

[1]MainActivity.java

/*

 コラッツの問題
 Android 4.1 (Jelly Bean)
 Copyright (C) K. Niwa 2021. 3. 13

*/

package jp.kiyo.wuena.myp3xplus12;

import android.app.Activity; //ライブラリを読み込む

import android.graphics.Color;

import android.os.Bundle;

import android.text.Editable;

import android.view.*;

import android.view.View.OnClickListener;

import android.widget.*;

import androidx.appcompat.app.AppCompatActivity;

public class MainActivity **extends** AppCompatActivity {

 TextView

text, text0, text1, text2, text3, text4, text5, text6, text7, text8, text9, text10, text11, text12, text13,
 text14, text15; //テキストを表示する「テキストビュー型」で宣言する

 TextView **text22, text33, text92, text122;**

 EditText **edit1;** //テキストを入力する「エディットテキスト型」で宣言する

 Button **button1, button2;** //「ボタン型」で宣言する

int n; //整数型で宣言する 入力する最初の自然数

int res; //整数型で宣言する 1に達するまでの途中のそれぞれの計算結果

int flag=0; //整数型で宣言し初期化する 初期化識別子

 @Override

protected void onCreate(Bundle savedInstanceState) {

super. onCreate(savedInstanceState);

```
setContentView(R.layout. activity_main); //★★★画面を呼び出す
```

```
LinearLayout layout = new LinearLayout(this); //リニアレイアウト型で宣言する
```

```
layout.setOrientation(LinearLayout. VERTICAL); //部品の並べ方を縦に宣言する
```

```
text = new TextView(this); //テキストビュー型として実体化する  
text.setTextColor(Color. BLUE); //テキストを白色に指定する  
text.set textSize(22f); //テキストの大きさを 15f に指定する  
text.setText(" 【3 x + 1 の問題】"); //テキストを表示する  
layout.addView(text); //テキストビューをレイアウトに貼り付ける
```

```
text1 = new TextView(this); //テキストビュー型として実体化する  
text1.setTextColor(Color. BLACK); //テキストを白色に指定する  
text1.set textSize(13f); //テキストの大きさを 15f に指定する  
text1.setText(""); //テキストを表示する  
layout.addView(text1); //テキストビューをレイアウトに貼り付ける
```

```
text2 = new TextView(this); //テキストビュー型として実体化する  
text2.setTextColor(Color. BLACK); //テキストを白色に指定する  
text2.set textSize(18f); //テキストの大きさを 15f に指定する  
text2.setText(" ■まず、[計算]ボタンの上の線の上側をタ"); //テキストを
```

示する

```
layout.addView(text2); //テキストビューをレイアウトに貼り付ける
```

```
text22 = new TextView(this); //テキストビュー型として実体化する  
text22.setTextColor(Color. BLACK); //テキストを白色に指定する  
text22.set textSize(18f); //テキストの大きさを 15f に指定する  
text22.setText(" タッチしてください..."); //テキストを表示する  
layout.addView(text22); //テキストビューをレイアウトに貼り付ける
```

```
text3 = new TextView(this); //テキストビュー型として実体化する  
text3.setTextColor(Color. BLACK); //テキストを白色に指定する  
text3.set textSize(18f); //テキストの大きさを 15f に指定する  
text3.setText(" ■2以上の自然数をキー入力し、[計算]ボ"); //テキストを表示する  
layout.addView(text3); //テキストビューをレイアウトに貼り付ける
```

```
text33 = new TextView(this); //テキストビュー型として実体化する
text33.setTextColors(Color.BLACK); //テキストを白色に指定する
text33.setTextSize(18f); //テキストの大きさを 15f に指定する
text33.setText(" タンを次々とタッチして、数字の変化を"); //テキストを表示する
layout.addView(text33); //テキストビューをレイアウトに貼り付ける
```

```
text14 = new TextView(this); //テキストビュー型として実体化する
text14.setTextColors(Color.BLACK); //テキストを白色に指定する
text14.setTextSize(18f); //テキストの大きさを 15f に指定する
text14.setText(" 観察してみましょう。"); //テキストを表示する
layout.addView(text14); //テキストビューをレイアウトに貼り付ける
```

```
text4 = new TextView(this); //テキストビュー型として実体化する
text4.setTextColors(Color.BLACK); //テキストを白色に指定する
text4.setTextSize(13f); //テキストの大きさを 15f に指定する
text4.setText(""); //テキストを表示する
layout.addView(text4); //テキストビューをレイアウトに貼り付ける
```

```
text5 = new TextView(this); //テキストビュー型として実体化する
text5.setTextColors(Color.BLACK); //テキストを白色に指定する
text5.setTextSize(18f); //テキストの大きさを 15f に指定する
text5.setText(" コラッツの問題 (3 x + 1 の問題) とは"); //テキストを表示する
layout.addView(text5); //テキストビューをレイアウトに貼り付ける
```

```
text6 = new TextView(this); //テキストビュー型として実体化する
text6.setTextColors(Color.BLACK); //テキストを白色に指定する
text6.setTextSize(17f); //テキストの大きさを 15f に指定する
text6.setText(" どんな自然数から始めても、その数が奇数"); //テキストを表示す
```

