［１］MainActivity.java

```
/*
-----------------------------------------------------------------
                    コラッツの問題
              Android 4.4（Kit Kat）
        Copyright（C） K.Niwa 2019.12.13
-----------------------------------------------------------------
*/

package jp.kiyo.wuena.myp3xplus1;

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
    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); //リニアー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.setTextSize(16f);
//テキストの大きさを 15f に指定する
        text.setText("        【コラッツの問題（３ｘ＋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.setTextSize(15f);
//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.setTextSize(13f);
//テキストの大きさを 15f に指定する
        text2.setText("        ※まず、画面中央線の上側をタッチしてください...");
                        //テキストを表示する
        layout.addView(text2);
//テキストビューをレイアウトに貼り付ける

        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.setTextSize(13f);
//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]を次々と");
        //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

        text14 = new TextView(this);
//テキストビュー型として実体化する
        text14.setTextColor(Color.BLACK);                                   //テキスト
を白色に指定する
        text14.setTextSize(13f);
//テキストの大きさを 15f に指定する
        text14.setText("          タッチして、数字の変化を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.setTextColor(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.setTextColor(Color.BLACK);                                    //テキス
トを白色に指定する
        text5.setTextSize(13f);
//テキストの大きさを 15f に指定する
        text5.setText("        コラッツの問題とは");            //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.setTextColor(Color.BLACK);                               //テキス
トを白色に指定する
        text6.setTextSize(13f);
//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.setTextColor(Color.BLACK);                               //テキス
トを白色に指定する
        text7.setTextSize(13f);
//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２");
//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(13f);
//テキストの大きさを 15f に指定する
        text8.setText("          で割ることを繰り返しおこなうと、必ず１に");
//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(13f);
//テキストの大きさを 15f に指定する
        text9.setText("          なるというものです。");                //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

        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(13f);
//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(13f);
//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(15f);
        //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

        button2 = new Button(this);
//ボタン型として実体化する
        button2.setTextSize(15f);
        //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(13f);
//テキストの大きさを 15f に指定する
        text13.setText("                Copyright(C) K.Niwa 2019.12.1");
```

```
//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.WHITE);                                    //テキストの色を白色にする
                text15.setText("");       //テキストを表示する

            }//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〉
```