

[1]MyZyannkenEng. java

/*

3人のじゃんけん(英語版)

Android 4.1 (Jelly Bean)

Copyright (C) K. Niwa 2021. 9. 22

*/

package jp.kiyo.wuena.myzyannkeneng;

import android.content.Context;

import android.graphics.Canvas;

import android.graphics.Color;

import android.graphics.Paint;

import android.graphics.Rect;

import android.util.AttributeSet;

import android.view.View;

import android.content.res.Resources; //画像用

import android.graphics.*;

import android.view.*;

public class MyZyannkenEng **extends** View {

private Bitmap **bitmap1** = **null**;

private Bitmap **bitmap2** = **null**;

private Bitmap **bitmap3** = **null**;

int **ct**=0; //実験回数カウンター

int **r1**=1, **r2**=2, **r3**=3; //A君はグー、B君はチョキ、C君はパー

int **d1**=0, **d2**=0, **d3**=0, **d4**=0, **d5**=0, **d6**=0, **d7**=0; //勝ち負けのカウンター

int **flag**=0; //自動識別子

int **syoki**=0; //初期化識別子

int **width**;

int **height**;

```
float a=0;
```

```
float b=0;
```

```
public MyZyannkenEng(Context context) {  
    super(context);  
    init(context);  
}
```

```
public MyZyannkenEng(Context context, AttributeSet attrs) {  
    super(context, attrs);  
    init(context);  
}
```

```
public MyZyannkenEng(Context context, AttributeSet attrs, int defStyle) {  
    super(context, attrs, defStyle);  
    init(context);  
}
```

```
private void init(Context context) {  
    Resources res = context.getResources();  
    bitmap1 = BitmapFactory.decodeResource(res, R.drawable.guu);  
    bitmap2 = BitmapFactory.decodeResource(res, R.drawable.pii);  
    bitmap3 = BitmapFactory.decodeResource(res, R.drawable.paa);  
  
    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;

ct++;

super.onDraw(canvas);
canvas.drawColor(Color.WHITE);
Paint paint = new Paint();
paint.setColor(Color.BLUE);
paint.setAlpha(50);
canvas.drawRect((getWidth()/2-360)+10, (getHeight()/2-600)+10, (getWidth()/2-
360)+710, (getHeight()/2-600)+1190, paint);

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

for (int i=0;i<2;i++) {
    canvas.drawLine((getWidth()/2-360)+10+i, (getHeight()/2-600)+10+i, (getWidth()/2-
360)+10+i, (getHeight()/2-600)+1190-i, paint);
    canvas.drawLine((getWidth()/2-360)+10+i, (getHeight()/2-600)+1190-i, (getWidth()/2-
360)+710-i, (getHeight()/2-600)+1190-i, paint);
    canvas.drawLine((getWidth()/2-360)+710-i, (getHeight()/2-600)+1190-i, (getWidth()/2-
360)+710-i, (getHeight()/2-600)+10+i, paint);
    canvas.drawLine((getWidth()/2-360)+710-i, (getHeight()/2-600)+10+i, (getWidth()/2-
360)+10+i, (getHeight()/2-600)+10+i, paint);
}

paint.setColor(Color.BLUE);
paint.setTextSize(45.0f);
canvas.drawText(" 【Three Rock-Paper-Scissors】 ", (getWidth()/2-360)+120-30+55-95,
(getHeight()/2-600)+80, paint);

paint.setColor(Color.BLACK);
paint.setTextSize(30.0f);
canvas.drawText(" A ", (getWidth()/2-360)+123+110+10, (getHeight()/2-600)+210+40-25,

```

```

paint);

    canvas.drawText("B", (getWidth()/2-360)+223+110+10, (getHeight()/2-600)+210+40-25,
paint);

    canvas.drawText("C", (getWidth()/2-360)+323+110+10, (getHeight()/2-600)+210+40-25,
paint);

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

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

    //Mat.postScale(a, b); //-----***
    Bitmap bitmap33 = Bitmap.createBitmap( //-----***

