

```
[1]MyTeppouEng.java
```

```
/*
```

```
-----  
    下手な鉄砲も数撃ちや当たる！(英語版)  
    Android 4.1 (Jelly Bean)  
    Copyright (C) K.Niwa 2021. 11. 29  
-----
```

```
*/
```

```
package jp.kiyo.wuena.myteppoueng;
```

```
import android.content.Context;
```

```
import android.graphics.Canvas;
```

```
import android.graphics.Color;
```

```
import android.graphics.Paint;
```

```
import android.graphics.Rect;
```

```
import android.graphics.RectF;
```

```
import android.util.AttributeSet;
```

```
import android.view.MotionEvent;
```

```
import android.view.View;
```

```
public class MyTeppouEng extends View {
```

```
    int px,py;        //玉の位置
```

```
    double i;        //的の描写に使用
```

```
    int flag=0;        //自動識別子 (flag==1:自動 start、flag==2:自動 stop、flag==0:発射回数・  
    当たり回数・実験回数等の初期化)
```

```
    int ct1=0;        //発射回数
```

```
    int ct2=0;        //当たり回数
```

```
    int ct3=0;        //実験回数
```

```
    int seikou=0;    //成功回数
```

```
    float ritu=0;    //成功率
```

```
    double r1;        //動径の長さ0以上100未満 (乱数)
```

```
    double r2;        //偏角の大きさ0以上2π未満 (乱数)
```

```
public MyTeppouEng(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 MyTeppouEng(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 MyTeppouEng(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);
}

@Override
protected void onDraw(Canvas canvas) {
    // TODO 自動生成されたメソッド・スタブ
    super.onDraw(canvas);
    canvas.drawColor(Color.WHITE);
    Paint paint = new Paint();
    paint.setColor(Color.BLUE);
    paint.setAlpha(50);
    canvas.drawRect((getWidth()/2-360)+20, (getHeight()/2-600)+10, (getWidth()/2-
360)+700, (getHeight()/2-600)+1190, paint);

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

    for (int i=0;i<2;i++) {
        canvas.drawLine((getWidth()/2-360)+20+i, (getHeight()/2-600)+10+i, (getWidth()/2-
360)+20+i, (getHeight()/2-600)+1190-i, paint);
        canvas.drawLine((getWidth()/2-360)+20+i, (getHeight()/2-600)+1190-i, (getWidth()/2-
360)+700-i, (getHeight()/2-600)+1190-i, paint);
    }
}

```

```

        canvas.drawLine((getWidth()/2-360)+700-i, (getHeight()/2-600)+1190-i, (getWidth()/2-
360)+700-i, (getHeight()/2-600)+10+i, paint);
        canvas.drawLine((getWidth()/2-360)+700-i, (getHeight()/2-600)+10+i, (getWidth()/2-
360)+20+i, (getHeight()/2-600)+10+i, paint);
    }

    paint.setColor(Color.BLUE);
    paint.setTextSize(35.0f);
    canvas.drawText("【Even If You Shoot a Bad Gun,”, (getWidth()/2-360)+55-10-
5, (getHeight()/2-600)+70-10, paint);
    canvas.drawText("You Can Hit If You Shoot a Lot】”, (getWidth()/2-360)+55+100-
60+80, (getHeight()/2-600)+70-10+30, paint);

    paint.setColor(Color.BLUE);
    paint.setTextSize(30.0f);
    canvas.drawText("Copyright(C) Sohun 2021.11.29”, (getWidth()/2-
360)+150, (getHeight()/2-600)+1130, paint);

    //枠の作成
    paint.setColor(Color.BLACK);
    canvas.drawLine((getWidth()/2-360)+90+120-30, (getHeight()/2-600)+90+30,
(getWidth()/2-360)+390+120+30, (getHeight()/2-600)+90+30, paint);
    canvas.drawLine((getWidth()/2-360)+90+120-30, (getHeight()/2-600)+90+30,
(getWidth()/2-360)+90+120-30, (getHeight()/2-600)+390+30, paint);
    canvas.drawLine((getWidth()/2-360)+90+120-30, (getHeight()/2-600)+390+30,
(getWidth()/2-360)+390+120+30, (getHeight()/2-600)+390+30, paint);
    canvas.drawLine((getWidth()/2-360)+390+120+30, (getHeight()/2-600)+90+30,
(getWidth()/2-360)+390+120+30, (getHeight()/2-600)+390+30, paint);
    canvas.drawLine((getWidth()/2-360)+91+120-30, (getHeight()/2-600)+91+30,
(getWidth()/2-360)+389+120+30, (getHeight()/2-600)+91+30, paint);
    canvas.drawLine((getWidth()/2-360)+91+120-30, (getHeight()/2-600)+91+30,
(getWidth()/2-360)+91+120-30, (getHeight()/2-600)+389+30, paint);
    canvas.drawLine((getWidth()/2-360)+91+120-30, (getHeight()/2-600)+389+30,
(getWidth()/2-360)+389+120+30, (getHeight()/2-600)+389+30, paint);
    canvas.drawLine((getWidth()/2-360)+389+120+30, (getHeight()/2-600)+91+30,
(getWidth()/2-360)+389+120+30, (getHeight()/2-600)+389+30, paint);

