安卓应用开发——计算器的设计与实现

完成人：夏镇雄

学号：20172131133

完成时间：2019 年 10 月 3 日星期四

软件名称

简易计算器

软件简介

此次安卓软件开发的是一款简易计算器，因为有一些功能尚不完全，所以叫做简易计算器。使用的 IDE 是 Android Studio，UI 设计借鉴了苹果手机自带的计算器，并设计了横版竖版不同的 UI 界面，并配备了两个 Activity。

具体的实现计算器的计算功能算法方面，我是采用了中缀表达式转换成后缀表达式，然后用后缀表达式来计算出最终的结果的方法。在网上查找了有关方法之后，利用自己在之前学过的一些数据结构，比如栈，字符串组，StringBuffer 类型，通过读取按钮的内容，来对计算式进行计算。

由于算法的代码是自己打的，可能不是最优的，效率各方面还有待提升。一些逻辑关系也可以再进行完善，如果有更多时间的话。

算法设计流程

这个计算器我设计了 2 个界面，分别是横板的和竖版的。竖版的只有数字，四则运算，百分号和正负号；横板的在竖版的基础上增设了幂函数，对数函数，倒数，阶乘，三角函数，括号，随机数，数据存取键以及第二运算符盘。由于数据存取键和第二运算符盘实现起来比较复杂，所以本次开发未对数据存取键和第二运算符盘进行实现，只是在增设了 Button。

由于横板和竖版的计算算法是一样的，所以接下来不区分横竖版。

我将运算符分为一元运算符和二元运算符。

一元运算符有：正负号，百分号，对数函数，e，Π，随机数，倒数，阶乘，三角函数，二次函数，三次函数，平方根，立方根。

二元运算符有：加减乘除，幂函数，根函数，EE。

括号优先级最高，其次是一元运算符，最后是二元运算符。而二元运算符中，幂函数优先级最高，然后是根函数和 EE，接着是乘除，最后是加减。

用户输入数字时，可直接输入。

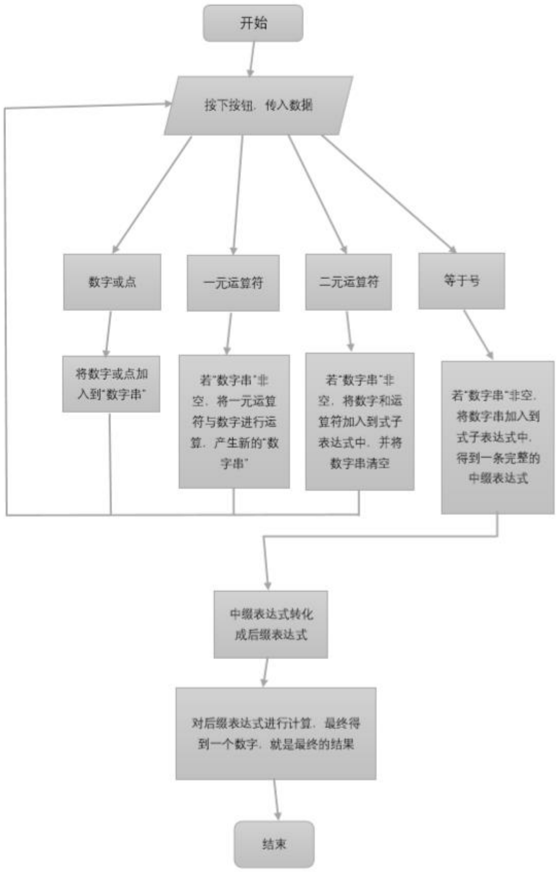
用户运用一元运算符时，需要先输入数字，再按下相应的运算符按钮。比如要算 cos(Π)，要先按Π，再按 cos 按钮。

多个一元运算符复合的话，按运算顺序先后按下相应的按钮。比如算 cos(ln(e))，顺序应该是 e，ln，cos。

二元运算符的话，按运算符的运算逻辑顺序输入，比如算 3^2，按钮顺序是 3，x^y，2。

计算器内部的设定是只有数字与运算符，所有一元运算符运算完毕后在计算器内部都会转化成一元运算符运算后的数字。所以在真正计算式子的时候，式子都是数字以及二元运算符的交替出现，并可能穿插着括号。

由上述的数学式子计算逻辑，可得到下面的流程图。

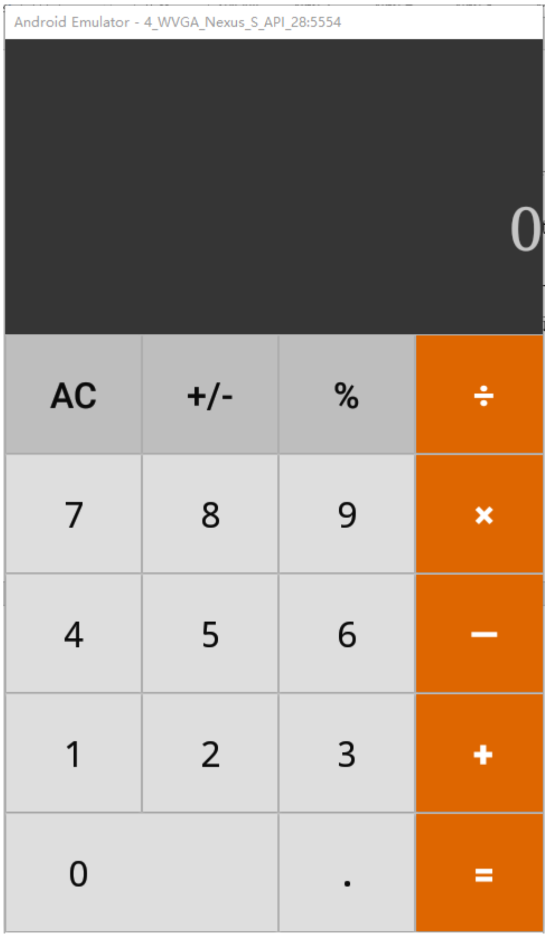

开始
按下按钮，传入数据
数字或点
一元运算符
二元运算符
等于号
将数字或点加入到"数字串"
若"数字串"非空，将一元运算符与数字进行运算，产生新的"数字串"
若"数字串"非空，将数字和运算符加入到式子表达式中，并将数字串清空
若"数字串"非空，将数字串加入到式子表达式中，得到一条完整的中缀表达式
中缀表达式转化成后缀表达式
对后缀表达式进行计算，最终得到一个数字，就是最终的结果
结束

UI 设计

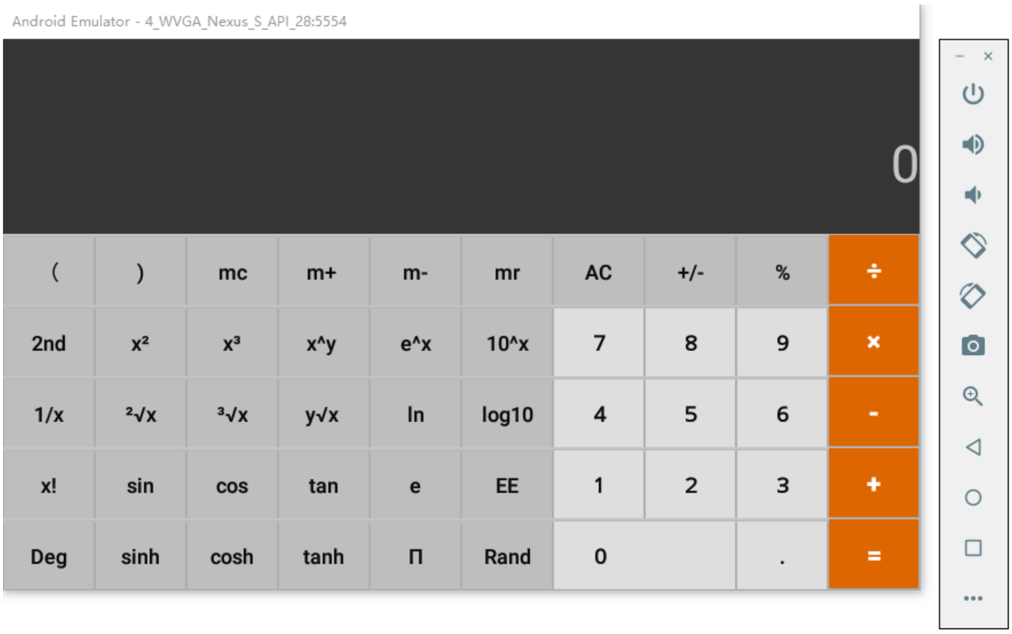
本次界面的一个缺点我先讲一下，就是我之前没学过网格分布（GridLayout），所以我用了表格分布(TableLayout)，设置的 TextView 和 Button 大小只能适应我的 AVD 的屏幕大小。我所采用的 AVD 是 4_WVGA_Nexus_S_API_28。因为当我同学告诉我网格分布更加适合已经太晚了，来不及再做修改。所以我就先用表格分布做，如果有时间的话，会采用更好的网格分布。

在显示界面，我是采用逻辑方法的先按数字，后按运算符。比如－8，要先按 8 再按＋/-。运算过程中不会显示运算符，只显示数字，只有当按等于号之后才会显示一整条式子。