```

```

        bitmap3, 0, 0,    //-----***
        bitmap3.getWidth(),    //-----***
        bitmap3.getHeight(),    //-----***
        Mat, true    //-----***
    ); //-----***

    if (bitmap11 != null && bitmap22 != null && bitmap33 != null) {

        //A 君のじゃんけんの判断
        r1=(int) (1+3*Math.random());
        if (r1==1) {
            canvas.drawBitmap(bitmap11, (getWidth()/2-360)+128-10+110, (getHeight()/2-
600)+150-5-25, paint);
        }
        else if (r1==2) {
            canvas.drawBitmap(bitmap22, (getWidth()/2-360)+128-10+110, (getHeight()/2-
600)+150-5-25, paint);
        }
        else if (r1==3) {
            canvas.drawBitmap(bitmap33, (getWidth()/2-360)+128-10+110, (getHeight()/2-
600)+150-5-25, paint);
        }

        //B 君のじゃんけんの判断
        r2=(int) (1+3*Math.random());
        if (r2==1) {
            canvas.drawBitmap(bitmap11, (getWidth()/2-360)+228-10+110, (getHeight()/2-
600)+150-5-25, paint);
        }
        else if (r2==2) {
            canvas.drawBitmap(bitmap22, (getWidth()/2-360)+228-10+110, (getHeight()/2-
600)+150-5-25, paint);
        }
        else if (r2==3) {
            canvas.drawBitmap(bitmap33, (getWidth()/2-360)+228-10+110, (getHeight()/2-
600)+150-5-25, paint);
        }
    }
}

```

```

    }

    //C 君のじゃんけんの判断
    r3=(int) (1+3*Math.random());
    if (r3==1) {
        canvas.drawBitmap(bitmap11, (getWidth()/2-360)+328-10+110, (getHeight()/2-
600)+150-5-25, paint);
    }
    else if (r3==2) {
        canvas.drawBitmap(bitmap22, (getWidth()/2-360)+328-10+110, (getHeight()/2-
600)+150-5-25, paint);
    }
    else if (r3==3) {
        canvas.drawBitmap(bitmap33, (getWidth()/2-360)+328-10+110, (getHeight()/2-
600)+150-5-25, paint);
    }
}

//勝ち負けの判断
//あいこ
if (r1==1 && r2==1 && r3==1) {
    d1++;
}
else if (r1==2 && r2==2 && r3==2) {
    d1++;
}
else if (r1==3 && r2==3 && r3==3) {
    d1++;
}
else if (r1==1 && r2==2 && r3==3) {
    d1++;
}
else if (r1==1 && r2==3 && r3==2) {
    d1++;
}
else if (r1==2 && r2==1 && r3==3) {

```

```
        d1++;
    }
    else if (r1==2 && r2==3 && r3==1) {
        d1++;
    }
    else if (r1==3 && r2==1 && r3==2) {
        d1++;
    }
    else if (r1==3 && r2==2 && r3==1) {
        d1++;
    }
    //A 君だけが勝つ
    if (r1==1 && r2==2 && r3==2) {
        d2++;
    }
    else if (r1==2 && r2==3 && r3==3) {
        d2++;
    }
    else if (r1==3 && r2==1 && r3==1) {
        d2++;
    }
    //B 君だけが勝つ
    if (r1==2 && r2==1 && r3==2) {
        d3++;
    }
    else if (r1==3 && r2==2 && r3==3) {
        d3++;
    }
    else if (r1==1 && r2==3 && r3==1) {
        d3++;
    }
    //C 君だけが勝つ
    if (r1==2 && r2==2 && r3==1) {
        d4++;
    }
    else if (r1==3 && r2==3 && r3==2) {
```

```
        d4++;
    }
    else if (r1==1 && r2==1 && r3==3) {
        d4++;
    }

    //A 君だけが負ける
    if (r1==1 && r2==3 && r3==3) {
        d5++;
    }
    else if (r1==2 && r2==1 && r3==1) {
        d5++;
    }
    else if (r1==3 && r2==2 && r3==2) {
        d5++;
    }

    //B 君だけが負ける
    if (r1==3 && r2==1 && r3==3) {
        d6++;
    }
    else if (r1==1 && r2==2 && r3==1) {
        d6++;
    }
    else if (r1==2 && r2==3 && r3==2) {
        d6++;
    }

    //C 君だけが負ける
    if (r1==3 && r2==3 && r3==1) {
        d7++;
    }
    else if (r1==1 && r2==1 && r3==2) {
        d7++;
    }
    else if (r1==2 && r2==2 && r3==3) {
        d7++;
    }
}
```

```

        paint.setColor (Color. BLACK) ;
        paint.setTextSize (35.0f) ;
        canvas.drawText ("Number of draws "+d1+" ( "+((float) (d1)/(float) (ct))+ " )",
(getWidth()/2-360)+30+20, (getHeight()/2-600)+335, paint);
        canvas.drawText ("Number of times"+"", (getWidth()/2-360)+30+20, (getHeight()/2-
600)+375, paint);
        canvas.drawText ("only A won "+d2+" ( "+((float) (d2)/(float) (ct))+ " )", (getWidth()/2-
360)+30+100+20, (getHeight()/2-600)+405, paint);
        canvas.drawText ("Number of times"+"", (getWidth()/2-360)+30+20, (getHeight()/2-
600)+445, paint);
        canvas.drawText ("only B won "+d3+" ( "+((float) (d3)/(float) (ct))+ " )", (getWidth()/2-
360)+30+100+20, (getHeight()/2-600)+475, paint);
        canvas.drawText ("Number of times"+"", (getWidth()/2-360)+30+20, (getHeight()/2-
600)+515, paint);
        canvas.drawText ("only C won "+d4+" ( "+((float) (d4)/(float) (ct))+ " )", (getWidth()/2-
360)+30+100+20, (getHeight()/2-600)+545, paint);
        canvas.drawText ("Number of times"+"", (getWidth()/2-360)+30+20, (getHeight()/2-
600)+585, paint);
        canvas.drawText ("only A lost "+d5+" ( "+((float) (d5)/(float) (ct))+ " )", (getWidth()/2-
360)+30+100+20, (getHeight()/2-600)+615, paint);
        canvas.drawText ("Number of times"+"", (getWidth()/2-360)+30+20, (getHeight()/2-
600)+655, paint);
        canvas.drawText ("only B lost "+d6+" ( "+((float) (d6)/(float) (ct))+ " )", (getWidth()/2-
360)+30+20+100, (getHeight()/2-600)+685, paint);
        canvas.drawText ("Number of times"+"", (getWidth()/2-360)+30+20, (getHeight()/2-
600)+725, paint);
        canvas.drawText ("only C lost "+d7+" ( "+((float) (d7)/(float) (ct))+ " )", (getWidth()/2-
360)+30+100+20, (getHeight()/2-600)+755, paint);
        paint.setColor (Color. BLUE) ;
        canvas.drawText ("Number of experiments "+ct, (getWidth()/2-360)+30+20, (getHeight()/2-
600)+295, paint);

        paint.setColor (Color. BLACK) ;
        paint.setTextSize (30.0f) ;
        canvas.drawText ("■The numbers in parentheses represent ", (getWidth()/2-360)+50,
(getHeight()/2-600)+795, paint);

