

计算器的设计与实现说明文档

完成人：许博勇 学号：20152100188

完成时间：2017 月 10 月 13 号

一、软件名称

较为科学的计算器

二、软件内容简介

这是一个使用 android studio 编写的较为科学的计算器，它的软件界面的交互我借鉴了我的手机自带的计算器，但是我按照了谷歌的 Material Design 风格重新对计算器界面进行了新的配色。此外，我对计算器内置的 ActionBar 改为谷歌最新的 ToolBar，使界面的风格更为自然。

在计算器功能的实现方面，由于我在大二的数据结构课程中是有学过中缀表达式、后缀表达式与其之间的转化，所以在这个软件中也大体按照了之前的思想，在有关算法的代码方面，我是在参考了算法思想后自己敲的代码，所以有可能代码并不是最简洁的，但是测试起来是可以完成的。

具体的算法实现方面，我在对计算器进行界面布局之后，使用 `ArrayList<Character>` 读取我们输入的每一个数字以及操作符，每一位的数字和操作符均占用一个位置，然后对这个 `Arraylist<Character>` 进行处理，将多位数字合成为一个数（包括点），再对此中缀表达式使用 `Arraylist<String>` 和 `Stack<Character>` 存储并利用优先级转化为后缀表达式，在后缀表达式中也正好可以针对括号进行运算顺序调整，最后对后缀表达式进行计算，得出结果并显示。

而这个计算器叫做较为科学的计算器而不是科学计算器的原因是因为和我手机上自带的科学计算器相比还是有一些不足之处，这个具体我们会在后面的不足之处中说明。

三、界面设计

(一) 主界面:



(二) 右划:



(三) 输入普通计算：



(四) 输入等于：



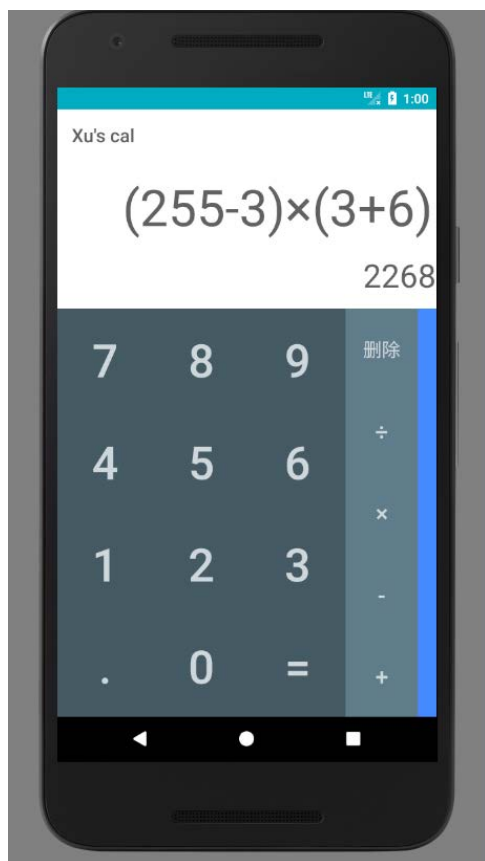
(五) 删除按钮:



(六) 长按删除按钮清空:



(七) 进行包含括号的计算：



(八) 进行较为复杂的计算：



(九) 计算式过长时分屏显示:



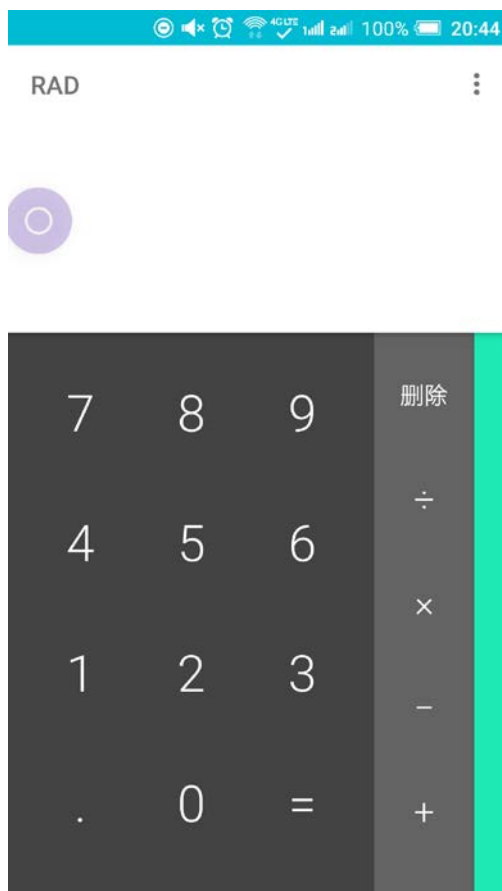
(十) 输入小数计算:



(十一) 对数计算展示:



(十二) 借鉴的自带计算器界面：



五、代码设计

(一) 主界面布局代码 activity_main.xml:

```
<?xml version="1.0" encoding="utf-8"?>
<LinearLayout 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:orientation="vertical"
    tools:context="com.example.yong.calculateapp.MainActivity">

    <android.support.v7.widget.Toolbar
        android:id="@+id/toolbar"
        android:layout_width="match_parent"
        android:layout_height="?attr/actionBarSize"
        android:background="?attr/colorPrimary"
        app:title="Xu's cal"
        app:titleTextColor="#616161"/>

    <EditText
        android:id="@+id/textViewIn"
        android:layout_width="match_parent"
        android:layout_height="100dp"
        android:maxLines="1"
        android:textColor="#616161"
        android:focusable="false"
        android:cursorVisible="false"
        android:scrollbars="vertical"
        android:gravity="center_vertical|end"
        android:textSize="55sp" />

    <TextView
        android:id="@+id/textViewResult"
        android:layout_width="match_parent"
        android:layout_height="60dp"
        android:gravity="end"
        android:textColor="#616161"
        android:textSize="35sp" />

    <android.support.v4.widget.DrawerLayout
        android:layout_width="match_parent"
```

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

```
<LinearLayout
```

```
    android:layout_width="match_parent"  
    android:layout_height="match_parent"  
    android:orientation="horizontal">
```

```
<GridLayout
```

```
    android:layout_width="0dp"  
    android:layout_height="match_parent"  
    android:layout_weight="8"  
    android:columnCount="3"  
    android:rowCount="4">
```

```
<Button
```

```
    android:id="@+id/btn7"  
    style="@style/BtnStyle"  
    android:layout_width="wrap_content"  
    android:layout_height="wrap_content"  
    android:layout_columnWeight="1"  
    android:layout_rowWeight="1"  
    android:text="7" />
```

```
<Button
```

```
    android:id="@+id/btn8"  
    style="@style/BtnStyle"  
    android:layout_width="wrap_content"  
    android:layout_height="wrap_content"  
    android:layout_columnWeight="1"  
    android:layout_rowWeight="1"  
    android:text="8" />
```

```
<Button
```

```
    android:id="@+id/btn9"  
    style="@style/BtnStyle"  
    android:layout_width="wrap_content"  
    android:layout_height="wrap_content"  
    android:layout_columnWeight="1"  
    android:layout_rowWeight="1"  
    android:text="9" />
```

```
<Button
    android:id="@+id/btn4"
    style="@style/BtnStyle"
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:layout_columnWeight="1"
    android:layout_rowWeight="1"
    android:text="4" />
```

```
<Button
    android:id="@+id/btn5"
    style="@style/BtnStyle"
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:layout_columnWeight="1"
    android:layout_rowWeight="1"
    android:text="5" />
```

```
<Button
    android:id="@+id/btn6"
    style="@style/BtnStyle"
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:layout_columnWeight="1"
    android:layout_rowWeight="1"
    android:text="6" />
```

```
<Button
    android:id="@+id/btn1"
    style="@style/BtnStyle"
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:layout_columnWeight="1"
    android:layout_rowWeight="1"
    android:text="1" />
```

```
<Button
    android:id="@+id/btn2"
    style="@style/BtnStyle"
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
```

```
        android:layout_columnWeight="1"
        android:layout_rowWeight="1"
        android:text="2" />
```

<Button

```
    android:id="@+id/btn3"
    style="@style/BtnStyle"
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:layout_columnWeight="1"
    android:layout_rowWeight="1"
    android:text="3" />
```

<Button

```
    android:id="@+id/btnPoint"
    style="@style/BtnStyle"
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:layout_columnWeight="1"
    android:layout_rowWeight="1"
    android:text="." />
```

<Button

```
    android:id="@+id/btn0"
    style="@style/BtnStyle"
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:layout_columnWeight="1"
    android:layout_rowWeight="1"
    android:text="0" />
```

<Button

```
    android:id="@+id/btnEqual"
    style="@style/BtnStyle"
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:layout_columnWeight="1"
    android:layout_rowWeight="1"
    android:text="=" />
```