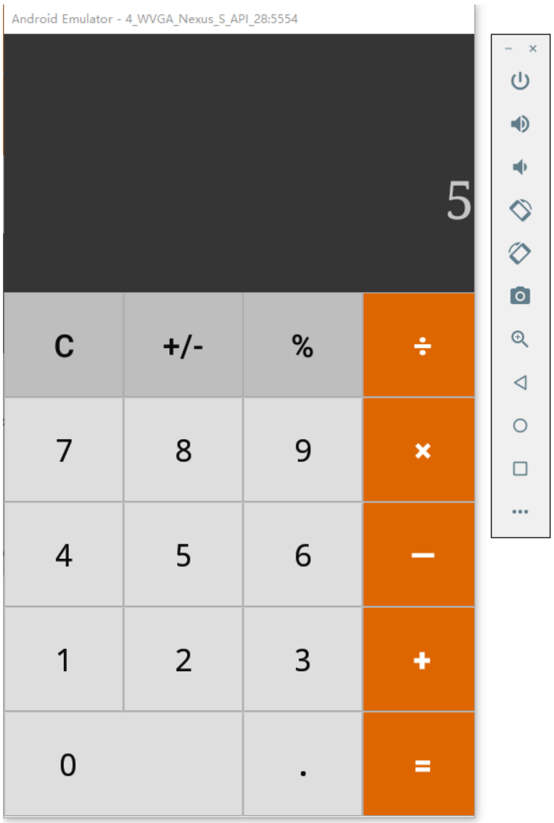
1. **竖板界面**



2. **横版界面**

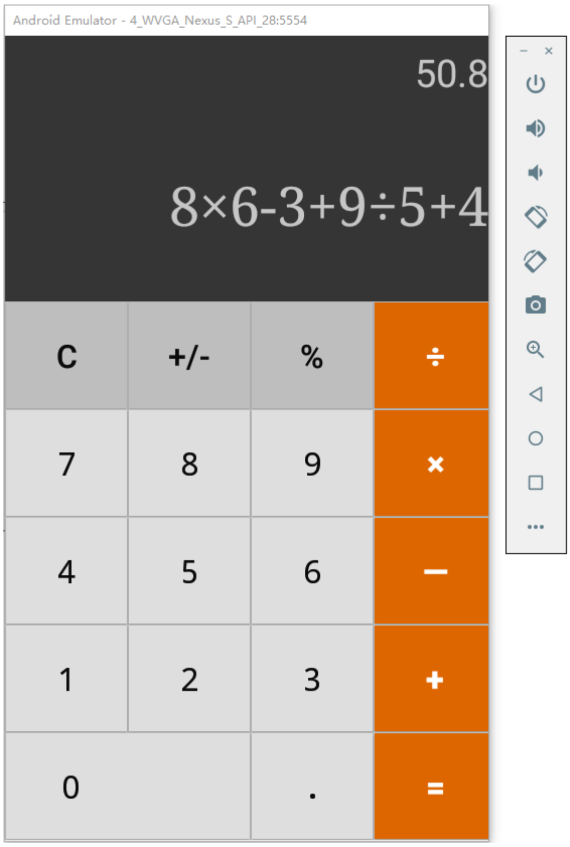

Android Emulator - 4_WVGA_Nexus_S_API_28:5554
0
( ) mc m+ m- mr AC +/- % ÷
2nd x² x³ x^y e^x 10^x 7 8 9 ×
1/x ²√x ³√x y√x ln log10 4 5 6 -
x! sin cos tan e EE 1 2 3 +
Deg sinh cosh tanh Π Rand 0 . =

3. 只显示数字



4. 横板界面简单的四则运算

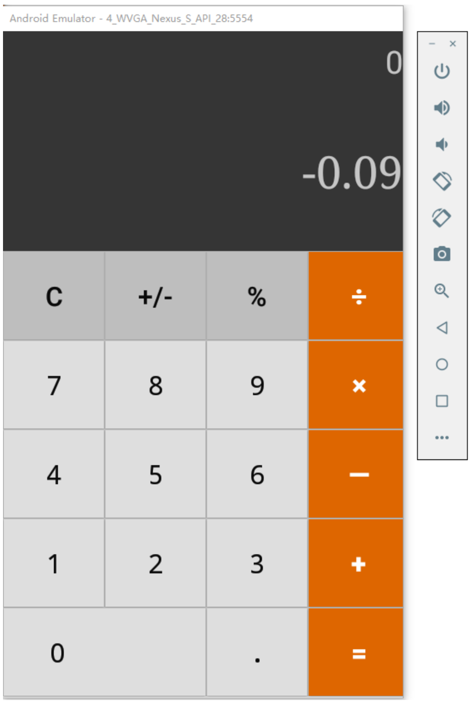
6.　横板的括号简单运算


Android Emulator - 4_WVGA_Nexus_S_API_28:5554

50.8

8×6-3+9÷5+4

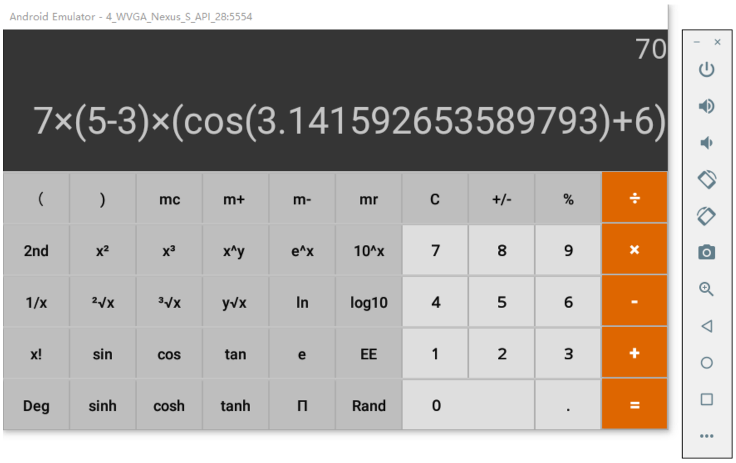
5.　按 9 之后按+/−再按%（不会显示百分号）


Android Emulator - 4_WVGA_Nexus_S_API_28:5554

0

-0.09

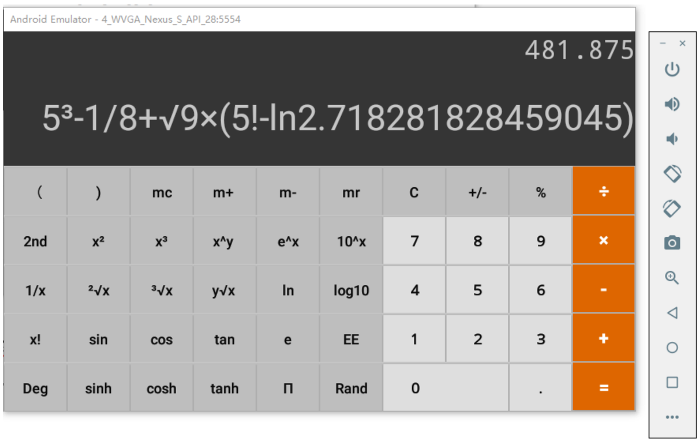
6.　横板的括号简单运算


Android Emulator - 4_WVGA_Nexus_S_API_28:5554
( ) mc m+ m- mr C +/- % ÷
2nd x² x³ x^y e^x 10^x 7 8 9 ×
1/x ²√x ³√x y√x ln log10 4 5 6 -
x! sin cos tan e EE 1 2 3 +
Deg sinh cosh tanh π Rand 0 . =

70

7×(5-3)×(cos(3.141592653589793)+6)

7. 幂，根，阶乘，倒数，对数运算


Android Emulator - 4_WVGA_Nexus_S_API_28:5554
( ) mc m+ m- mr C +/- % ÷
2nd x² x³ x^y e^x 10^x 7 8 9 ×
1/x ²√x ³√x y√x ln log10 4 5 6 -
x! sin cos tan e EE 1 2 3 +
Deg sinh cosh tanh π Rand 0 . =

481.875

5³-1/8+√9×(5!-ln2.718281828459045)