```

```

//的の描写
paint.setColor(Color.WHITE);
canvas.drawCircle((getWidth()/2-360)+240+120, (getHeight()/2-600)+240+30, 10, paint);

//発射回数を1~20回に設定する
ct1=ct1 % 20;
if (ct1==0) { //発射回数が0回になったら、当たり回数を0回にし、実験回数を1回増やす
    ct2=0;
    ct3++;
}

if (ct1<20) { //発射回数が19回目までのとき
    ct1=ct1+1;
    r1=Math.random(); //極座標の動径の長さ(0以上100未満を作る)
    r2=Math.random(); //極座標の偏角の大きさ(0以上2π未満を作る)
    px=(int)(120+150+90+100*r1*Math.cos(2*Math.PI*r2));
    py=(int)(30+150+90-100*r1*Math.sin(2*Math.PI*r2));

//鉄砲の弾の描写
paint.setColor(Color.RED);
canvas.drawCircle((getWidth()/2-360)+px, (getHeight()/2-600)+py, 3, paint);

if (r1<0.1) {
    ct2=ct2+1; //当たり回数を1つ増やす
}

if (ct1==20 &&ct2>0) { //発射20回目で当たり回数が1回以上のとき
    seikou=seikou+1; //成功回数を1回増やす
    paint.setColor(Color.BLUE);
    paint.setTextSize(60.0f);
    canvas.drawText("success!!", (getWidth()/2-360)+100+120, (getHeight()/2-
600)+180+30, paint);
}

else if (ct1==20 && ct2==0) { //発射20回目で当たり回数が0回するとき
    paint.setColor(Color.RED);

```

```

        paint.setTextSize(60.0f);
        canvas.drawText("failure!!", (getWidth()/2-360)+100+120, (getHeight()/2-
600)+180+30, paint);
    }

}

paint.setColor(Color.BLACK);
paint.setTextSize(30.0f);
canvas.drawText("Number of shots = "+ct1, (getWidth()/2-360)+190+80-50, (getHeight()/2-
600)+120+30, paint);
canvas.drawText("Number of hits = "+ct2, (getWidth()/2-360)+180+80-20, (getHeight()/2-
600)+375+30, paint);
paint.setTextSize(40.0f);
canvas.drawText("Number of successes", (getWidth()/2-360)+50+30, (getHeight()/2-
600)+530-50, paint);
canvas.drawText("/ Number of experiments", (getWidth()/2-
360)+50+30+100, (getHeight()/2-600)+565+5-50, paint);
canvas.drawText("= "+seikou+" / "+ct3, (getWidth()/2-360)+50+30, (getHeight()/2-
600)+600+10-50, paint);
ritu=(float)seikou/(float)ct3;
paint.setColor(Color.BLUE);
paint.setTextSize(40.0f);
canvas.drawText("= "+ritu, (getWidth()/2-360)+50+30, (getHeight()/2-600)+565+35+35+15-
50, paint);

paint.setColor(Color.BLACK);
paint.setTextSize(40.0f);
canvas.drawText("Mathematical probability", (getWidth()/2-360)+50+30, (getHeight()/2-
600)+730-50, paint);
canvas.drawText("of hitting at least once", (getWidth()/2-
360)+50+30+100, (getHeight()/2-600)+730+40-50, paint);
canvas.drawText("= 0.8784233", (getWidth()/2-360)+50+30, (getHeight()/2-600)+810-50,
paint);

paint.setColor(Color.BLACK);