る

```
layout.addView(text6); //テキストビューをレイアウトに貼り付ける
```

```
text7 = new TextView(this); //テキストビュー型として実体化する
text7.setTextColors(Color.BLACK); //テキストを白色に指定する
text7.setTextSize(17f); //テキストの大きさを 15f に指定する
text7.setText(" だったら3倍して1を足し、偶数だったら"); //テキストを表示す
```

る

```

layout.addView(text7); //テキストビューをレイアウトに貼り付ける

text8 = new TextView(this); //テキストビュー型として実体化する
text8.setTextColor(Color.BLACK); //テキストを白色に指定する
text8.setTextSize(17f); //テキストの大きさを 15f に指定する
text8.setText(" 2で割ることを繰り返しおこなうと、必ず"); //テキストを表示す
る

layout.addView(text8); //テキストビューをレイアウトに貼り付ける

text9 = new TextView(this); //テキストビュー型として実体化する
text9.setTextColor(Color.BLACK); //テキストを白色に指定する
text9.setTextSize(17f); //テキストの大きさを 15f に指定する
text9.setText(" 1になるというものです (未解決問題で、)"); //テキストを表示す
る

layout.addView(text9); //テキストビューをレイアウトに貼り付ける

text92 = new TextView(this); //テキストビュー型として実体化する
text92.setTextColor(Color.BLACK); //テキストを白色に指定する
text92.setTextSize(17f); //テキストの大きさを 15f に指定する
text92.setText(" 未だ証明されていません。"); //テキストを表示する
layout.addView(text92); //テキストビューをレイアウトに貼り付ける

text10 = new TextView(this); //テキストビュー型として実体化する
text10.setTextColor(Color.BLACK); //テキストを白色に指定する
text10.setTextSize(17f); //テキストの大きさを 15f に指定する
text10.setText(" 例えば、11→34→17→52→26→13→40"); //テキストを表示する
layout.addView(text10); //テキストビューをレイアウトに貼り付ける

text11 = new TextView(this); //テキストビュー型として実体化する
text11.setTextColor(Color.BLACK); //テキストを白色に指定する
text11.setTextSize(17f); //テキストの大きさを 15f に指定する
text11.setText(" →20→10→5→16→8→4→2→1"); //テキストを表示する
layout.addView(text11); //テキストビューをレイアウトに貼り付ける

text15 = new TextView(this); //テキストビュー型として実体化する
text15.setTextColor(Color.BLACK); //テキストを白色に指定する

```

```
text15.setTextSize(13f); //テキストの大きさを 15f に指定する
text15.setText(""); //テキストを表示する
layout.addView(text15); //テキストビューをレイアウトに貼り付ける
```

```
edit1 = new EditText(this); //エディットテキスト型として実体化する
layout.addView(edit1); //エディットテキストをレイアウトに貼り付ける
```

```
button1 = new Button(this); //ボタン型として実体化する
button1.setTextSize(20f); //テキストの大きさを 15f に指定する
button1.setText(" [計算] "); //テキストを表示する
layout.addView(button1); //ボタンをレイアウトに貼り付ける
button1.setOnClickListener(new MyBtnClickAdapter()); //「計算」 ボタンにイベント処理
```