代码设计

1. 竖板界面布局代码 activity2_main.xml

```
<?xml version="1.0" encoding="utf-8"?>
<TableLayout
    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"
    android:stretchColumns="3"
    >
    <TextView
        android:id="@+id/textResult"
        android:layout_width="match_parent"
        android:layout_height="48dp"
        android:gravity="center_vertical|end"
        android:textSize="25sp"
        android:backgroundi="@color/colordi"
        android:textColori="@color/colorwhite"
        app:layout_constraintBottom_toBottomOf="parent"
        app:layout_constraintLeft_toLeftOf="parent"
        app:layout_constraintRight_toRightOf="parent"
        app:layout_constraintTop_toTopOf="parent" />

    <TextView
        android:background="@color/colordi"
        android:textColor="@color/colorwhite"
        android:id="@+id/textIn"
        android:text="0"
        android:typeface="serif"
        android:layout_width="match_parent"
        android:layout_height="128dp"
        android:gravity="center_vertical|end"
        android:hint="0"
        android:maxLines="3"
        android:scrollHorizontally="true"
        android:textSize="36sp" />
    <LinearLayout
        android:background="@color/colorwhite">
        <Button
            android:id="@+id/btn_AC"
            style="@style/SymbolBtnStyle"
            android:text="AC" />

        <Button
            android:id="@+id/btn_neg"
            style="@style/SymbolBtnStyle"
            android:text="+/-" />

        <Button
            android:id="@+id/btn_percent"
            style="@style/SymbolBtnStyle"
            android:text="%" />

        <Button
            android:id="@+id/btn_div"
            style="@style/RuleBtnStyle"
            android:text="÷" />
    </LinearLayout>

    <LinearLayout
```

```
        android:background="@color/colorwhite">
        <Button
            android:id="@+id/btn_7"
            style="@style/NumBtnStyle"
            android:text="7" />
        <Button
            style="@style/NumBtnStyle"
            android:id="@+id/btn_8"
            android:text="8"/>
        <Button
            style="@style/NumBtnStyle"
            android:id="@+id/btn_9"
            android:text="9"/>
        <Button
            style="@style/RuleBtnStyle"
            android:id="@+id/btn_mul"
            android:text="×"/>
    </LinearLayout>

    <LinearLayout
        android:background="@color/colorwhite">
        <Button
            android:id="@+id/btn_4"
            style="@style/NumBtnStyle"
            android:text="4" />
        <Button
            android:id="@+id/btn_5"
            style="@style/NumBtnStyle"
            android:text="5"/>
        <Button
            android:id="@+id/btn_6"
            style="@style/NumBtnStyle"
            android:text="6"/>
        <Button
            android:id="@+id/btn_sub"
            style="@style/RuleBtnStyle"
            android:text="—"/>
    </LinearLayout>

    <LinearLayout
        android:background="@color/colorwhite">
        <Button
            android:id="@+id/btn_1"
            style="@style/NumBtnStyle"
            android:text="1"/>
        <Button
            android:id="@+id/btn_2"
            style="@style/NumBtnStyle"
            android:text="2"/>
        <Button
            android:id="@+id/btn_3"
            style="@style/NumBtnStyle"
            android:text="3"/>
        <Button
            android:id="@+id/btn_add"
            style="@style/RuleBtnStyle"
            android:text="+"/>
    </LinearLayout>

    <LinearLayout
        android:background="@color/colorwhite"
        >

        <Button
            android:id="@+id/btn_0"
            style="@style/NumBtnStyle"
            android:layout_width="161dp"
            android:text="0        " />

        <Button
            android:id="@+id/btn_point"
            style="@style/NumBtnStyle"
            android:text="." />

        <Button
            android:id="@+id/btn_end"
            style="@style/RuleBtnStyle"
            android:text="=" />

    </LinearLayout>
</TableLayout>
```

2. 横板界面布局代码 layout.xml

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

```xml
    android:layout_width="match_parent"
    android:layout_height="match_parent"
    android:stretchColumns="3"
    style="@style/NoTitle"
    >

    <TextView
        android:typeface="monospace"
        android:id="@+id/textResult"
        android:layout_width="match_parent"
        android:layout_height="30dp"
        android:background="@color/colordi"
        android:gravity="center_vertical|end"
        android:textSize="22sp"
        android:textColor="@color/colorwhite"
        app:layout_constraintBottom_toBottomOf="parent"
        app:layout_constraintLeft_toLeftOf="parent"
        app:layout_constraintRight_toRightOf="parent"
        app:layout_constraintTop_toTopOf="parent"/>

    <TextView
        android:id="@+id/textIn"
        android:layout_width="match_parent"
        android:layout_height="83dp"
        android:background="@color/colordi"
        android:gravity="center_vertical|end"
        android:hint="0"
        android:maxLines="2"
        android:scrollHorizontally="true"
        android:text="0"
        android:textColor="@color/colorwhite"
        android:textSize="30sp" />
    <!--scrollHorizontally 设置文本超出 TextView 的宽度的情况下,是否出现横拉条-->

    <LinearLayout
        android:background="@color/colorwhite"
        >

        <Button
            android:id="@+id/btn_left"
            style="@style/HSymbolBtnStyle"
            android:text="（" />
        <Button
            android:id="@+id/btn_right"
            style="@style/HSymbolBtnStyle"
            android:text=")" />

        <Button
            android:textAllCaps="false"
            android:id="@+id/btn_mc"
            style="@style/HSymbolBtnStyle"
            android:text="mc" />
        <Button
            android:textAllCaps="false"
            android:id="@+id/btn_m_add"
            style="@style/HSymbolBtnStyle"
            android:text="m+ "/>
        <Button
            android:textAllCaps="false"
            android:id="@+id/btn_m_sub"
            style="@style/HSymbolBtnStyle"
            android:text="m-" />
        <Button
            android:textAllCaps="false"
            android:id="@+id/btn_mr"
            style="@style/HSymbolBtnStyle"
            android:text="mr" />
        <Button
            android:textAllCaps="false"
            android:id="@+id/btn_AC"
            style="@style/HSymbolBtnStyle"
            android:text="AC" />

        <Button
            android:id="@+id/btn_neg"
            style="@style/HSymbolBtnStyle"
            android:text="+/-" />
        <Button
            android:id="@+id/btn_percent"
            style="@style/HSymbolBtnStyle"

            android:text="%" />

        <Button
            android:id="@+id/btn_div"
            style="@style/HRuleBtnStyle"
            android:text="÷" />
    </LinearLayout>

    <LinearLayout
        android:background="@color/colorwhite">

        <Button
            android:id="@+id/btn_second"
            style="@style/HSymbolBtnStyle"
            android:text="2nd" />

        <Button
            android:id="@+id/btn_square"
            style="@style/HSymbolBtnStyle"
            android:text="x²" />

        <Button
            android:id="@+id/btn_cube"
            style="@style/HSymbolBtnStyle"
            android:text="x³" />
        <Button
            android:id="@+id/btn_power"
            style="@style/HSymbolBtnStyle"
            android:text="x^y "/>
        <Button
            android:id="@+id/btn_epower"
            style="@style/HSymbolBtnStyle"
            android:text="e^x" />
        <Button
            android:id="@+id/btn_10power"
            style="@style/HSymbolBtnStyle"
            android:text="10^x" />
        <Button
            android:id="@+id/btn_7"
            style="@style/HNumBtnStyle"
            android:text="7" />

        <Button
            android:id="@+id/btn_8"
            style="@style/HNumBtnStyle"
            android:text="8" />

        <Button
            android:id="@+id/btn_9"
```

```xml
            style="@style/HNumBtnStyle"
            android:text="9" />

        <Button
            android:id="@+id/btn_mul"
            style="@style/HRuleBtnStyle"
            android:text="×" />
    </LinearLayout>

    <LinearLayout
        android:background="@color/colorwhite">

        <Button
            android:id="@+id/btn_rec"
            style="@style/HSymbolBtnStyle"
            android:text="1/x" />

        <Button
            android:id="@+id/btn_sqrt"
            style="@style/HSymbolBtnStyle"
            android:text="²√x" />

        <Button
            android:id="@+id/btn_cube_root"
            style="@style/HSymbolBtnStyle"
            android:text="³√x" />
        <Button
            android:id="@+id/btn_y_root"
            style="@style/HSymbolBtnStyle"
            android:text="y√x "/>
        <Button
            android:id="@+id/btn_ln"
            style="@style/HSymbolBtnStyle"
            android:text="ln" />
        <Button
            android:id="@+id/btn_log10"
            style="@style/HSymbolBtnStyle"
            android:text="log10" />
        <Button
            android:id="@+id/btn_4"
            style="@style/HNumBtnStyle"
            android:text="4" />

        <Button
```

```
            android:id="@+id/btn_5"
            style="@style/HNumBtnStyle"
            android:text="5" />

        <Button
            android:id="@+id/btn_6"
            style="@style/HNumBtnStyle"
            android:text="6" />

        <Button
            android:id="@+id/btn_sub"
            style="@style/HRuleBtnStyle"
            android:text="-" />
    </LinearLayout>

    <LinearLayout
        android:background="@color/colorwhite">

        <Button
            android:id="@+id/btn_Factorial"
            style="@style/HSymbolBtnStyle"
            android:text="x!" />

        <Button
            android:id="@+id/btn_sin"
            style="@style/HSymbolBtnStyle"
            android:text="sin" />

        <Button
            android:id="@+id/btn_cos"
            style="@style/HSymbolBtnStyle"
            android:text="cos" />
        <Button
            android:id="@+id/btn_tan"
            style="@style/HSymbolBtnStyle"
            android:text="tan"/>

        <Button
            android:id="@+id/btn_e"
            style="@style/HSymbolBtnStyle"
            android:text="e" />
        <Button
            android:id="@+id/btn_EE"
            style="@style/HSymbolBtnStyle"
            android:text="EE" />
        <Button
            android:id="@+id/btn_1"
            style="@style/HNumBtnStyle"
            android:text="1" />

        <Button
            android:id="@+id/btn_2"
            style="@style/HNumBtnStyle"
            android:text="2" />

        <Button
            android:id="@+id/btn_3"
            style="@style/HNumBtnStyle"
            android:text="3" />

        <Button
            android:id="@+id/btn_add"
            style="@style/HRuleBtnStyle"
            android:text="+" />
    </LinearLayout>

    <LinearLayout
        android:background="@color/colorwhite">

        <Button
            android:id="@+id/btn_Deg"
            style="@style/HSymbolBtnStyle"
            android:text="Deg" />

        <Button
            android:id="@+id/btn_sinh"
            style="@style/HSymbolBtnStyle"
            android:text="sinh" />

        <Button
            android:id="@+id/btn_cosh"
            style="@style/HSymbolBtnStyle"
            android:text="cosh" />
        <Button
            android:id="@+id/btn_tanh"
            style="@style/HSymbolBtnStyle"
            android:text="tanh"/>
        <Button
```

```
            android:id="@+id/btn_pai"
            style="@style/HSymbolBtnStyle"
            android:text="Π" />
        <Button
            android:id="@+id/btn_Rand"
            style="@style/HSymbolBtnStyle"
            android:text="Rand" />
        <Button
            android:id="@+id/btn_0"
            style="@style/HNumBtnStyle"
            android:layout_width="105.5dp"
            android:layout_height="40dp"
            android:text="0       " />
        <Button
            android:id="@+id/btn_point"
            style="@style/HNumBtnStyle"
            android:text="." />

        <Button
            android:id="@+id/btn_end"
            style="@style/HRuleBtnStyle"
            android:text="=" />
    </LinearLayout>
</TableLayout>
```

