = BitmapFactory.decodeResource(res, R.drawable.sai6);

                WindowManager wm = (WindowManager)context.getSystemService(Context.WINDOW_SERVICE);
                Display disp = wm.getDefaultDisplay();
                width = disp.getWidth();
                height = disp.getHeight();
        }

        @Override
        protected void onDraw(Canvas canvas) {
                // TODO 自動生成されたメソッド・スタブ

                float a=0;
                float b=0;

                super.onDraw(canvas);
                canvas.drawColor(Color.WHITE);
                Paint paint = new Paint();
                paint.setColor(Color.BLUE);
                paint.setAlpha(50);
                canvas.drawRect((getWidth()/2-240)+10,(getHeight()/2-343)+10,(getWidth()/2-240)+470,(getH
eight()/2-343)+675, paint);

                paint.setAlpha(10000);
                paint.setColor(Color.BLUE);

                for (int i=0;i<2;i++) {
                        canvas.drawLine((getWidth()/2-240)+10+i,(getHeight()/2-343)+10+i,(getWidth()/2-24
```

```
0)+10+i,(getHeight()/2-343)+675-i, paint);
                        canvas.drawLine((getWidth()/2-240)+10+i,(getHeight()/2-343)+675-i,(getWidth()/2-2
40)+470-i,(getHeight()/2-343)+675-i, paint);
                        canvas.drawLine((getWidth()/2-240)+470-i,(getHeight()/2-343)+675-i,(getWidth()/2-
240)+470-i,(getHeight()/2-343)+10+i, paint);
                        canvas.drawLine((getWidth()/2-240)+470-i,(getHeight()/2-343)+10+i,(getWidth()/2-2
40)+10+i,(getHeight()/2-343)+10+i, paint);
                }

                paint.setColor(Color.BLUE);
                paint.setTextSize(20.0f);
                canvas.drawText("【積が奇数の２個のさいころ】", (getWidth()/2-240)+120-20, (getHeight()/2
-343)+80,paint);

                paint.setColor(Color.BLACK);
                paint.setTextSize(17.0f);
                canvas.drawText("さいころⅠ",(getWidth()/2-240)+135,(getHeight()/2-343)+188,paint);
                canvas.drawText("さいころⅡ",(getWidth()/2-240)+255,(getHeight()/2-343)+188,paint);

                if (SaicoroActivity.ritsu != 0) {
                        a=(float)0.9*320/SaicoroActivity.ritsu;    //---------------------------- ＜画像の
拡大・縮小の横の倍率を指定する＞
                        b=(float)0.9*320/SaicoroActivity.ritsu;    //---------------------------- ＜画像
の拡大・縮小の縦の倍率を指定する＞
                }
                else {
                        a=(float) 1.0;
                        b=(float) 1.0;
                }

                Matrix Mat = new Matrix();        //----------------------***
                Mat.postScale(a, b);      //-------------------------***
                Bitmap bitmap11 = Bitmap.createBitmap(    //---------***
                                bitmap1,0,0,      //-------------------------***
                                bitmap1.getWidth(),       //---------------------***
                                bitmap1.getHeight(),      //-----------------***
                                Mat,true //-------------------------------***
                                );       //-------------------------------------***

                Bitmap bitmap22 = Bitmap.createBitmap(    //---------***
                                bitmap2,0,0,      //-------------------------***
                                bitmap2.getWidth(),       //---------------------***
                                bitmap2.getHeight(),      //-----------------***
                                Mat,true //-------------------------------***
                                );       //-------------------------------------***

                Bitmap bitmap33 = Bitmap.createBitmap(    //---------***
                                bitmap3,0,0,      //-------------------------***
                                bitmap3.getWidth(),       //---------------------***
                                bitmap3.getHeight(),      //-----------------***
                                Mat,true //-------------------------------***
                                );       //-------------------------------------***