</GridLayout>

<LinearLayout

```
    android:layout_width="0dp"
    android:layout_height="match_parent"
    android:layout_weight="2"
    android:orientation="vertical">
```

<Button

```
    android:id="@+id/btnDel"
    style="@style/CalculateStyle"
    android:layout_width="match_parent"
    android:layout_height="match_parent"
    android:text="删除"
    android:textSize="20sp" />
```

<Button

```
    android:id="@+id/btnDivide"
    style="@style/CalculateStyle"
    android:layout_width="match_parent"
    android:layout_height="match_parent"
    android:text="÷"
    android:textSize="25sp" />
```

<Button

```
    android:id="@+id/btnMul"
    style="@style/CalculateStyle"
    android:layout_width="match_parent"
    android:layout_height="match_parent"
    android:text="×"
    android:textSize="25sp" />
```

<Button

```
    android:id="@+id/btnSub"
    style="@style/CalculateStyle"
    android:layout_width="match_parent"
    android:layout_height="match_parent"
    android:text="-"
    android:textSize="25sp" />
```

<Button

```
    android:id="@+id/btnAdd"
```

```
style="@style/CalculateStyle"
android:layout_width="match_parent"
android:layout_height="match_parent"
android:text="+"
android:textSize="25sp" />
```

</LinearLayout>

<TextView

```
android:layout_width="0dp"
android:layout_height="match_parent"
android:layout_weight="0.5"
android:background="#448AFF" />
```

</LinearLayout>

<GridLayout

```
android:layout_width="260dp"
android:layout_height="match_parent"
android:layout_gravity="end"
android:columnCount="3"
android:rowCount="5">
```

<Button

```
android:id="@+id/btnLeft"
style="@style/extraStyle"
android:layout_width="wrap_content"
android:layout_height="wrap_content"
android:layout_columnWeight="1"
android:layout_rowWeight="1"
android:text="(" />
```

<Button

```
android:id="@+id/btnRight"
style="@style/extraStyle"
android:layout_width="wrap_content"
android:layout_height="wrap_content"
android:layout_columnWeight="1"
android:layout_rowWeight="1"
android:text=")" />
```

<Button

```
    android:id="@+id/btnMod"
    style="@style/extraStyle"
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:layout_columnWeight="1"
    android:layout_rowWeight="1"
    android:text="%" />
```

<Button

```
    android:id="@+id/btnPai"
    style="@style/extraStyle"
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:textAllCaps="false"
    android:layout_columnWeight="1"
    android:layout_rowWeight="1"
    android:text="π" />
```

<Button

```
    android:id="@+id/btnE"
    style="@style/extraStyle"
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:textAllCaps="false"
    android:layout_columnWeight="1"
    android:layout_rowWeight="1"
    android:text="e" />
```

<Button

```
    android:id="@+id/btnFactorial"
    style="@style/extraStyle"
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:layout_columnWeight="1"
    android:layout_rowWeight="1"
    android:text="!" />
```

<Button

```
    android:id="@+id/btnSqu"
```

```
style="@style/extraStyle"
android:layout_width="wrap_content"
android:layout_height="wrap_content"
android:layout_columnWeight="1"
android:layout_rowWeight="1"
android:text="2" />
```

<Button

```
android:id="@+id/btnThree"
style="@style/extraStyle"
android:layout_width="wrap_content"
android:layout_height="wrap_content"
android:layout_columnWeight="1"
android:layout_rowWeight="1"
android:text="3" />
```

<Button

```
android:id="@+id/btnExtract"
style="@style/extraStyle"
android:layout_width="wrap_content"
android:layout_height="wrap_content"
android:layout_columnWeight="1"
android:layout_rowWeight="1"
android:text="✓" />
```

<Button

```
android:id="@+id/btnLn"
style="@style/extraStyle"
android:layout_width="wrap_content"
android:layout_height="wrap_content"
android:textAllCaps="false"
android:layout_columnWeight="1"
android:layout_rowWeight="1"
android:text="ln" />
```

<Button

```
android:id="@+id/btnLog10"
style="@style/extraStyle"
android:layout_width="wrap_content"
```



```

        android:layout_height="wrap_content"
        android:textAllCaps="false"
        android:layout_columnWeight="1"
        android:layout_rowWeight="1"
        android:text="lg" />

        <Button
            android:id="@+id/btnE10"
            style="@style/extraStyle"
            android:layout_width="wrap_content"
            android:layout_height="wrap_content"
            android:layout_columnWeight="1"
            android:layout_rowWeight="1"
            android:text="" />

    </GridLayout>

</android.support.v4.widget.DrawerLayout>

</LinearLayout>