3.　竖版界面 java 文件 Main2Activity.java

```
package com.exaimple.xiong.counter;

import android.content.Intent;
import android.content.res.Configuration;
import android.support.v7.app.AppCompatActivity;
import android.os.Bundle;
import android.view.View;
import android.widget.Button;
import android.widget.TextView;
import java.util.ArrayList;
import android.util.Log;

public class Main2Activity extends AppCompatActivity  implements
View.OnClickListeneri
{
    private Button myBtn_1;
    private Button myBtn_2;
    private Button myBtn_3;
    private Button myBtn_4;
    private Button myBtn_5;
    private Button myBtn_6;
    private Button myBtn_7;
    private Button myBtn_8;
    private Button myBtn_9;
    private Button myBtn_0;
    private Button myBtn_sub;
    private Button myBtn_add;
    private Button myBtn_AC;
    private Button myBtn_neg;
    private Button myBtn_percent;
    private Button myBtn_point;
    private Button myBtn_mul;
    private Button myBtn_div;
    private Button myBtin_end;

    private TextView TextIn;
    private TextView TextResult;
    private StringBuffer Num;
    private int i=0;
    private ArrayList<String> Formula;

    @Override
    protected void onCreate(Bundle savedInstanceState) {

        super.onCreate(savedInstanceState);
        setContentView(R.layout.activity2_main);
        Configuration mConfiguration =
this.getResources().getConfiguration(); //获取设置的配置信息
        int ori = mConfiguration.orientation;
        if (ori == mConfiguration.ORIENTATION_LANDSCAPE)//如果是横屏
        {
            Intent intent = new Intent();
            intent.setClass(Main2Activity.this, MainActivity.class);
//从 Main2Activity 转到 MainActivity
            startActivity(intent);  //开始切换 Activity
        }

        //通过 findViewById()初始化控件
        myBtn_1 = (Button) findViewById(R.id.btn_1);
        myBtn_2 = (Button) findViewById(R.id.btn_2);
        myBtn_3 = (Button) findViewById(R.id.btn_3);
        myBtn_4 = (Button) findViewById(R.id.btn_4);
        myBtn_5 = (Button) findViewById(R.id.btn_5);
        myBtn_6 = (Button) findViewById(R.id.btn_6);
        myBtn_7 = (Button) findViewById(R.id.btn_7);
        myBtn_8 = (Button) findViewById(R.id.btn_8);
        myBtn_9 = (Button) findViewById(R.id.btn_9);
        myBtn_0 = (Button) findViewById(R.id.btn_0);
        myBtn_sub = (Button) findViewById(R.id.btn_sub);
        myBtn_add = (Button) findViewById(R.id.btn_add);
        myBtn_div = (Button) findViewById(R.id.btn_div);
        myBtn_end = (Button) findViewById(R.id.btn_end);
        myBtn_mul = (Button) findViewById(R.id.btn_mul);
        myBtn_neg = (Button) findViewById(R.id.btn_neg);
        myBtn_percent = (Button) findViewById(R.id.btn_percent);
        myBtn_point = (Button) findViewById(R.id.btn_point);
        myBtn_AC = (Button) findViewById(R.id.btn_AC);

        TextIn=(TextView) findViewById(R.id.textIn);//输入的字符串
        TextResult=(TextView) findViewById(R.id.textResult);//显示结
果的字符串

        myBtn_1.setOnClickListener(this);
        myBtn_2.setOnClickListener(this);
        myBtn_3.setOnClickListener(this);
        myBtn_4.setOnClickListener(this);
        myBtn_5.setOnClickListener(this);
        myBtn_6.setOnClickListener(this);
        myBtn_7.setOnClickListener(this);
        myBtn_8.setOnClickListener(this);
        myBtn_9.setOnClickListener(this);
        myBtn_0.setOnClickListener(this);
        myBtn_add.setOnClickListener(this);
        myBtn_AC.setOnClickListener(this);
        myBtn_point.setOnClickListener(this);
        myBtn_percent.setOnClickListener(this);
        myBtn_sub.setOnClickListener(this);
        myBtn_mul.setOnClickListener(this);
        myBtn_div.setOnClickListener(this);
        myBtn_neg.setOnClickListener(this);
        myBtn_end.setOnClickListener(this);

        Num=new StringBuffer();
        Formula = new ArrayList<String>();
    }
    @Override
    public void onClick(View v)
    {
        switch(v.getId())
        {
            case R.id.btn_1:
                Num.append("1");
                TextIn.setText(Num);
                break;
            case R.id.btn_2:
                Num.append("2");
                TextIn.setText(Num);
                break;
            case R.id.btn_3:
                Num.append("3");
                TextIn.setText(Num);
                break;
            case R.id.btn_4:
                Num.append("4");
                TextIn.setText(Num);
                break;
            case R.id.btn_5:
                Num.append("5");
                TextIn.setText(Num);
                break;
            case R.id.btn_6:
                Num.append("6");
                TextIn.setText(Num);
                break;
            case R.id.btn_7:
                Num.append("7");
                TextIn.setText(Num);
                break;
            case R.id.btn_8:
                Num.append("8");
                TextIn.setText(Num);
                break;
            case R.id.btn_9:
                Num.append("9");
                TextIn.setText(Num);
                break;
            case R.id.btn_0:
                if (Num.length()!=0)
                    Num.append("0");
                TextIn.setText(Num);
                break;
            case R.id.btn_AC:
                if (Num.length()==0)
                    Formula.clear();
                else {
                    Num.delete(0, Num.length());
                    TextIn.setText("0");
                    TextResult.setText("0");
                }
                break;
            case R.id.btn_add:
                if(Num.length()!=0)
                {
                    Formula.add(i++, Num.toString());
                    Formula.add(i++,"add");
                    Num.delete(0, Num.length());
                }
                break;
            case R.id.btn_sub:
                if(Num.length()!=0)
                {
                    Formula.add(i++, Num.toString());
                    Formula.add(i++,"sub");
                    Num.delete(0, Num.length());
                }
                break;
            case R.id.btn_percent:
                if(Num.length()!=0)
                {
                    double num=Double.valueOf(Num.toString());
                    num=num/100;
                    String NUM=String.valueOf(num);
                    Num=new StringBuffer(NUM);
                    TextIn.setText(Num);
                }
                break;
            case R.id.btn_point:
                Num.append(".");
                TextIn.setText(Num);
                break;
            case R.id.btn_neg:
                if(Num.length()!=0)
```

```
                {
                    String NUM;
                    double num = Double.valueOf(Num.toString());
                    if (num-(int)num==0) {
                        int intnum=-(int)num;
                        NUM = String.valueOf(intnum);
                    }
                    else{
                        num = -num ;
                        NUM = String.valueOf(num);
                    }

                    Num = new StringBuffer(NUM);
                    TextIn.setText(Num);
                }
                break;
            case R.id.btn_mul:
                if(Num.length()!=0)
                {
                    Formula.add(i++, Num.toString());
                    Formula.add(i++,"mul");
                    Num.delete(0, Num.length());
                }
                break;
            case R.id.btn_div:
                if(Num.length()!=0)
                {
                    Formula.add(i++, Num.toString());
                    Formula.add(i++,"div");
                    Num.delete(0, Num.length());
                }
                break;
            case R.id.btn_end:
                if (Num.length()!=0) {
                    Formula.add(i++, Num.toString());
                    for (int p = 0; p < i; p++) {
                        Log.v(Formula.get(p), "式子");
                    }
                    ArrayList<String> Suffix = GetSuffix(Formula);
                    for (int p = 0; p < i; p++) {
                        Log.v(Suffix.get(p), "后缀表达式");
                    }
                    Num = new StringBuffer(CalculaSuffix(Suffix));

                    double num = Double.valueOf(Num.toString());
                    String NUM;
                    if (num - (int) num == 0) {
                        int intnum = (int) num;
                        NUM = String.valueOf(intnum);
                    } else {
                        NUM = String.valueOf(num);
                    }
                    Num = new StringBuffer(NUM);
                    TextIn.setText(Show(Formula));
                    TextResult.setText(Num);

                    Formula.clear();
                    i = 0;
                    break;
                }

        }
        //如果没有数字，那么 AC 键是 Clear
        if (Num.length()==0)
            myBtn_AC.setText("AC");
        else myBtn_AC.setText("C");
    }

    public ArrayList<String> GetSuffix(ArrayList<String> Formula)
    {
        java.util.Stack<String> operatorStack = new
java.util.Stack<>();
        ArrayList<String> houzhui=new ArrayList<String>();
        int index=0;

        for (int j=0;j<i;j++)
        {
            if (j%2==0)
            {
                houzhui.add(index++,Formula.get(j));
            }
            else
            {
                if (operatorStack.empty())
                {
                    operatorStack.push(Formula.get(j));
                }
                else
                {
                    switch (Formula.get(j))
                    {
                        //当遇到加号和减号时，符号栈内全部符号输出，
压入当前符号
                        case "sub":
                        case "add":
                            while (!operatorStack.empty())
                            {

houzhui.add(index++,operatorStack.pop());
                            }
                            operatorStack.push(Formula.get(j));
                            break;
                        //当遇到乘号与除号，压入符号栈
                        case "mul":
                        case "div":
                            operatorStack.push(Formula.get(j));
                            break;
                    }
                }
            }
        }

        while (!operatorStack.empty())
        {
            houzhui.add(index++,operatorStack.pop());
        }
        return houzhui;
    }

    public String CalculaSuffix(ArrayList<String> Suffix)
    {
        java.util.Stack<String> numStack = new java.util.Stack<>();
        double num1;
        double num2;
        double num3;
        String NUM3;
        for (int j=0;j<i;j++)
        {
            switch (Suffix.get(j))
            {
                case "add":
                    num1 = Double.valueOf(numStack.pop());
                    num2 = Double.valueOf(numStack.pop());
                    num3 = num1 +num2;
                    NUM3=String.valueOf(num3);
                    numStack.push(NUM3);
                    break;
                case "sub":
                    num1 = Double.valueOf(numStack.pop());
                    num2 = Double.valueOf(numStack.pop());
                    num3 = num2-num1;
                    NUM3=String.valueOf(num3);
                    numStack.push(NUM3);
                    break;
                case "mul":
                    num1 = Double.valueOf(numStack.pop());
                    num2 = Double.valueOf(numStack.pop());
                    num3 = num1*num2;
                    NUM3=String.valueOf(num3);
                    numStack.push(NUM3);
                    break;
                case "div":
                    num1 = Double.valueOf(numStack.pop());
                    num2 = Double.valueOf(numStack.pop());
                    num3 = num2/num1;
                    NUM3=String.valueOf(num3);
                    numStack.push(NUM3);
                    break;
                default:
                    numStack.push(Suffix.get(j));
                    break;
            }
        }
        return numStack.pop();
    }
    public String Show(ArrayList<String> Formula)
    {
        StringBuffer Show=new StringBuffer();
        for (int j=0;j<i;j++)
        {
            switch (Formula.get(j)) {
                case "mul":
                    Show.append("×");
                    break;
                case "div":
                    Show.append("÷");
                    break;
                case "add":
                    Show.append("+");
                    break;
                case "sub":
                    Show.append("-");
                    break;
                default:
                    Show.append(Formula.get(j));
                    break;
            }
        }
        return Show.toString();
    }
}
```

