

计算器的设计与实现说明文档

完成人：黎炜堂

学号：20202132030

完成时间：2023年4月20日

一、软件名称

科学计算器

二、软件内容简介

这款安卓手机科学计算器是一款功能强大的工具，旨在帮助用户进行各种复杂的科学计算。该应用程序具有丰富的功能和易于使用的界面，可以满足各种计算需求。

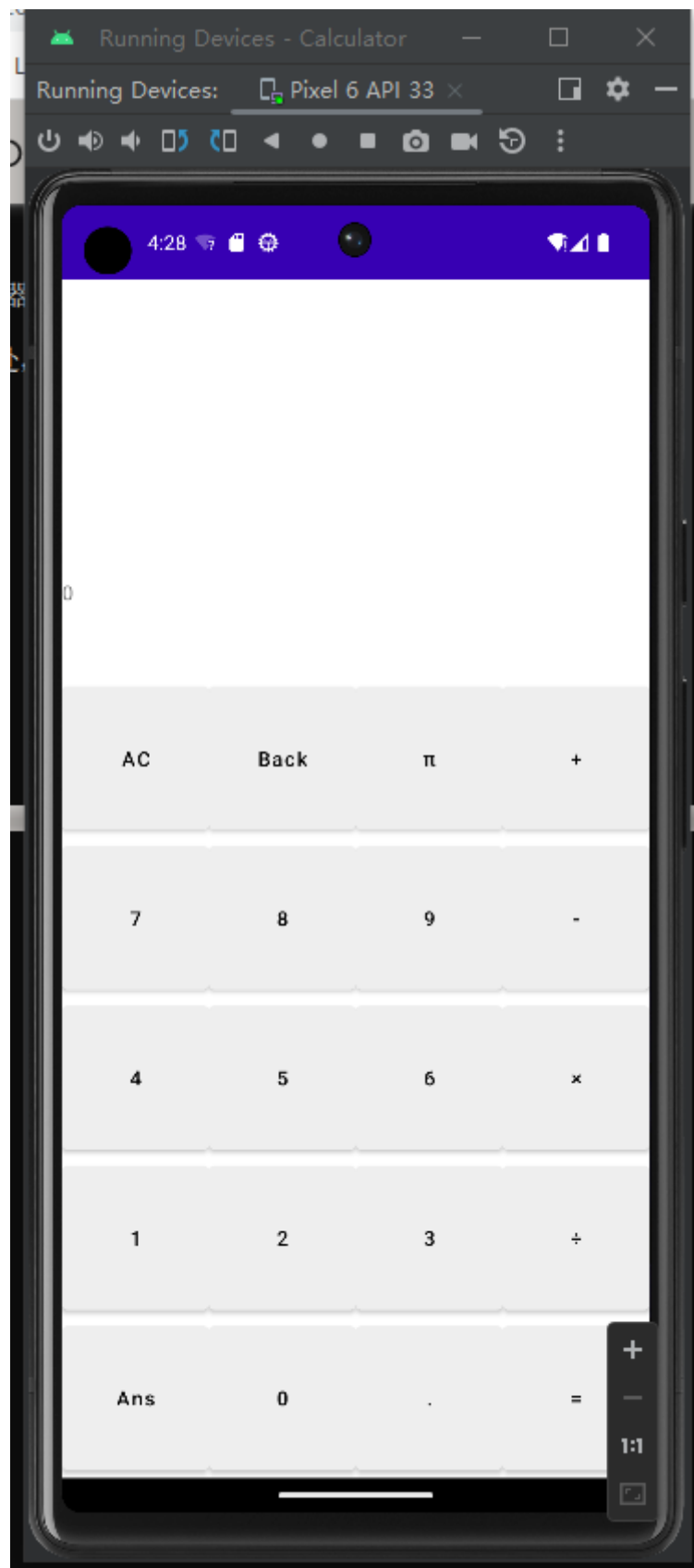
该计算器支持各种基本的数学运算，如加、减、乘、除、幂等。此外，它还支持一些高级数学运算，如三角函数、对数函数、指数函数等。用户可以轻松地进行复杂的数学计算，无需手动计算或使用其他计算工具。