                Bitmap bitmap44 = Bitmap.createBitmap(    //---------***
                                bitmap4,0,0,      //-------------------------***
                                bitmap4.getWidth(),       //---------------------***
                                bitmap4.getHeight(),      //-----------------***
                                Mat,true //-------------------------------***
                                );       //-------------------------------------***

                Bitmap bitmap55 = Bitmap.createBitmap(    //---------***
                                bitmap5,0,0,      //-------------------------***
                                bitmap5.getWidth(),       //---------------------***
                                bitmap5.getHeight(),      //-----------------***
                                Mat,true //-------------------------------***
                                );       //-------------------------------------***

                Bitmap bitmap66 = Bitmap.createBitmap(    //---------***
```

```
                                bitmap6,0,0,      //-------------------------***
                                bitmap6.getWidth(),       //---------------------***
                                bitmap6.getHeight(),      //-------------------***
                                Mat,true //-----------------------------***
                                );       //-------------------------------------***

                if (bitmap11 != null && bitmap22 != null && bitmap33 != null && bitmap44 != null && bitma
p55 != null && bitmap66 != null) {

                        ct1++;

                        r1=(int) (1+6*Math.random());
                        if (r1==1) {
                                canvas.drawBitmap(bitmap11, (getWidth()/2-240)+160,(getHeight()/2-343)+1
30, paint);
                        }
                        else if (r1==2) {
                                canvas.drawBitmap(bitmap22, (getWidth()/2-240)+160,(getHeight()/2-343)+1
30, paint);

                        }
                        else if (r1==3) {
                                canvas.drawBitmap(bitmap33, (getWidth()/2-240)+160,(getHeight()/2-343)+1
30, paint);

                        }
                        else if (r1==4) {
                                canvas.drawBitmap(bitmap44, (getWidth()/2-240)+160,(getHeight()/2-343)+1
30, paint);

                        }
                        else if (r1==5) {
                                canvas.drawBitmap(bitmap55, (getWidth()/2-240)+160,(getHeight()/2-343)+1
30, paint);

                        }
                        else if (r1==6) {
                                canvas.drawBitmap(bitmap66, (getWidth()/2-240)+160,(getHeight()/2-343)+1
30, paint);

                        }

                        r2=(int) (1+6*Math.random());
                        if (r2==1) {
                                canvas.drawBitmap(bitmap11, (getWidth()/2-240)+280,(getHeight()/2-343)+1
30, paint);
                        }
                        else if (r2==2) {
                                canvas.drawBitmap(bitmap22, (getWidth()/2-240)+280,(getHeight()/2-343)+1
30, paint);

                        }
                        else if (r2==3) {
                                canvas.drawBitmap(bitmap33, (getWidth()/2-240)+280,(getHeight()/2-343)+1
30, paint);

                        }
                        else if (r2==4) {
                                canvas.drawBitmap(bitmap44, (getWidth()/2-240)+280,(getHeight()/2-343)+1
30, paint);

                        }
                        else if (r2==5) {
                                canvas.drawBitmap(bitmap55, (getWidth()/2-240)+280,(getHeight()/2-343)+1
30, paint);

                        }
                        else if (r2==6) {
```

```
                                canvas.drawBitmap(bitmap66, (getWidth()/2-240)+280,(getHeight()/2-343)+1
30, paint);