4. 横版界面 java 文件 MainActivity.java

```
package com.example.xiong.counter;

import android.content.Intent;
import android.support.v7.app.AppCompatActivity;
import android.os.Bundle;
import android.view.View;
import android.widget.Button;
import android.widget.TextView;
import java.util.ArrayList;
import android.util.Log;
import android.content.res.Configuration;
public class MainActivity extends AppCompatActivity  implements
View.OnClickListener
{
    private Button myBtn_1;
    private Button myBtn_2;
    private Button myBtn_3;
    private Button myBtn_4;
    private Button myBtn_5;
    private Button myBtn_6;
    private Button myBtn_7;
    private Button myBtn_8;
    private Button myBtn_9;
    private Button myBtn_0;
    private Button myBtn_sub;
    private Button myBtn_add;
    private Button myBtn_AC;
    private Button myBtn_neg;
    private Button myBtn_percent;
    private Button myBtn_point;
    private Button myBtn_mul;
    private Button myBtn_div;
    private Button myBtn_end;
    private Button myBtn_left;
    private Button myBtn_right;
    private Button myBtn_mc;
    private Button myBtn_m_add;
    private Button myBtn_m_sub;
    private Button myBtn_mr;
    private Button myBtn_second;
    private Button myBtn_square;
    private Button myBtn_cube;
    private Button myBtn_power;
    private Button myBtn_epower;
    private Button myBtn_10power;
    private Button myBtn_rec;
    private Button myBtn_sqrt;
    private Button myBtn_cube_root;
    private Button myBtn_y_root;
    private Button myBtn_ln;
    private Button myBtn_log10;
    private Button myBtn_Factorial;
    private Button myBtn_sin;
    private Button myBtn_cos;
    private Button myBtn_tan;
    private Button myBtn_e;
    private Button myBtn_EE;
    private Button myBtn_Deg;
    private Button myBtn_sinh;
    private Button myBtn_cosh;
    private Button myBtn_tanh;
    private Button myBtn_pai;
    private Button myBtn_Rand;

    private TextView Text1;
    private TextView Text2;
    private StringBuffer Num;
    private int i=0;
    private ArrayList<String> Formula;
    private StringBuffer result;
    private StringBuffer ShowFormula;
    private boolean combination;
    private int bracketsExist;

    @Override
    protected void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);
        setContentView(R.layout.layout);
        Configuration mConfiguration =
this.getResources().getConfiguration(); //获取设置的配置信息
        int ori = mConfiguration.orientation; //获取屏幕方向
        if (ori == mConfiguration.ORIENTATION_PORTRAIT)//竖屏
        {
            Intent intent = new Intent();
            intent.setClass(MainActivity.this,
Main2Activity.class);//从 MainActivity 转到 Main2Activity
            startActivity(intent);  //开始切换 Activity
        }

        //通过 findViewById()初始化控件
        myBtn_1 = (Button) findViewById(R.id.btn_1);
        myBtn_2 = (Button) findViewById(R.id.btn_2);
        myBtn_3 = (Button) findViewById(R.id.btn_3);
        myBtn_4 = (Button) findViewById(R.id.btn_4);
        myBtn_5 = (Button) findViewById(R.id.btn_5);
        myBtn_6 = (Button) findViewById(R.id.btn_6);
        myBtn_7 = (Button) findViewById(R.id.btn_7);
        myBtn_8 = (Button) findViewById(R.id.btn_8);
        myBtn_9 = (Button) findViewById(R.id.btn_9);
        myBtn_0 = (Button) findViewById(R.id.btn_0);
        myBtn_sub = (Button) findViewById(R.id.btn_sub);
        myBtn_add = (Button) findViewById(R.id.btn_add);
        myBtn_div = (Button) findViewById(R.id.btn_div);
        myBtn_end = (Button) findViewById(R.id.btn_end);
        myBtn_mul = (Button) findViewById(R.id.btn_mul);
        myBtn_neg = (Button) findViewById(R.id.btn_neg);
        myBtn_percent = (Button) findViewById(R.id.btn_percent);
        myBtn_point = (Button) findViewById(R.id.btn_point);
        myBtn_AC = (Button) findViewById(R.id.btn_AC);

        myBtn_left=(Button) findViewById(R.id.btn_left);
        myBtn_right=(Button) findViewById(R.id.btn_right);
        myBtn_mc=(Button) findViewById(R.id.btn_mc);
        myBtn_mr=(Button) findViewById(R.id.btn_mr);
        myBtn_second=(Button) findViewById(R.id.btn_second);
        myBtn_square=(Button) findViewById(R.id.btn_square);
        myBtn_cube=(Button) findViewById(R.id.btn_cube);
        myBtn_power=(Button) findViewById(R.id.btn_power);
        myBtn_epower=(Button) findViewById(R.id.btn_epower);
        myBtn_10power=(Button) findViewById(R.id.btn_10power);
        myBtn_rec=(Button) findViewById(R.id.btn_rec);
        myBtn_sqrt=(Button) findViewById(R.id.btn_sqrt);
        myBtn_cube_root=(Button) findViewById(R.id.btn_cube_root);
        myBtn_y_root=(Button) findViewById(R.id.btn_y_root);
        myBtn_ln=(Button) findViewById(R.id.btn_ln);
        myBtn_log10=(Button) findViewById(R.id.btn_log10);
        myBtn_Factorial=(Button) findViewById(R.id.btn_Factorial);
        myBtn_sin=(Button) findViewById(R.id.btn_sin);
        myBtn_cos=(Button) findViewById(R.id.btn_cos);
        myBtn_tan=(Button) findViewById(R.id.btn_tan);
        myBtn_e=(Button) findViewById(R.id.btn_e);
        myBtn_EE=(Button) findViewById(R.id.btn_EE);
        myBtn_Deg=(Button) findViewById(R.id.btn_Deg);
        myBtn_sinh=(Button) findViewById(R.id.btn_sinh);
        myBtn_cosh=(Button) findViewById(R.id.btn_cosh);
        myBtn_tanh=(Button) findViewById(R.id.btn_tanh);
        myBtn_pai=(Button) findViewById(R.id.btn_pai);
        myBtn_Rand=(Button) findViewById(R.id.btn_Rand);

        Text1=(TextView) findViewById(R.id.textIn);
        Text2=(TextView) findViewById(R.id.textResult);

        myBtn_1.setOnClickListener(this);
        myBtn_2.setOnClickListener(this);
        myBtn_3.setOnClickListener(this);
        myBtn_4.setOnClickListener(this);
        myBtn_5.setOnClickListener(this);
        myBtn_6.setOnClickListener(this);
        myBtn_7.setOnClickListener(this);
        myBtn_8.setOnClickListener(this);
        myBtn_9.setOnClickListener(this);
        myBtn_0.setOnClickListener(this);
        myBtn_add.setOnClickListener(this);
        myBtn_AC.setOnClickListener(this);
        myBtn_point.setOnClickListener(this);
        myBtn_percent.setOnClickListener(this);
        myBtn_sub.setOnClickListener(this);
        myBtn_mul.setOnClickListener(this);
        myBtn_div.setOnClickListener(this);
        myBtn_neg.setOnClickListener(this);
        myBtn_end.setOnClickListener(this);

        myBtn_left.setOnClickListener(this);
        myBtn_right.setOnClickListener(this);
        myBtn_mc.setOnClickListener(this);
        myBtn_square.setOnClickListener(this);
        myBtn_cube.setOnClickListener(this);
        myBtn_power.setOnClickListener(this);
        myBtn_epower.setOnClickListener(this);
        myBtn_10power.setOnClickListener(this);
        myBtn_rec.setOnClickListener(this);
        myBtn_sqrt.setOnClickListener(this);
        myBtn_cube_root.setOnClickListener(this);
        myBtn_y_root.setOnClickListener(this);
        myBtn_ln.setOnClickListener(this);
        myBtn_log10.setOnClickListener(this);
        myBtn_Factorial.setOnClickListener(this);
        myBtn_sin.setOnClickListener(this);
        myBtn_cos.setOnClickListener(this);
        myBtn_tan.setOnClickListener(this);
        myBtn_e.setOnClickListener(this);
        myBtn_EE.setOnClickListener(this);
        myBtn_Deg.setOnClickListener(this);
        myBtn_sinh.setOnClickListener(this);
        myBtn_cosh.setOnClickListener(this);
        myBtn_tanh.setOnClickListener(this);
        myBtn_pai.setOnClickListener(this);
        myBtn_Rand.setOnClickListener(this);

        //初始化
        Num=new StringBuffer();
        Formula = new ArrayList<String>();
        ShowFormula=new StringBuffer();
        combination=false;
        bracketsExist=0;
    }

    @Override
    public void onClick(View v)
    {
        switch(v.getId())
        {
            case R.id.btn_1:
                Num.append("1");
                Text1.setText(Num);
                break;
            case R.id.btn_2:
                Num.append("2");
                Text1.setText(Num);
                break;
            case R.id.btn_3:
                Num.append("3");
                Text1.setText(Num);
                break;
            case R.id.btn_4:
                Num.append("4");
                Text1.setText(Num);
                break;
            case R.id.btn_5:
                Num.append("5");
                Text1.setText(Num);
                break;
            case R.id.btn_6:
                Num.append("6");
                Text1.setText(Num);
                break;
            case R.id.btn_7:
                Num.append("7");
                Text1.setText(Num);
                break;
            case R.id.btn_8:
                Num.append("8");
                Text1.setText(Num);
                break;
            case R.id.btn_9:
                Num.append("9");
                Text1.setText(Num);
                break;
            case R.id.btn_0:
                if(Num.length()!=0)
                    Num.append("0");
                Text1.setText(Num);
                break;
            case R.id.btn_AC:
                if (Num.length()==0){
```

```
                Formula.clear();
                i=0;
                ShowFormula.delete(0,ShowFormula.length());
            }
            else {
                Num.delete(0, Num.length());
                Text1.setText("0");
                i=0;
            }
            break;
        case R.id.btn_add:
            if(Num.length()!=0)
            {
                Formula.add(i++, Num.toString());
                if (!combination)
                    ShowFormula.append(Num);
                Num.delete(0, Num.length());
            }
            Formula.add(i++,"add");
            ShowFormula.append("+");
            combination=false;
            break;
        case R.id.btn_sub:

            if(Num.length()!=0)
            {
                Formula.add(i++, Num.toString());
                if (!combination)
                    ShowFormula.append(Num);
                Num.delete(0, Num.length());
            }