```

(二) MainActivity:

```

package com.example.yong.calculateapp;

import android.support.v7.app.AppCompatActivity;
import android.os.Bundle;
import android.support.v7.widget.Toolbar;
import android.text.TextUtils;
import android.view.View;
import android.widget.EditText;
import android.widget.TextView;

import java.math.BigDecimal;
import java.util.ArrayList;
import java.util.Stack;

public class MainActivity extends AppCompatActivity implements
View.OnClickListener, View.OnLongClickListener {

    private TextView inView;
    private TextView resultView;

```

```

private ArrayList<Character> inCallList = new ArrayList<>();
@Override
protected void onCreate(Bundle savedInstanceState) {
    super.onCreate(savedInstanceState);
    setContentView(R.layout.activity_main);
    inView = (EditText) findViewById(R.id.textViewIn);
    Toolbar toolbar = (Toolbar) findViewById(R.id.toolbar);
    resultView = (TextView) findViewById(R.id.textViewResult);
    setSupportActionBar(toolbar);
    findViewById(R.id.btn0).setOnClickListener(this);
    findViewById(R.id.btn1).setOnClickListener(this);
    findViewById(R.id.btn2).setOnClickListener(this);
    findViewById(R.id.btn3).setOnClickListener(this);
    findViewById(R.id.btn4).setOnClickListener(this);
    findViewById(R.id.btn5).setOnClickListener(this);
    findViewById(R.id.btn6).setOnClickListener(this);
    findViewById(R.id.btn7).setOnClickListener(this);
    findViewById(R.id.btn8).setOnClickListener(this);
    findViewById(R.id.btn9).setOnClickListener(this);
    findViewById(R.id.btnAdd).setOnClickListener(this);
    findViewById(R.id.btnDel).setOnClickListener(this);
    findViewById(R.id.btnDel).setOnLongClickListener(this);
    findViewById(R.id.btnDivide).setOnClickListener(this);
    findViewById(R.id.btnSub).setOnClickListener(this);
    findViewById(R.id.btnEqual).setOnClickListener(this);
    findViewById(R.id.btnMul).setOnClickListener(this);
    findViewById(R.id.btnPoint).setOnClickListener(this);
    findViewById(R.id.btnMod).setOnClickListener(this);
    findViewById(R.id.btnLeft).setOnClickListener(this);
    findViewById(R.id.btnRight).setOnClickListener(this);
    findViewById(R.id.btnE).setOnClickListener(this);
    findViewById(R.id.btnPai).setOnClickListener(this);
    findViewById(R.id.btnSqu).setOnClickListener(this);
    findViewById(R.id.btnThree).setOnClickListener(this);
    findViewById(R.id.btnExtract).setOnClickListener(this);
    findViewById(R.id.btnFactorial).setOnClickListener(this);
    findViewById(R.id.btnLn).setOnClickListener(this);
    findViewById(R.id.btnLog10).setOnClickListener(this);
}