                        }
                }       //if (bitmap1 != null && ···

                if (r1==1 && r2==1) {
                        d11++;
                }
                else if (r1==2 && r2==1) {
                        d21++;
                }
                else if (r1==3 && r2==1) {
                        d31++;
                }
                else if (r1==4 && r2==1) {
                        d41++;
                }
                else if (r1==5 && r2==1) {
                        d51++;
                }
                else if (r1==6 && r2==1) {
                        d61++;
                }
                else if (r1==1 && r2==2) {
                        d12++;
                }
                else if (r1==2 && r2==2) {
                        d22++;
                }
                else if (r1==3 && r2==2) {
                        d32++;
                }
                else if (r1==4 && r2==2) {
                        d42++;
                }
                else if (r1==5 && r2==2) {
                        d52++;
                }
                else if (r1==6 && r2==2) {
                        d62++;
                }
                else if (r1==1 && r2==3) {
                        d13++;
                }
                else if (r1==2 && r2==3) {
                        d23++;
                }
                else if (r1==3 && r2==3) {
                        d33++;
                }
                else if (r1==4 && r2==3) {
                        d43++;
                }
                else if (r1==5 && r2==3) {
                        d53++;
                }
                else if (r1==6 && r2==3) {
                        d63++;
                }
                else if (r1==1 && r2==4) {
                        d14++;
                }
                else if (r1==2 && r2==4) {
                        d24++;
                }
                else if (r1==3 && r2==4) {
                        d34++;
```

```
                }
                else if (r1==4 && r2==4) {
                        d44++;
                }
                else if (r1==5 && r2==4) {
                        d54++;
                }
                else if (r1==6 && r2==4) {
                        d64++;
                }
                else if (r1==1 && r2==5) {
                        d15++;
                }
                else if (r1==2 && r2==5) {
                        d25++;
                }
                else if (r1==3 && r2==5) {
                        d35++;
                }
                else if (r1==4 && r2==5) {
                        d45++;
                }
                else if (r1==5 && r2==5) {
                        d55++;
                }
                else if (r1==6 && r2==5) {
                        d65++;
                }
                else if (r1==1 && r2==6) {
                        d16++;
                }
                else if (r1==2 && r2==6) {
                        d26++;
                }
                else if (r1==3 && r2==6) {
                        d36++;
                }
                else if (r1==4 && r2==6) {
                        d46++;
                }
                else if (r1==5 && r2==6) {
                        d56++;
                }
                else if (r1==6 && r2==6) {
                        d66++;
                }

                d=d11+d13+d15+d31+d33+d35+d51+d53+d55;     //積が奇数の度数
                paint.setColor(Color.BLACK);
                paint.setTextSize(19.0F);
                canvas.drawText("積が奇数の回数＝　"+d, (getWidth()/2-240)+100, (getHeight()/2-343)+460,
paint);

                if (ct1 != 0) {
                        ritu = (float)d/(float)ct1;        //積が奇数の割合
                        paint.setColor(Color.BLUE);
                        paint.setTextSize(19.0F);
                        canvas.drawText("積が奇数の割合＝　"+ritu, (getWidth()/2-240)+100, (getHeight()/2
-343)+490, paint);
            }

                paint.setColor(Color.BLACK);
                paint.setTextSize(19.0F);
                canvas.drawText("実験回数＝ "+ct1, (getWidth()/2-240)+170, (getHeight()/2-343)+230, pain
t);

                //表の枠
                paint.setColor(Color.BLACK);
```

```
                canvas.drawLine((getWidth()/2-240)+128, (getHeight()/2-343)+255, (getWidth()/2-240)+353,
(getHeight()/2-343)+255, paint);
                canvas.drawLine((getWidth()/2-240)+128, (getHeight()/2-343)+275, (getWidth()/2-240)+193,
(getHeight()/2-343)+275, paint);
                canvas.drawLine((getWidth()/2-240)+193, (getHeight()/2-343)+275, (getWidth()/2-240)+209,
(getHeight()/2-343)+295, paint);

                for (yy=335;yy<=455;yy=yy+20){
                        canvas.drawLine((getWidth()/2-240)+128, (getHeight()/2-343)+yy-40, (getWidth()/2-
240)+353, (getHeight()/2-343)+yy-40, paint);
                }
                for (xx=353;xx>208;xx=xx-24){
                        canvas.drawLine((getWidth()/2-240)+xx, (getHeight()/2-343)+255, (getWidth()/2-24
0)+xx, (getHeight()/2-343)+415, paint);
                }
                canvas.drawLine((getWidth()/2-240)+128, (getHeight()/2-343)+255, (getWidth()/2-240)+128,
(getHeight()/2-343)+415, paint);