            Formula.add(i++,"sub");
            ShowFormula.append("-");
            combination=false;
            break;
        case R.id.btn_percent:
            if(Num.length()!=0)
            {
                double num=Double.valueOf(Num.toString());
                num=num/100;
                String NUM=String.valueOf(num);
                Num=new StringBuffer(NUM);
                Text1.setText(Num);
            }
            break;
        case R.id.btn_point:
            Num.append(".");
            Text1.setText(Num);
            break;
        case R.id.btn_neg:
            if(Num.length()!=0)
            {
                String NUM;
                double num = Double.valueOf(Num.toString());
                if (num-(int)num==0) {
                    int intnum=-(int)num;
                    NUM = String.valueOf(intnum);
                }
                else{
                    num = -num ;
                    NUM = String.valueOf(num);
                }

                Num = new StringBuffer(NUM);
                Text1.setText(Num);
            }
            break;
        case R.id.btn_mul:
            if(Num.length()!=0)
            {
                Formula.add(i++, Num.toString());
                if (!combination)
                    ShowFormula.append(Num);
                Num.delete(0, Num.length());
            }
            Formula.add(i++,"mul");
            ShowFormula.append("×");
            combination=false;
            break;
        case R.id.btn_div:
            if(Num.length()!=0)
            {
                Formula.add(i++, Num.toString());
                if (!combination)
                    ShowFormula.append(Num);
                Num.delete(0, Num.length());
            }
            Formula.add(i++,"div");
            ShowFormula.append("÷");
            combination=false;
            break;
        case R.id.btn_end:
            if (!combination)
                ShowFormula.append(Num);
            if (Num.length()!=0)
                Formula.add(i++, Num.toString());

            ArrayList<String> Suffix=GetSuffix(Formula);


            Num=new StringBuffer(CalculaSuffix(Suffix));


            double num = Double.valueOf(Num.toString());
            String NUM;
            if (num-(int)num==0)
            {
                int intnum=(int)num;
                NUM = String.valueOf(intnum);
            }
            else
            {
                NUM = String.valueOf(num);
            }
            Num = new StringBuffer(NUM);
            Text1.setText(ShowFormula);
            Text2.setText(Num);
            //初始化


            Formula = new ArrayList<String>();
            ShowFormula=new StringBuffer();
            combination=false;
            bracketsExist=0;
            i=0;
            break;
        case R.id.btn_left:
            Formula.add(i++,"(");
            ShowFormula.append("(");
            combination=false;
            break;
        case R.id.btn_right:
            Formula.add(i++,Num.toString());
            Formula.add(i++,")");
            if (!combination)
```

```
                ShowFormula.append(Num);
            ShowFormula.append(")");
            Num.delete(0,Num.length());
            break;
        case R.id.btn_square:
            if(Num.length()!=0)
            {
                if (!combination)
                    ShowFormula.append(Num);
                ShowFormula.append("²");
                combination=true;
                num=Double.valueOf(Num.toString());
                num=Math.pow(num,2);
                NUM=String.valueOf(num);
                Num=new StringBuffer(NUM);
                Text1.setText(Num);
            }
            else Text1.setText("请先输入数字");
            break;
        case R.id.btn_cube:
            if(Num.length()!=0)
            {
                if (!combination)
                    ShowFormula.append(Num);
                ShowFormula.append("³");
                combination=true;
                num=Double.valueOf(Num.toString());
                num=Math.pow(num,3);
                NUM=String.valueOf(num);
                Num=new StringBuffer(NUM);
                Text1.setText(Num);
            }
            break;
        case R.id.btn_power:
            if(Num.length()!=0)
            {
                Formula.add(i++, Num.toString());
                if (!combination)
                    ShowFormula.append(Num);
                Num.delete(0, Num.length());
            }
            Formula.add(i++,"pow");
            ShowFormula.append("^");
            combination=false;
            break;
        case R.id.btn_epower:
            if(Num.length()!=0)
            {   ShowFormula.append("e^");
                if (!combination){

                    ShowFormula.append(Num);
                }
                combination=true;

                num=Double.valueOf(Num.toString());
                num=Math.exp(num);
                NUM=String.valueOf(num);
                Num=new StringBuffer(NUM);
                Text1.setText(Num);
            }
            break;
        case R.id.btn_10power:
            if(Num.length()!=0)
            {
                ShowFormula.append("10^");
                if (!combination){
                    ShowFormula.append(Num);
                }
                combination=true;
                num=Double.valueOf(Num.toString());
                num=Math.pow(10,num);
                NUM=String.valueOf(num);
                Num=new StringBuffer(NUM);
                Text1.setText(Num);
            }
            break;
        case R.id.btn_rec://倒数
            if(Num.length()!=0)
            {
                ShowFormula.append("1/");
                if (!combination){

                    ShowFormula.append(Num);
                }
                combination=true;
                num=Double.valueOf(Num.toString());
                num=1/num;
                NUM=String.valueOf(num);
```

```
                    Num=new StringBuffer(NUM);
                    Text1.setText(Num);
                }
                break;
            case R.id.btn_sqrt:
                if(Num.length()!=0)
                {
                    ShowFormula.append("√");
                    if (!combination){

                        ShowFormula.append(Num);
                    }
                    combination=true;
                    num=Double.valueOf(Num.toString());
                    if (num>=0)
                    {
                        num=Math.sqrt(num);
                        NUM=String.valueOf(num);
                        Num=new StringBuffer(NUM);
                        Text1.setText(Num);
                    }
                    else
                    {
                        Text1.setText("错误,请重新输入数字");
                        Num.delete(0, Num.length());
                    }
                }
                break;
            case R.id.btn_cube_root:
                if(Num.length()!=0)
                {
                    ShowFormula.append("³√");
                    if (!combination){

                        ShowFormula.append(Num);
                    }
                    combination=true;
                    num=Double.valueOf(Num.toString());
                    num=Math.cbrt(num);
                    NUM=String.valueOf(num);
                    Num=new StringBuffer(NUM);
                    Text1.setText(Num);
                }
                break;
            case R.id.btn_y_root:
                //这个公式的 y 要变成 y/1
                if(Num.length()!=0)
                {
                    ShowFormula.append("y√");
                    if (!combination){
                        ShowFormula.append(Num);
                    }
                    combination=true;
                    Formula.add(i++, Num.toString());
                    Formula.add(i++,"root");
                    Num.delete(0, Num.length());
                }
                break;
            case R.id.btn_ln:
                if(Num.length()!=0)
                {
                    ShowFormula.append("ln");
                    if (!combination){
                        ShowFormula.append(Num);
                    }
                    combination=true;
                    num=Double.valueOf(Num.toString());
                    num=Math.log(num);
                    NUM=String.valueOf(num);
                    Num=new StringBuffer(NUM);
                    Text1.setText(Num);
                }
                break;
            case R.id.btn_log10:
                if(Num.length()!=0)
                {
                    ShowFormula.append("log10(");
                    if (!combination){

                        ShowFormula.append(Num);
                    }
                    ShowFormula.append(")");
                    combination=true;
                    num=Double.valueOf(Num.toString());
                    num=Math.log10(num);
                    NUM=String.valueOf(num);
                    Num=new StringBuffer(NUM);
                    Text1.setText(Num);
                }
                break;
            case R.id.btn_Factorial://阶乘
                if(Num.length()!=0)
                {
                    if (!combination){

                        ShowFormula.append(Num);
                    }
                    ShowFormula.append("!");
                    combination=true;
                    num=Double.valueOf(Num.toString());
                    if (num-(int)num==0)
                    {
                        num=Factorial(num);
                        NUM=String.valueOf(num);
                        Num=new StringBuffer(NUM);
                        Text1.setText(Num);
                    }
                    else
                    {
                        Text1.setText("错误,请输入一个正整数");
                        Num.delete(0, Num.length());
                    }
                }
                break;
            case R.id.btn_sin:
                //弧度表示
                if(Num.length()!=0)
                {
                    ShowFormula.append("sin(");