```

```
}  
  
@Override  
public boolean onLongClick(View view) {  
    switch (view.getId()) {  
        case R.id. btnDel:  
            inCallList.clear();  
            inView.setText("");  
            resultView.setText("");  
            break;  
    }  
    return true;  
}  
  
@Override  
public void onClick(View view) {  
    switch (view.getId()) {  
        case R.id. btn0:  
            inCallList.add('0');  
            inView.append("0");  
            break;  
  
        case R.id. btn1:  
            inCallList.add('1');  
            inView.append("1");  
            break;  
  
        case R.id. btn2:  
            inCallList.add('2');  
            inView.append("2");  
            break;  
  
        case R.id. btn3:  
            inCallList.add('3');  
            inView.append("3");  
            break;  
  
        case R.id. btn4:  
            inCallList.add('4');  
            inView.append("4");  
            break;  
  
        case R.id. btn5:  
            inCallList.add('5');  
            inView.append("5");  
            break;  
  
        case R.id. btn6:
```

```
        inCallList.add(' 6' );
        inView.append("6");
        break;
    case R.id.btn7:
        inCallList.add(' 7' );
        inView.append("7");
        break;
    case R.id.btn8:
        inCallList.add(' 8' );
        inView.append("8");
        break;
    case R.id.btn9:
        inCallList.add(' 9' );
        inView.append("9");
        break;
    case R.id.btnAdd:
        inCallList.add(' +' );
        inView.append("+");
        break;
    case R.id.btnSub:
        inCallList.add(' -' );
        inView.append("-");
        break;
    case R.id.btnDivide:
        inCallList.add(' /' );
        inView.append("÷");
        break;
    case R.id.btnMul:
        inCallList.add(' *' );
        inView.append("×");
        break;
    case R.id.btnDelete:
        if (!TextUtils.isEmpty(inView.getText())) {
            inCallList.remove(inCallList.size() - 1);
            String a = inView.getText().toString();
            a = a.substring(0, a.length() - 1);
            inView.setText(a);
            resultView.setText("");
        }
        break;
    case R.id.btnPoint:
```

```

        inCallList.add(' ');
        inView.append(" ");
        break;
    case R.id.btnEqual:
        convert(inCallList);
        break;
    case R.id.btnLeft:
        inCallList.add('(');
        inView.append("(");
        break;
    case R.id.btnRight:
        inCallList.add(')');
        inView.append(")");
        break;
    case R.id.btnMod:
        inCallList.add('%');
        inView.append("%");
        break;
    case R.id.btnE:
        inView.append("e");
        String e = "2.718281828";
        for (int i=0;i<e.length();i++)
            inCallList.add(e.charAt(i));
        break;
    case R.id.btnPai:
        inView.append("π");
        String pai = "3.141592653";
        for (int i=0;i<pai.length();i++)
            inCallList.add(pai.charAt(i));
        break;
    case R.id.btnSqu:
        inView.append("²");
        inCallList.add('²');
        break;

    case R.id.btnExtract:
        inView.append("√");
        inCallList.add('√');
        break;
    case R.id.btnFactorial:
        inView.append("!");

```

```

        inCallList.add('!');
        break;
    case R.id.btnThree:
        inView.append("3");
        inCallList.add('3');
        break;
    case R.id.btnLn:
        inView.append("ln");
        inCallList.add('n');
        break;
    case R.id.btnLog10:
        inView.append("lg");
        inCallList.add('g');
        break;
    default:break;
}

}

private int isp(Character symbol) {
    switch (symbol) {
        case '+':
        case '-':return 1;
        case '*':
        case '/':
        case '%':return 3;
        case '(':return 0;

        default:
            System.out.println("no");
            return 0;
    }
}

private void convert(ArrayList<Character> operation) {
    Stack<Character> operationStack = new Stack<>();
    ArrayList<String> allRecord = new ArrayList<>();
    for (int i=0;i<operation.size();i++) {
        Character inOperation=operation.get(i);
        if (Character.isDigit(inOperation)) {
//            if((i-1<0||operation.get(i-1)=='+'||operation.get(i-1)=='-'
//
//            ||operation.get(i-1)=='*'||operation.get(i-1)=='/'||operation.charAt(i-1)=='%'

```

```

',
//
||operation.charAt(i-1)=='('||operation.charAt(i-1)=='')&&
//
(i+1<operation.length()&&Character.isDigit(operation.charAt(i+1)))
//
        System.out.print("(");
        //System.out.println(inOperation);
        if (i>0) {
            if
(Character.isDigit(operation.get(i-1))||operation.get(i-1)=='.'){
                String mid =
allRecord.remove(allRecord.size()-1)+inOperation;
                allRecord.add(mid);
            }
            else allRecord.add(inOperation+"");
        }
        else
            allRecord.add(inOperation+"");

//
if((i+1>=operation.length()||operation.charAt(i+1)=='+'||operation.charAt(i+1)
)=='-'
//
||operation.charAt(i+1)=='*'||operation.charAt(i+1)=='/'||operation.charAt(i+1)
)=='%'
//
||operation.charAt(i+1)=='('||operation.charAt(i+1)=='')&&
//
        i-1>=0&&(Character.isDigit(operation.charAt(i-1))))
//
        System.out.print(")");
    }
    else
        if(inOperation=='+'||inOperation=='-'||inOperation=='*'||inOperation=='/'||in
Operation=='%'){
            if (operationStack.isEmpty() ||
isp(inOperation)>isp(operationStack.peek()))
                operationStack.push(inOperation);
            else {
                do {
                    //System.out.println(operationStack.pop());
                    allRecord.add(operationStack.pop()+"");
                }
            }
        }
    }
}