                //表の文字
                paint.setTextSize(14.0F);
                canvas.drawText("さいころⅡ", (getWidth()/2-240)+133, (getHeight()/2-343)+270, paint);

                paint.setTextSize(15.0F);
                canvas.drawText("１", (getWidth()/2-240)+213, (getHeight()/2-343)+280, paint);
                canvas.drawText("２", (getWidth()/2-240)+238, (getHeight()/2-343)+280, paint);
                canvas.drawText("３", (getWidth()/2-240)+263, (getHeight()/2-343)+280, paint);
                canvas.drawText("４", (getWidth()/2-240)+288, (getHeight()/2-343)+280, paint);
                canvas.drawText("５", (getWidth()/2-240)+313, (getHeight()/2-343)+280, paint);
                canvas.drawText("６", (getWidth()/2-240)+338, (getHeight()/2-343)+280, paint);

                paint.setTextSize(14.0F);
                canvas.drawText("さいころⅠ", (getWidth()/2-240)+130, (getHeight()/2-343)+290, paint);

                paint.setTextSize(15.0F);
                canvas.drawText("１", (getWidth()/2-240)+158, (getHeight()/2-343)+310, paint);
                canvas.drawText("２", (getWidth()/2-240)+158, (getHeight()/2-343)+330, paint);
                canvas.drawText("３", (getWidth()/2-240)+158, (getHeight()/2-343)+350, paint);
                canvas.drawText("４", (getWidth()/2-240)+158, (getHeight()/2-343)+370, paint);
                canvas.drawText("５", (getWidth()/2-240)+158, (getHeight()/2-343)+390, paint);
                canvas.drawText("６", (getWidth()/2-240)+158, (getHeight()/2-343)+410, paint);

                //表の度数
                paint.setTextSize(15.0F);
                paint.setColor(Color.BLUE);canvas.drawText(""+d11, (getWidth()/2-240)+213, (getHeight()/2
-343)+310, paint);
                paint.setColor(Color.BLACK);canvas.drawText(""+d12, (getWidth()/2-240)+237, (getHeight()/
2-343)+310, paint);
                paint.setColor(Color.BLACK);canvas.drawText(""+d21, (getWidth()/2-240)+213, (getHeight()/
2-343)+330, paint);
                paint.setColor(Color.BLACK);canvas.drawText(""+d22, (getWidth()/2-240)+237, (getHeight()/
2-343)+330, paint);
                paint.setColor(Color.BLUE);canvas.drawText(""+d31, (getWidth()/2-240)+213, (getHeight()/2
-343)+350, paint);
                paint.setColor(Color.BLACK);canvas.drawText(""+d32, (getWidth()/2-240)+237, (getHeight()/
2-343)+350, paint);
                paint.setColor(Color.BLACK);canvas.drawText(""+d41, (getWidth()/2-240)+213, (getHeight()/
2-343)+370, paint);
                paint.setColor(Color.BLACK);canvas.drawText(""+d42, (getWidth()/2-240)+237, (getHeight()/
2-343)+370, paint);
                paint.setColor(Color.BLUE);canvas.drawText(""+d51, (getWidth()/2-240)+213, (getHeight()/2
-343)+390, paint);
                paint.setColor(Color.BLACK);canvas.drawText(""+d52, (getWidth()/2-240)+237, (getHeight()/
2-343)+390, paint);
                paint.setColor(Color.BLACK);canvas.drawText(""+d61, (getWidth()/2-240)+213, (getHeight()/
2-343)+410, paint);
                paint.setColor(Color.BLACK);canvas.drawText(""+d62, (getWidth()/2-240)+237, (getHeight()/
2-343)+410, paint);
                paint.setColor(Color.BLUE);canvas.drawText(""+d13, (getWidth()/2-240)+261, (getHeight()/2
```