                    if (!combination){

                        ShowFormula.append(Num);
                    }
                    ShowFormula.append(")");
                    combination=true;

                    num=Double.valueOf(Num.toString());
                    num=Math.sin(num);
                    NUM=String.valueOf(num);
                    Num=new StringBuffer(NUM);
                    Text1.setText(Num);
                }
                break;
            case R.id.btn_cos:
                //弧度表示
                if(Num.length()!=0)
                {
                    ShowFormula.append("cos(");
                    if (!combination){

                        ShowFormula.append(Num);
                    }
                    ShowFormula.append(")");
                    combination=true;

                    num=Double.valueOf(Num.toString());
                    num=Math.cos(num);
                    NUM=String.valueOf(num);
                    Num=new StringBuffer(NUM);
                    Text1.setText(Num);
                }
                break;
            case R.id.btn_tan:
                //弧度表示
                if(Num.length()!=0)
                {
                    ShowFormula.append("tan(");
                    if (!combination){
                        ShowFormula.append(Num);
                    }
                    ShowFormula.append(")");
                    combination=true;

                    num=Double.valueOf(Num.toString());
                    num=Math.tan(num);
                    NUM=String.valueOf(num);
                    Num=new StringBuffer(NUM);
                    Text1.setText(Num);
                }
                break;
            case R.id.btn_sinh:
                //弧度表示
                if(Num.length()!=0)
                {
                    ShowFormula.append("sinh(");
                    if (!combination){

                        ShowFormula.append(Num);
                    }
                    ShowFormula.append(")");
                    combination=true;

                    num=Double.valueOf(Num.toString());
                    num=Math.sinh(num);
                    NUM=String.valueOf(num);
                    Num=new StringBuffer(NUM);
                    Text1.setText(Num);
                }
                break;
            case R.id.btn_cosh:
                //弧度表示
                if(Num.length()!=0)
                {
                    ShowFormula.append("cosh(");
                    if (!combination)
                    {

                        ShowFormula.append(Num);
                    }
                    ShowFormula.append(")");
                    combination=true;

                    num=Double.valueOf(Num.toString());
                    num=Math.cosh(num);
                    NUM=String.valueOf(num);
                    Num=new StringBuffer(NUM);
                    Text1.setText(Num);
                }
                break;
            case R.id.btn_tanh:
                //弧度表示
                if(Num.length()!=0)
                {
                    ShowFormula.append("tanh(");
                    if (!combination){

                        ShowFormula.append(Num);
                    }
                    ShowFormula.append(")");
                    combination=true;
```

```

                    num=Double.valueOf(Num.toString());
                    num=Math.tanh(num);
                    NUM=String.valueOf(num);
                    Num=new StringBuffer(NUM);
                    Text1.setText(Num);
                }
                break;
            case R.id.btn_e:
                Num.delete(0, Num.length());
                num=Math.exp(1);
                NUM=String.valueOf(num);
                Num=new StringBuffer(NUM);
                break;
            case R.id.btn_EE:
                //x*10^y;
                if(Num.length()!=0)
                {
                    if (!combination){

                        ShowFormula.append(Num);
                    }
                    ShowFormula.append("×10^");
                    combination=true;

                    Formula.add(i++, Num.toString());
                    Formula.add(i++,"EE");
                    Num.delete(0, Num.length());
                }
                combination=false;
                break;
            case R.id.btn_pai:
                Num.delete(0, Num.length());
                num=Math.PI;
                NUM=String.valueOf(num);
                Num=new StringBuffer(NUM);
                break;
            case R.id.btn_Rand://随机数
                Num.delete(0, Num.length());
                num=Math.random();
                NUM=String.valueOf(num);
                Num=new StringBuffer(NUM);
                break;
        }
        //如果没有数字，那么 AC 键是 Clear
        if (Num.length()==0)
            myBtn_AC.setText("AC");
        else myBtn_AC.setText("C");
    }

    public ArrayList<String> GetSuffix(ArrayList<String> Formula)
    {
        java.util.Stack<String> operatorStack = new
java.util.Stack<>();
        ArrayList<String> houzhui=new ArrayList<String>();
        int index=0;
        int dan=-1;

        for (int j=0;j<i;j++)
        {
            dan++;
            if (Formula.get(j)=="(")//左括号
            {   bracketsExist+=2;
                operatorStack.push(Formula.get(j));
                dan--;
                continue;
            }
            else
            {
                if (Formula.get(j)==")")//右括号
                {
                    while (!operatorStack.empty() &&
operatorStack.peek() != "(")//非空且栈顶不是左括号
                    {
                        Log.v(operatorStack.peek(),"栈顶元素");
                        houzhui.add(index++, operatorStack.pop());//
弹出栈顶符号进入后缀表达式
                    }
                    Log.v(operatorStack.peek(),"栈顶元素");
                    String zhanding=operatorStack.pop();//输出左括号
                    dan--;
                    continue;
                }
            }

            if (dan%2==0)//数字
            {
                houzhui.add(index++,Formula.get(j));
            }
            else//非数字（符号）
            {
                if (operatorStack.empty())
                {
                    operatorStack.push(Formula.get(j));
                }
                else
                {
                    //符号栈非空且待进符号优先级小于等于栈顶符号
                    while (!operatorStack.empty() &&
operatorStack.peek()!="(" && getPriority(Formula.get(j)) <=
getPriority(operatorStack.peek()))
                    {
                        houzhui.add(index++, operatorStack.pop());//
弹出栈顶符号进入后缀表达式
                    }
                    operatorStack.push(Formula.get(j));//弹入待进符
号;
                }
            }

        }

        while (!operatorStack.empty())//最后将所有符号输出
        {
            houzhui.add(index++,operatorStack.pop());
        }
        return houzhui;
    }

    public String CalculaSuffix(ArrayList<String> Suffix)
    {
        java.util.Stack<String> numStack = new java.util.Stack<>();
        double num1;
        double num2;
        double num3;
        String NUM3;
        for (int j=0;j<i-bracketsExist;j++)
        {
            switch (Suffix.get(j))
            {
                case "add":
                    num1 = Double.valueOf(numStack.pop());
                    num2 = Double.valueOf(numStack.pop());
                    num3 = num1 +num2;
                    NUM3=String.valueOf(num3);
                    numStack.push(NUM3);
                    break;
                case "sub":
                    num1 = Double.valueOf(numStack.pop());
                    num2 = Double.valueOf(numStack.pop());
                    num3 = num2-num1;
                    NUM3=String.valueOf(num3);
                    numStack.push(NUM3);
                    break;
                case "mul":
                    num1 = Double.valueOf(numStack.pop());
                    num2 = Double.valueOf(numStack.pop());
                    num3 = num1*num2;
                    NUM3=String.valueOf(num3);
                    numStack.push(NUM3);
                    break;
                case "div":
                    num1 = Double.valueOf(numStack.pop());//y
                    num2 = Double.valueOf(numStack.pop());//x
                    num3 = num2/num1;//x/y
                    NUM3=String.valueOf(num3);
                    numStack.push(NUM3);
                    break;
                case "root":
                    num1 = Double.valueOf(numStack.pop());//y
                    num2 = Double.valueOf(numStack.pop());//x
                    num3 = Math.pow(num2,1/num1);//x^(1/y)
                    NUM3=String.valueOf(num3);
                    numStack.push(NUM3);
                    break;
                case "EE":
                    num1 = Double.valueOf(numStack.pop());//y
                    num2 = Double.valueOf(numStack.pop());//x
                    num3 = num2*Math.pow(10,num1);
                    NUM3=String.valueOf(num3);
                    numStack.push(NUM3);
                    break;
                case "pow":
                    num1 = Double.valueOf(numStack.pop());//y
                    num2 = Double.valueOf(numStack.pop());//x
                    num3 = Math.pow(num2,num1);
```

```
                    NUM3=String.valueOf(num3);
                    numStack.push(NUM3);
                    break;
                default:
                    numStack.push(Suffix.get(j));
                    break;
            }
        }
        return numStack.pop();
    }

    public double Factorial(double n)
    {
        if (n>=2)
            return n*Factorial(n-1);
        else return 1;
    }

    public int getPriority(String symbol)
    {
        switch (symbol)
        {
            case "(":
            case ")":
                return 5;
            case "pow":
                return 4;
            case "EE":
            case "root":
                return 3;
            case "div":
            case "mul":
                return 2;
            default:
                return 1;
        }
    }
}
```