```

```

while(!operationStack.isEmpty() && isp(inOperation) <= isp(operationStack.peek()))
);

        operationStack.push(inOperation);
    }
}
else if (inOperation == '(') operationStack.push(inOperation);
else if (inOperation == ')') {
    while (operationStack.peek() != '(') {
        if (operationStack.isEmpty()) {
            System.out.println("括号不匹配");
            break;
        }
        //System.out.println(operationStack.pop());
        allRecord.add(operationStack.pop() + "");
    }
    operationStack.pop();
}
else if (inOperation == '^') {
    if (!allRecord.isEmpty()) {
        BigDecimal a = new
BigDecimal(allRecord.remove(allRecord.size()-1));
        a = a.multiply(a);
        allRecord.add(a.toString());
    }
}
else if (inOperation == '^3') {
    if (!allRecord.isEmpty()) {
        double a =
Double.parseDouble(allRecord.remove(allRecord.size()-1));
        a = Math.pow(a, 3);
        allRecord.add(a + "");
    }
}
else if (inOperation == '√') {
    if (!allRecord.isEmpty()) {
        double a =
Double.parseDouble(allRecord.remove(allRecord.size()-1));
        //a = Math.pow(a, 1/3);
        a = Math.sqrt(a);
        allRecord.add(a + "");
    }
}

```



```

    }
}
else if (inOperation=='!'){
    if(!allRecord.isEmpty()){
        int a =
(int)Double.parseDouble(allRecord.remove(allRecord.size()-1));
        allRecord.add(factorial(a)+"");
    }
}
else if (inOperation=='n'){
    if(!allRecord.isEmpty()){
        double a =
Math.Log(Double.parseDouble(allRecord.remove(allRecord.size()-1)));
        allRecord.add(a+"");
    }
}
else if (inOperation=='g'){
    if(!allRecord.isEmpty()){
        double a =
Logarithm(Double.parseDouble(allRecord.remove(allRecord.size()-1)),10);
        allRecord.add(a+"");
    }
}
else if (inOperation=='.'){
    if (i>0){
        if (Character.isDigit(operation.get(i-1))){
            String mid =
allRecord.remove(allRecord.size()-1)+inOperation;
            allRecord.add(mid);
        }
        else allRecord.add("0"+inOperation);
    }
    else allRecord.add("0"+inOperation);
}
else {
    System.out.println("输入不符合规范");
    break;
}
}
while (!operationStack.isEmpty()) {
    allRecord.add(operationStack.pop()+"");
}

```

```

    }
    if (calculate(allRecord)%1==0)
        resultView.append(calculate(allRecord).intValue()+"");
    else
        resultView.append(calculate(allRecord)+"");
}

private Double calculate(ArrayList<String> allNumber) {
    Stack<Double> numberStack=new Stack<>();
    for (String ex:allNumber) {
        boolean flag = true;
        for (int i=0;i<ex.length();i++) {
            if (!Character.isDigit(ex.charAt(i))&&ex.charAt(i)!='.') {
                flag = false;
                break;
            }
        }
        if (flag==true) //the string is digit
            numberStack.push(Double.parseDouble(ex));
        else {
            Double a;
            Double b;
            switch (ex.charAt(0)) {
                case '+':
                    a = numberStack.pop();
                    b = numberStack.pop();
                    numberStack.push(b + a);
                    break;
                case '-':
                    a = numberStack.pop();
                    b = numberStack.pop();
                    numberStack.push(b - a);
                    break;
                case '*':
                    a = numberStack.pop();
                    b = numberStack.pop();
                    numberStack.push(b * a);
                    break;
                case '/':
                    a = numberStack.pop();
                    b = numberStack.pop();

```

```

        numberStack.push(b / a);
        break;
    case '%':
        a = numberStack.pop();
        b = numberStack.pop();
        numberStack.push(b % a);
        break;
    default:break;
    }
}
}
}
return numberStack.pop();
}

private int factorial (int n){
    int num = 1;
    for (int i = 1; i <= n; i++) {
        num *= i;
    }
    return num;
}

private double Logarithm(double value,double base) {
    return Math.Log(value) / Math.Log(base);
}
}