```
-343)+310, paint);
                paint.setColor(Color.BLACK);canvas.drawText(""+d14, (getWidth()/2-240)+285, (getHeight()/
2-343)+310, paint);
                paint.setColor(Color.BLACK);canvas.drawText(""+d23, (getWidth()/2-240)+261, (getHeight()/
2-343)+330, paint);
                paint.setColor(Color.BLACK);canvas.drawText(""+d24, (getWidth()/2-240)+285, (getHeight()/
2-343)+330, paint);
                paint.setColor(Color.BLUE);canvas.drawText(""+d33, (getWidth()/2-240)+261, (getHeight()/2
-343)+350, paint);
                paint.setColor(Color.BLACK);canvas.drawText(""+d34, (getWidth()/2-240)+285, (getHeight()/
2-343)+350, paint);
                paint.setColor(Color.BLACK);canvas.drawText(""+d43, (getWidth()/2-240)+261, (getHeight()/
2-343)+370, paint);
                paint.setColor(Color.BLACK);canvas.drawText(""+d44, (getWidth()/2-240)+285, (getHeight()/
2-343)+370, paint);
                paint.setColor(Color.BLUE);canvas.drawText(""+d53, (getWidth()/2-240)+261, (getHeight()/2
-343)+390, paint);
                paint.setColor(Color.BLACK);canvas.drawText(""+d54, (getWidth()/2-240)+285, (getHeight()/
2-343)+390, paint);
                paint.setColor(Color.BLACK);canvas.drawText(""+d63, (getWidth()/2-240)+261, (getHeight()/
2-343)+410, paint);
                paint.setColor(Color.BLACK);canvas.drawText(""+d64, (getWidth()/2-240)+285, (getHeight()/
2-343)+410, paint);
                paint.setColor(Color.BLUE);canvas.drawText(""+d15, (getWidth()/2-240)+309, (getHeight()/2
-343)+310, paint);
                paint.setColor(Color.BLACK);canvas.drawText(""+d16, (getWidth()/2-240)+333, (getHeight()/
2-343)+310, paint);
                paint.setColor(Color.BLACK);canvas.drawText(""+d25, (getWidth()/2-240)+309, (getHeight()/
2-343)+330, paint);
                paint.setColor(Color.BLACK);canvas.drawText(""+d26, (getWidth()/2-240)+333, (getHeight()/
2-343)+330, paint);
                paint.setColor(Color.BLUE);canvas.drawText(""+d35, (getWidth()/2-240)+309, (getHeight()/2
-343)+350, paint);
                paint.setColor(Color.BLACK);canvas.drawText(""+d36, (getWidth()/2-240)+333, (getHeight()/
2-343)+350, paint);
                paint.setColor(Color.BLACK);canvas.drawText(""+d45, (getWidth()/2-240)+309, (getHeight()/
2-343)+370, paint);
                paint.setColor(Color.BLACK);canvas.drawText(""+d46, (getWidth()/2-240)+333, (getHeight()/
2-343)+370, paint);
                paint.setColor(Color.BLUE);canvas.drawText(""+d55, (getWidth()/2-240)+309, (getHeight()/2
-343)+390, paint);
                paint.setColor(Color.BLACK);canvas.drawText(""+d56, (getWidth()/2-240)+333, (getHeight()/
2-343)+390, paint);
                paint.setColor(Color.BLACK);canvas.drawText(""+d65, (getWidth()/2-240)+309, (getHeight()/
2-343)+410, paint);
                paint.setColor(Color.BLACK);canvas.drawText(""+d66, (getWidth()/2-240)+333, (getHeight()/
2-343)+410, paint);

                paint.setColor(Color.BLACK);
                paint.setTextSize(18.0F);
                canvas.drawText("※ 画面を５回タッチすると自動になります。", (getWidth()/2-240)+50-20, (g
etHeight()/2-343)+530, paint);
                canvas.drawText("※ 画面をタッチすると自動が止まります。", (getWidth()/2-240)+50-20, (get
Height()/2-343)+555, paint);
                canvas.drawText("※ 更に画面をタッチすると初期化されます。", (getWidth()/2-240)+50-20, (g
etHeight()/2-343)+580, paint);
                canvas.drawText("※ 画面が暗くなったらステータスバーをタッチ！", (getWidth()/2-240)+50-2
0, (getHeight()/2-343)+605, paint);