5. Style.xml

```
<resources>

```

```
    <!-- Base application theme. -->
    <style name="AppTheme"
parent="Theme.AppCompat.Light.DarkActionBar">
        <!-- Customize your theme here. -->
        <item name="colorPrimary">@color/colorPrimary</item>
        <item name="colorPrimaryDark">@color/colorPrimaryDark</item>
        <item name="colorAccent">@color/colorAccent</item>
    </style>
    <style name="NumBtnStyle" >
        <item name="android:typeface">monospace</item>
        <item name="android:textAllCaps">false</item>
        <item name="android:textSize">21sp</item>
        <item name="android:layout_margin">0.1dp</item>
        <item name="android:textColor">#0A0A0A</item>
        <item name="android:background">#DEDEDE</item>
        <item name="android:layout_width">80dp</item>
        <item name="android:layout_height">70dp</item>
    </style>
    <style name="SymbolBtnStyle" >
        <item name="android:textAllCaps">false</item>
        <item name="android:textSize">21sp</item>
        <item name="android:layout_margin">0.1dp</item>
        <item name="android:textColor">#0A0A0A</item>
        <item name="android:background">#BEBEBE</item>
        <item name="android:layout_width">80dp</item>
        <item name="android:layout_height">70dp</item>
    </style>
    <style name="RuleBtnStyle" >
        <item name="android:textAllCaps">false</item>
        <item name="android:textSize">22sp</item>
        <item name="android:layout_margin">0.1dp</item>
        <item name="android:textColor">#FFFFFF</item>
        <item name="android:background">#DE6600</item>
        <item name="android:layout_width">80dp</item>
        <item name="android:layout_height">70dp</item>
        <item name="android:textStyle">bold</item>
    </style>
    <style name="NoTitle"
parent="Theme.AppCompat.DayNight.NoActionBar">
        <item name="android:windowNoTitle">true</item>
        <item name="android:windowFullscreen">true</item>
    </style>
    <style name="HNumBtnStyle" >
        <item name="android:typeface">monospace</item>
        <item name="android:textAllCaps">false</item>
        <item name="android:textSize">12sp</item>
        <item name="android:layout_margin">0.1dp</item>
        <item name="android:textColor">#0A0A0A</item>
        <item name="android:background">#DEDEDE</item>
        <item name="android:layout_width">52dp</item>
        <item name="android:layout_height">40dp</item>
        <item name="android:textStyle">bold</item>
    </style>
    <style name="HSymbolBtnStyle" >
        <item name="android:textAllCaps">false</item>
        <item name="android:textSize">12sp</item>
        <item name="android:layout_margin">0.1dp</item>
        <item name="android:textColor">#0A0A0A</item>
        <item name="android:background">#BEBEBE</item>
        <item name="android:layout_width">52dp</item>
        <item name="android:layout_height">40dp</item>
    </style>
    <style name="HRuleBtnStyle" >
        <item name="android:textAllCaps">false</item>
        <item name="android:textSize">15sp</item>
        <item name="android:layout_margin">0.1dp</item>
        <item name="android:textColor">#FFFFFF</item>
        <item name="android:background">#DE6600</item>
        <item name="android:layout_width">52dp</item>
        <item name="android:layout_height">40dp</item>
        <item name="android:textStyle">bold</item>
    </style>

</resources>
```

难点（或遇到的问题）和解决方案

第一个难点就是在设置按钮的时候，对按钮的大小，背景颜色，字体粗细等的调整比较繁琐，后来为 3 种类型的按钮设计了 3 种不同的 Style，就会比较简单一点。

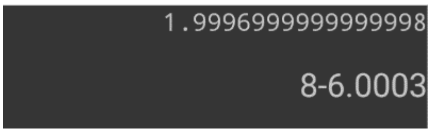
第二个难点就是计算的式子和显示的式子不一样。因为我用于计算的式子都是数字和二元运算符交替出现的字符串组，所以不适用于显示。我采取的解决方案是创建一个新的显示式子的字符串组，用于记录一元运算符的符号。比如当我依次按下 3，In ，x^2 =后，显示式子为 In3 ^{2}。

第三个难点就是横竖版之间的切换。我设计了横板界面和竖版界面，但是怎么实现当切

换横竖屏时自动切换界面，困扰了我许久。有一天我去问老师，老师是可以用 fragment，但是我去查了有关资料，觉得太麻烦了。然后我就看了书本的第三章，80 页里面提到了横竖屏切换会销毁并重建 Activity。那我就想，能不能再重建 Activity 时对横竖屏进行判断，并切换相应至的 Activity。然后我从网上查找了有关资料，通过 Configuration mConfiguration = this.getResources().getConfiguration()可以返回当前配置信息，mConfiguration.orientation 返回横竖屏信息，这样就可以对横竖屏进行判断了。然后翻开课本 83 页发现创建 Intent 对象可以实现对 Activity 的切换，问题迎刃而解。

不足之处

不足之处，也有挺多的吧。比如计算较多位小数时，会出现错误。

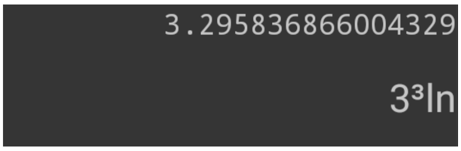


```
1.9996999999999998

8-6.0003
```

这是 Java 的精度缺失，其实许多语言都会出现这样的问题，java 内可以用 BigDecimal 类解决。

显示式子方面，也会有 BUG。因为是显示式子的数据类型是 StringBuffer，所以插入会比较困难。导致了一元运算符只能往后加，而不能插入。例如 ln(3^3)，只能显示成



3.295836866004329

3^3ln

如果可以的话，将显示式子改为 ArrayList<String>类型会比较容易操作一点。

AC 是 All Clear，C 是 Clear。清除的话只能清除数字，而不能清除运算符。而且清除数字只能整个数字都清除，不能清除一位数字。要清除运算符的话，就要重新输入式子。

还有布局文件我是根据 4_WVGA_Nexus_S 的屏幕做的，只适用于与 4_WVGA_Nexus_S 一样尺寸的屏幕，如果用于其他尺寸的手机，可能会出现空白或者按键消失。这是因为我用的布局方式不对。我用的是表格布局，但其实用网格布局会比较合理一点，比较方便一点，由于时间关系，就没有再修改。

今后的设想

这次作业给的时间有限，本人的水平也一般般，之前完全没接触过安卓软件开发，所以

有花一定的时间来适应。一开始我的电脑 C 盘空间不足，Android Studio 安装了好多次安装不上，后来重装系统才装上。

做这个计算器的过程是一次摸着石头过河的过程，首先要想有什么键位，什么运算，什么算法来实现计算。也问过老师问过同学，也得到了相应了解决方案。中缀转后缀这个方法是现学现卖的，计算后缀表达式也是。这次作业也让后我对 java 的一些数据结构更加熟悉，比如栈，比如 ArrayList<String>，StringBuffer。

虽然出现了很多 BUG，但是毕竟是第一次安卓开发，万事开头难，而且自己独立完成一个小软件也是蛮有成就感的，希望以后能再接再厉，以后从事移动应用开发方面的工作也未尝不可。还是要继续努力。