```

(三) styles.xml

```

<resources>

    <!-- Base application theme. -->
    <style name="AppTheme" parent="Theme.AppCompat.NoActionBar">
        <!-- Customize your theme here. -->
        <item name="colorPrimary">@color/colorPrimary</item>
        <item name="colorPrimaryDark">@color/colorPrimaryDark</item>
        <item name="colorAccent">@color/colorAccent</item>
        <item name="android:windowBackground">@color/windowBackColor</item>
    </style>
    <style name="BtnStyle">
        <item name="android:textSize">50sp</item>
        <item name="android:layout_margin">-8dp</item>
    </style>

```

```

        <item name="android:textColor">#CFD8DC</item>
        <item name="android:background">#455A64</item>
    </style>
    <style name="CalculateStyle">
        <item name="android:background">#607D8B</item>
        <item name="android:textColor">#CFD8DC</item>
        <item name="android:layout_marginLeft">-8dp</item>
        <item name="android:layout_marginRight">-8dp</item>
        <item name="android:layout_marginBottom">-8dp</item>
        <item name="android:layout_marginTop">-8dp</item>
        <item name="android:layout_weight">1</item>
    </style>
    <style name="extraStyle">
        <item name="android:textSize">28sp</item>
        <item name="android:textColor">#212121</item>
        <item name="android:layout_margin">-8dp</item>
        <item name="android:background">#448AFF</item>
    </style>
</resources>

```

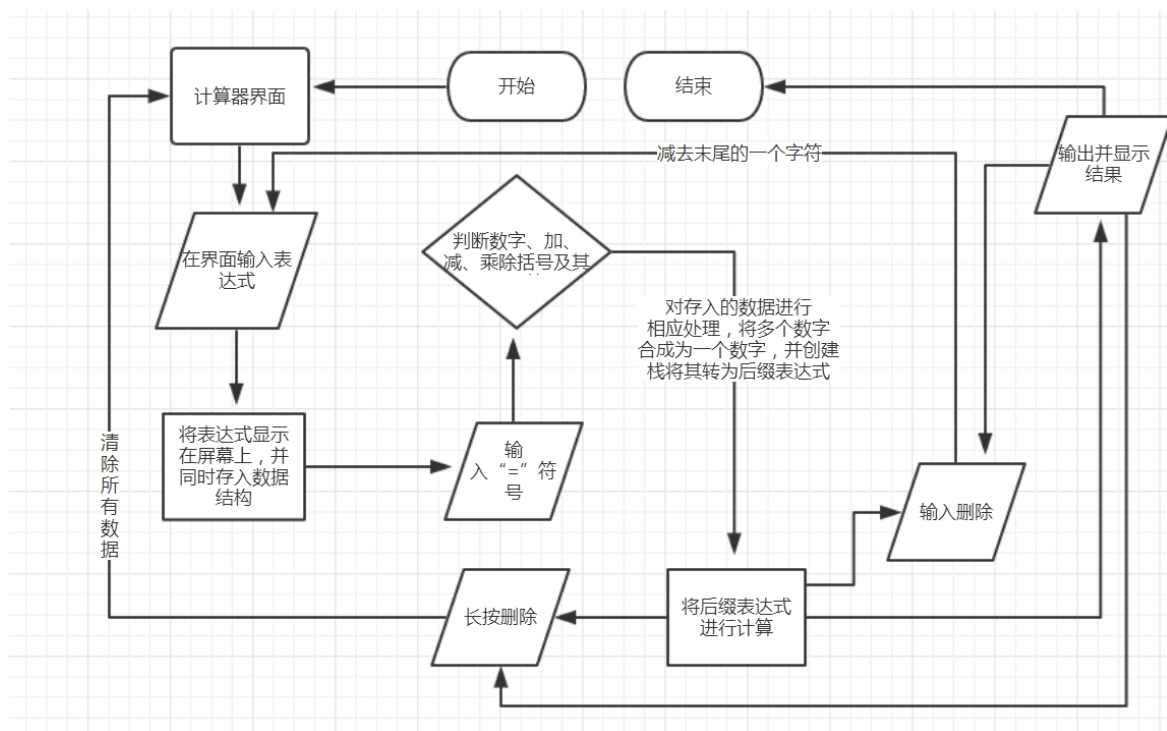
colors.xml

```

<?xml version="1.0" encoding="utf-8"?>
<resources>
    <color name="colorPrimary">#FFFFFF</color>
    <color name="colorPrimaryDark">#00ACC1</color>
    <color name="colorAccent">#FFFFFF</color>
    <color name="windowBackColor">#FFFFFF</color>
</resources>

```

六、软件操作流程



七、难点（或遇到的问题）和解决方案

这次计算器的设计与实现作业难点包括界面设计以及算法设计上面，由于是初次接触 android studio 设计，所以界面的难度也包括对 android studio 的上手难度。