```

```

        paint.setTextSize(30.0f);
        canvas.drawText("※ Consider the probability of hitting at least", (getWidth()/2-
360)+50, (getHeight()/2-600)+890-50, paint);
        canvas.drawText(" once when you shoot 20 times with a gun", (getWidth()/2-
360)+50, (getHeight()/2-600)+920-50, paint);
        canvas.drawText(" shot that hits once in 10 times.", (getWidth()/2-
360)+50, (getHeight()/2-600)+900, paint);

        canvas.drawText("※ When you touch the screen , it repeats", (getWidth()/2-
360)+50, (getHeight()/2-600)+940, paint);
        canvas.drawText(" in the order of [Auto start] → [Auto stop]", (getWidth()/2-
360)+50, (getHeight()/2-600)+1020-50, paint);
        canvas.drawText(" → [Initialization].", (getWidth()/2-360)+50, (getHeight()/2-
600)+1000, paint);
        canvas.drawText("※ When the screen goes dark ,", (getWidth()/2-
360)+50, (getHeight()/2-600)+1040, paint);
        canvas.drawText("touch the title bar !", (getWidth()/2-360)+50+300+30, (getHeight()/2-
600)+1040+30, paint);

        if (flag==1) { //flag==1 のとき、自動発射がスタートする。
            invalidate();
        }

    }

    public void init(Context context) {

    }

    @Override
    public boolean onTouchEvent(MotionEvent event) {
        flag++; //flag==2 のとき、自動発射がストップする。
        flag = flag % 3; //自動識別子 (flag==1:自動 start、flag==2:自動 stop、flag==0:発射回
数等の初期化)
        invalidate();
        if (flag==0) { //flag==0 のとき、発射回数等の初期化を行う。
            ct1=0; //発射回数

```

```

        ct2=0;    //当たり回数
        ct3=0;    //実験回数
        seikou=0; //成功回数
        ritu=0;   //成功率
    }

    return false;
}
}

```

[2]activity_main.xml

```

<?xml version="1.0" encoding="utf-8" ?>
<androidx.constraintlayout.widget.ConstraintLayout
xmlns:android="http://schemas.android.com/apk/res/android"
    xmlns:app="http://schemas.android.com/apk/res-auto"
    xmlns:tools="http://schemas.android.com/tools"
    android:layout_width="match_parent"
    android:layout_height="match_parent"
    tools:context=".MainActivity">

    <TextView
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:text="Hello World!"
        app:layout_constraintBottom_toBottomOf="parent"
        app:layout_constraintLeft_toLeftOf="parent"
        app:layout_constraintRight_toRightOf="parent"
        app:layout_constraintTop_toTopOf="parent" />

    <jp.kiyo.wuena.myteppoueng.MyTeppouEng
        android:id="@+id/myfview1"
        android:layout_height="match_parent"
        android:layout_width="match_parent"/>

```

```
</androidx.constraintlayout.widget.ConstraintLayout>
```

```
[3]MainActivity.java
```

```
/*
```

```
-----  
    下手な鉄砲も数撃ちや当たる！(英語版)
```

```
        Android 4.1 (Jelly Bean)
```

```
        Copyright (C) K. Niwa 2021. 9. 29  
-----
```

```
*/
```

```
package jp.kiyo.wuena.myteppoueng;
```

```
import androidx.appcompat.app.AppCompatActivity;
```

```
import android.os.Bundle;
```

```
public class MainActivity extends AppCompatActivity {
```

```
    @Override
```

```
    protected void onCreate(Bundle savedInstanceState) {
```

```
        super.onCreate(savedInstanceState);
```

```
        setContentView(R.layout.activity_main);
```

```
    }
```

```
}
```