                paint.setColor(Color.BLUE);
                paint.setTextSize(19.0F);
                canvas.drawText("Copyright(C) K.Niwa 2014.08.06", (getWidth()/2-240)+100, (getHeight()/2-
343)+640, paint);          //作者表示

                if (flag >= 5) {
                    if (d11<98 && d21<98 && d31<98 && d41<98 && d51<98 && d61<98 && d12<98 && d22<98
&& d32<98 && d42<98 && d52<98 && d62<98 && d13<98 && d23<98 && d33<98 && d43<98 && d53<98 && d63<98 && d14
<98 && d24<98 && d34<98 && d44<98 && d54<98 && d64<98 && d15<98 && d25<98 && d35<98 && d45<98 && d55<98 &&
```

```
 d65<98 && d16<98 && d26<98 && d36<98 && d46<98 && d56<98 && d66<98) {
                                invalidate();
                        }
                }

        }       //protected void onDraw(Canvas canvas)

        @Override
        public boolean onTouchEvent(MotionEvent event) {
                flag++;
                flag = flag % 6;

                syoki++;
                if (syoki > 6) {
                        flag=0;                                                                         //自
動識別子
                syoki=0;                                                                        //初期化識別子
                        ct1=0;                                                                          //実
験回数
                        d11=0;d21=0;d31=0;d41=0;d51=0;d61=0;      //度数 d31とはさいころ１の目が３で、さ
いころ２の目が１の度数
                        d12=0;d22=0;d32=0;d42=0;d52=0;d62=0;
                        d13=0;d23=0;d33=0;d43=0;d53=0;d63=0;
                        d14=0;d24=0;d34=0;d44=0;d54=0;d64=0;
                        d15=0;d25=0;d35=0;d45=0;d55=0;d65=0;
                        d16=0;d26=0;d36=0;d46=0;d56=0;d66=0;
                        d=0;                                                                            //積
が奇数の度数
                        ritu=0;                                                                         //積
が奇数の割合
                }

                invalidate();
                return false;
        }

}
```

[２] main.xml

```
<?xml version="1.0" encoding="utf-8"?>

<LinearLayout
        xmlns:android="http://schemas.android.com/apk/res/android"
        android:orientation="vertical"
        android:layout_width="match_parent"
        android:layout_height="match_parent"
        android:weightSum="1" >

        <jp.seitoku.saicoro.Saicoro
                android:layout_height="match_parent"
                android:layout_width="match_parent"
                android:id="@+id/myview1">
        </jp.seitoku.saicoro.Saicoro>
</LinearLayout>
```

[３] SaicoroActivity.java

```
package jp.seitoku.saicoro;

import android.os.Bundle;
import android.app.Activity;
import android.view.Menu;
```

```
import android.util.DisplayMetrics;       //----------------------------〈画像の拡大・縮小に必要なライブラリ〉

public class SaicoroActivity extends Activity {

    static int ritsu;

        @Override
    public void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);
        setContentView(R.layout.main);

        DisplayMetrics metrics = new DisplayMetrics();     //---------------------------〈端末の情報を取得する〉
                getWindowManager().getDefaultDisplay().getMetrics(metrics);//-----------------------------------
                StringBuilder buffer = new StringBuilder();        //-------------------------------------------
        buffer.append("densityDpi（ドット数／インチ）:" + String.valueOf(metrics.densityDpi) + "¥n");
//------
        ritsu=metrics.densityDpi;// ------------------------------------------------これで値が取り出せた!
        }

    @Override
    public boolean onCreateOptionsMenu(Menu menu) {
        getMenuInflater().inflate(R.menu.main, menu);
        return true;
    }
}
```