リスナーを設定する

```
text122 = new TextView(this); //テキストビュー型として実体化する
text122.setTextColor(Color.BLACK); //テキストを白色に指定する
text122.setTextSize(13f); //テキストの大きさを 15f に指定する
text122.setText(""); //テキストを表示する
layout.addView(text122); //テキストビューをレイアウトに貼り付ける
```

```
button2 = new Button(this); //ボタン型として実体化する
button2.setTextSize(20f); //テキストの大きさを 15f に指定する
button2.setText(" [初期化] "); //テキストを表示する
layout.addView(button2); //ボタンをレイアウトに貼り付ける
button2.setOnClickListener(new MyBtnClickAdapter2()); //「初期化」 ボタンにイベント処
```

理リスナーを設定する

```
text12 = new TextView(this); //テキストビュー型として実体化する
text12.setTextColor(Color.BLACK); //テキストを白色に指定する
text12.setTextSize(13f); //テキストの大きさを 15f に指定する
text12.setText(""); //テキストを表示する
layout.addView(text12); //テキストビューをレイアウトに貼り付ける
```

```
text13 = new TextView(this); //テキストビュー型として実体化する
text13.setTextColor(Color.BLACK); //テキストを白色に指定する
text13.setTextSize(16f); //テキストの大きさを 15f に指定する
```

```

text13.setText("      Copyright (C) K.Niwa 2021.3.13");    //テキストを表示する
layout.addView(text13);                                //テキストビューをレイアウトに貼り付ける

setContentView(layout);                                //★★★★レイアウトを画面に貼り付ける
}
//public void onCreate(Bundle savedInstanceState)

class MyBtnClickAdapter implements OnClickListener {    //「計算」ボタンのイベント処理リ
スナー
    public void onClick(View view) {    //「計算」ボタンをタッチしたとき

        flag++;                                    //初期化識別子に1を加える

        Editable ed = edit1.getText();    //edをEditable型として宣言し、edにエディットテ
キストに入力した文字を入力する
        try {
            n=Integer.parseInt(ed.toString()); //edを文字列型に変換し、さらに整数型に変換
して、nに代入する

        } catch (NumberFormatException e) {    //nが整数以外だったら、nを0にする
            n=0;
        }

        if (flag==1) {    //初期状態だったら、resにnを代入する
            res=n;
        }

        if (res%2==0) { //resが偶数だったら、2で割る
            res=res/2;
        }
        else if (res%2==1 && res!=1) { //resが奇数で1でなかったら、3倍して1を加える
            res=3*res+1;
        }

        text15.setTextSize(20f);                //テキストの大きさを20fにする
        text15.setTextColor(Color.RED);        //テキストの色を赤色にする

```

```

        text15.setText("    計算結果 : "+res);    //計算結果を表示する

        //Toast toast = Toast.makeText(getApplicationContext(), "計算結果 : "+res+"です。
", Toast.LENGTH_LONG);
        //toast.show();

    }//public void onClick(View view)
} //class MyBtnClickAdapter implements OnClickListener

class MyBtnClickAdapter2 implements OnClickListener {    //「初期化」ボタンのイベント処理
リスナー
    public void onClick(View view) {    //「初期化」ボタンをタッチしたとき
        flag=0;    //初期化識別子を0にする
        text15.setTextSize(15f);    //テキストの大きさを15fにする
        text15.setTextColor(Color.WHYTE);    //テキストの色を白色にする
        text15.setText("");    //テキストを表示する
        edit1.setText("");    //テキストビューをレイアウトに貼り付ける

    }//public void onClick(View view)
} //class MyBtnClickAdapter2 implements OnClickListener

} //public class MainActivity extends AppCompatActivity

```

[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" />
```

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