1. 布局：让按钮没有间距的问题，在 android studio 中按钮在初设置中都默认保持有一定的间距，即使 margin 设置为 0 也有间距。

解决方案：巧妙地将 margin 设为 -8db 就会消除间距，可知默认按钮的间距为 8db。

2. 在左边操作符一栏不知道怎么回事设置 android:layout_margin 没有效果，按钮间还是有间距。

解决方案：分别对 margin 的 marginLeft、marginRight、marginBottom、marginTop 均设置为 -8db 即可实现按钮间没有间距的效果，这个我至今都不知道原因。

3. 计算器需要配置大量按钮，其中对于区域按钮很多的配置是一样的，如果单独对每一个按钮进行大量重复的设置不仅繁琐，而且使代码变得很冗杂。

解决方案：可以在 styles.xml 中定义一个 style 并在所有 button 中都引用这一个 style，这样代码会简洁很多。

4. 一开始使用 system.out.println 实现代码的调试查看，发现显示错误。

解决方案：system.out.println 显示具有 bug，在使用虚拟机 textview 输出时结果是正确的。

5. 中缀表达式中判断运算顺序、括号有一定的难度。

解决方案：先将中缀表达式用算法转换为后缀表达式，这样计算就不需要考虑运算顺序和括号问题。

6. 在一开始的 ArrayList<character> 中，将每一个输入的字符都加到 ArrayList 中，这样在遇到多位数以及小数点时，会把整个数分开。

解决方案：再创建一个 ArrayList<String>，对最终的 ArrayList<character> 进行读取处理，将一个数字中的多位数及小数点进行整合成一个 String 再进行存放。

八、不足之处

1. 实际上这一次之所以称之为较为科学的计算器而不是“科学计算器”很大原因就是这一个不足，这一次的计算器中的复杂运算，包括平方、开方、取对数、阶乘等计算只能允许一个数的计算而无法实现包含括号的计算。

主要原因是这些复杂运算的产生使得表达式中出现了多个运算符，而后缀表达式主要是用栈的方式解决一个个操作符之间的问题，我在网上暂时也没有搜到复杂的多个运算符的后缀计算方法，今后改进的方式可以从直接用中缀表达式计算以及多个操作符后缀运算方式入手，实现包含括号的复杂运算。

2. 表达式输入过多时虽然可以上下滚动，但实际上左右滚动的显示效果可能更好一点，但是试了一些时间没有找到成功左右滚动的方法，今后可以再尝试一些方法。

3. 可以设置一个查看最近计算式的功能，这样如果偶尔忘记结果或者清 0 后想回头检查算式时不需要重新输入，而是可以选择显示之前的算式。

4. 以及一个显示最近计算结果的功能，有时候我们计算后想根据计算结果立即进行复杂运算，很多科学计算器都必须重新输入才可以编辑。而原本显示的结果是不可直接饮用并进行接下去的计算的。

5. 实际上我还想加一个光标，可以直接点击更改表达式中的任意顺序，而不只是只

能在最后面加，这样在写表达式时突然想在哪加一个括号就很方便。

九、今后的设想

之前使用过 app inventor 编写计算器，当时我在我的说明文档上写下了很多不足，其中包括：无法实现更加科学的计算，只能实现简单的计算、可以添加一个下拉菜单，拉下来实现一些比较复杂的计算，如平方根，绝对值，开方，开方、无法将结果显示在显示屏上，可以添加一个新的显示屏记录输入的任何数据，删除也会对应删除等等。

而在这次计算器的编写中我便针对上次所指出的不足进行了改进，将简单计算器变成较为复杂的计算器，算是针对上一次的计算器的一次大改进，在算法方面也有了一些进步。这种循序渐进的感觉让我在移动应用开发的学习上方向感更加明确，很感谢老师在课程上的方向指导。

而据我所知，android studio 是基于 IntelliJ IDEA 编写的，个人感觉非常好用，个人感觉 IntelliJ IDEA 之所以还没有和 eclipse 在现在那么普及（不过现在用 IntelliJ IDEA 的人也很多了）很大一部分原因就是收费的问题吧，这个 IntelliJ IDEA 的自动完成功能真的超赞，非常人性化。

在这次 android studio 的学习以及代码的编写中我也感觉自己的能力又有了一些提升，不过未来的路也还很长，我还需要更努力地学习！