```

```

        canvas.drawText(" the percentage of times.", (getWidth()/2-360)+50, (getHeight()/2-600)+820, paint);
        canvas.drawText("■Let's observe that draw rate approaches ", (getWidth()/2-360)+50, (getHeight()/2-600)+860, paint);
        canvas.drawText(" 0.3333... and percentages other than ", (getWidth()/2-360)+50, (getHeight()/2-600)+885, paint);
        canvas.drawText(" draws approach 0.1111....", (getWidth()/2-360)+50, (getHeight()/2-600)+910, paint);

```

```

        canvas.drawText("Touch the screen five times to activate.", (getWidth()/2-360)+50, (getHeight()/2-600)+950, paint);
        canvas.drawText("Touch the screen again to stop the auto.", (getWidth()/2-360)+50, (getHeight()/2-600)+990, paint);
        canvas.drawText("If you touch it further, it will be initialized.", (getWidth()/2-360)+50, (getHeight()/2-600)+1030, paint);
        canvas.drawText("When the screen goes dark, touch the title bar !", (getWidth()/2-360)+50, (getHeight()/2-600)+1070, paint);

```

```

        paint.setColor(Color.BLUE);
        paint.setTextSize(30.0f);
        canvas.drawText("Copyright(C) Sohun 2021.9.22", (getWidth()/2-360)+150+5, (getHeight()/2-600)+1130, paint);

```

```

        if (flag>=5) {
            invalidate();
        }
    }
}

```

```

@Override
public boolean onTouchEvent(MotionEvent event) {
    //invalidate();
    flag=flag+1;
    flag=flag % 6;
}

```

```

        syoki=syoki+1;
        if (syoki > 6) {
            ct=0;                                //実験回数カウンター
            d1=0;d2=0;d3=0;d4=0;d5=0;d6=0;d7=0;    //勝ち負けのカウンター
            flag=0;                                //自動識別子
            syoki=0;                                //初期化識別子
        }

        invalidate();
        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.myzyannkeneng.MyZyannkenEng
        android:id="@+id/myfview1"

```

```
        android:layout_height="match_parent"
        android:layout_width="match_parent"/>
```

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

[3]MainActivity.java

```
/*
```

```
-----
        3人のじゃんけん(英語版)
        Android 4.1 (Jelly Bean)
        Copyright (C) K. Niwa 2021. 9. 22
    -----
```

```
*/
```

```
package jp.kiyo.wuena.myzyannkeneng;
```

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

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

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

```
import android.app.Activity;
```

```
import android.view.Menu;
```

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

```
    static int ritsu;
```

```
    @Override
```

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

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

```
        setContentView(R.layout. activity_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)
```

```
+ "Yn");  
    ritsu=metrics.densityDpi;  
  }  
}
```