此外，该应用程序具有易于使用的用户界面，使用户可以轻松找到所需的功能。它还具有历史记录功能，可以查看之前进行的计算，以及保存并重新使用以前计算的结果。

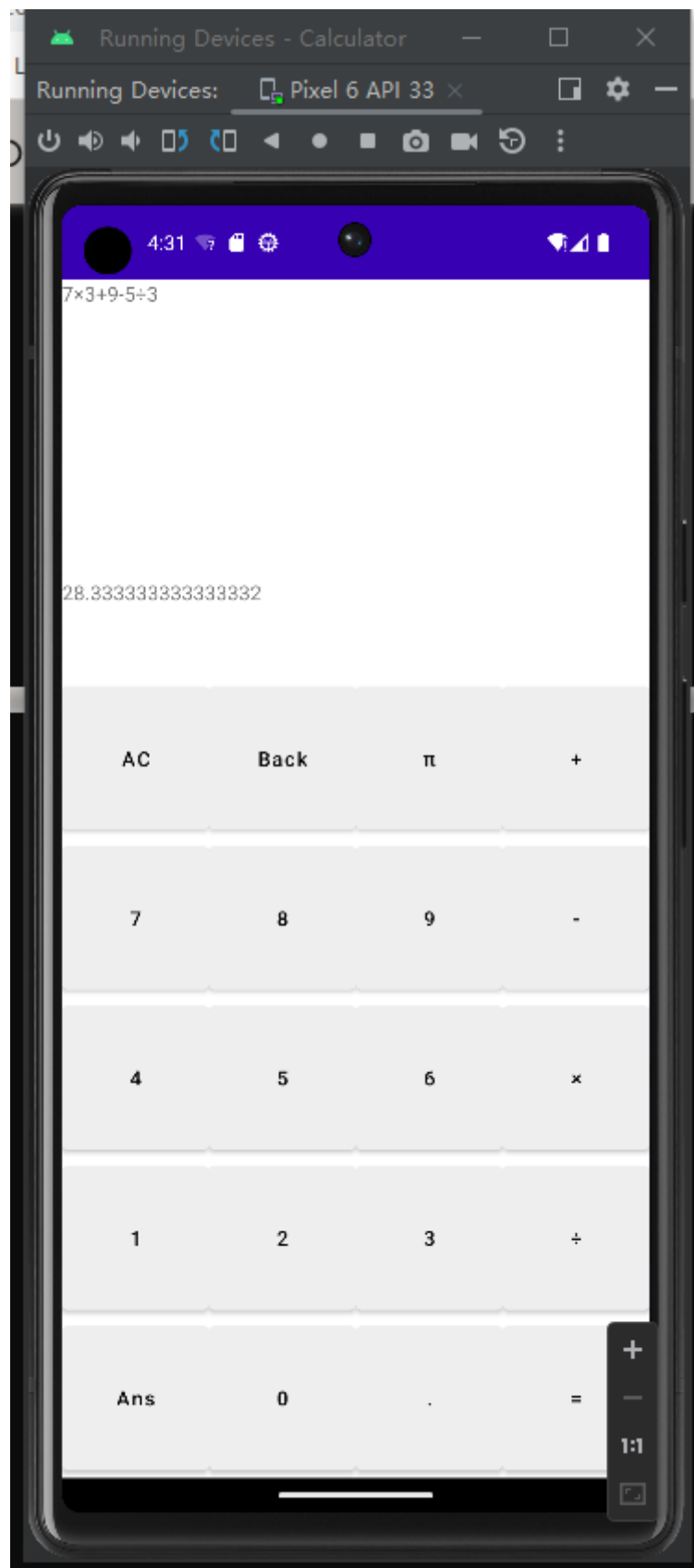
总之，这款安卓手机科学计算器是一款功能丰富、易于使用的应用程序，可以满足各种科学计算需求。

界面设计

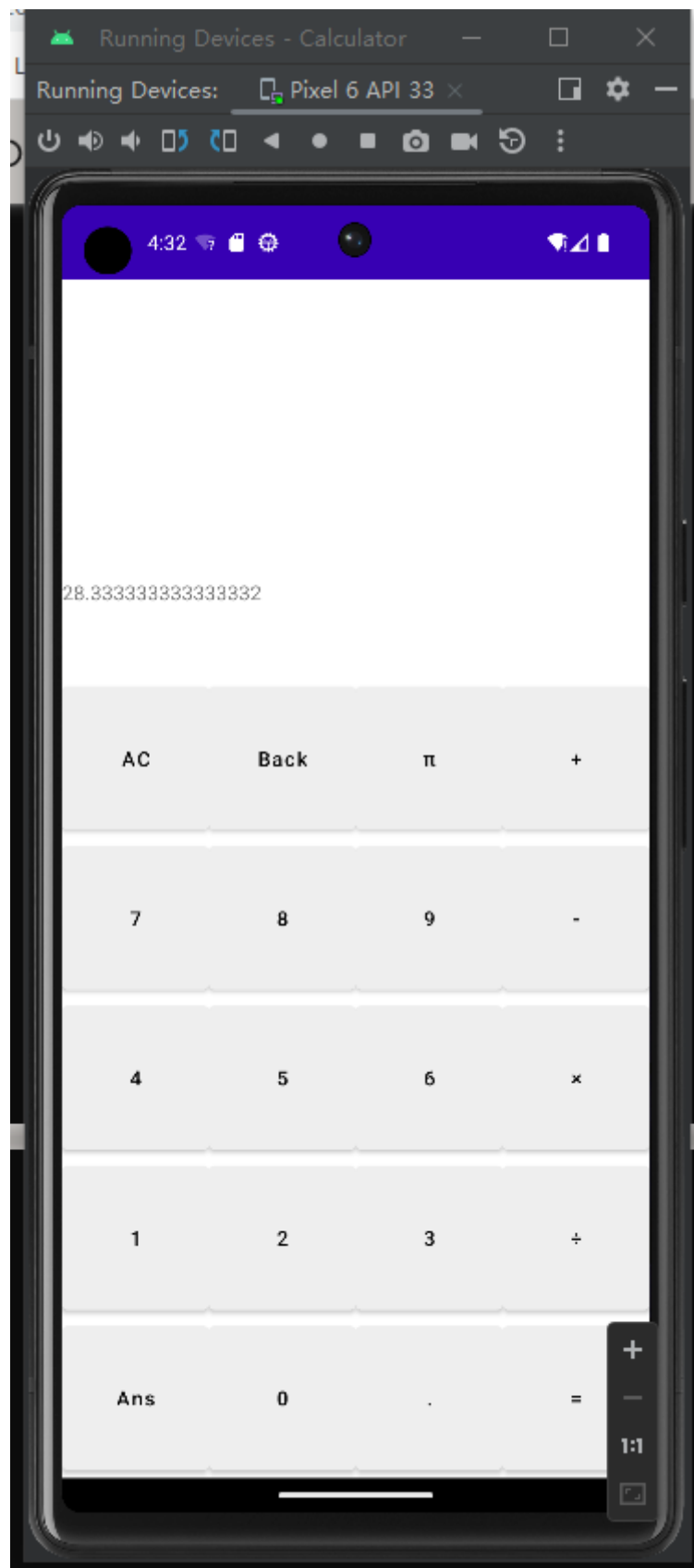
(一) 主界面



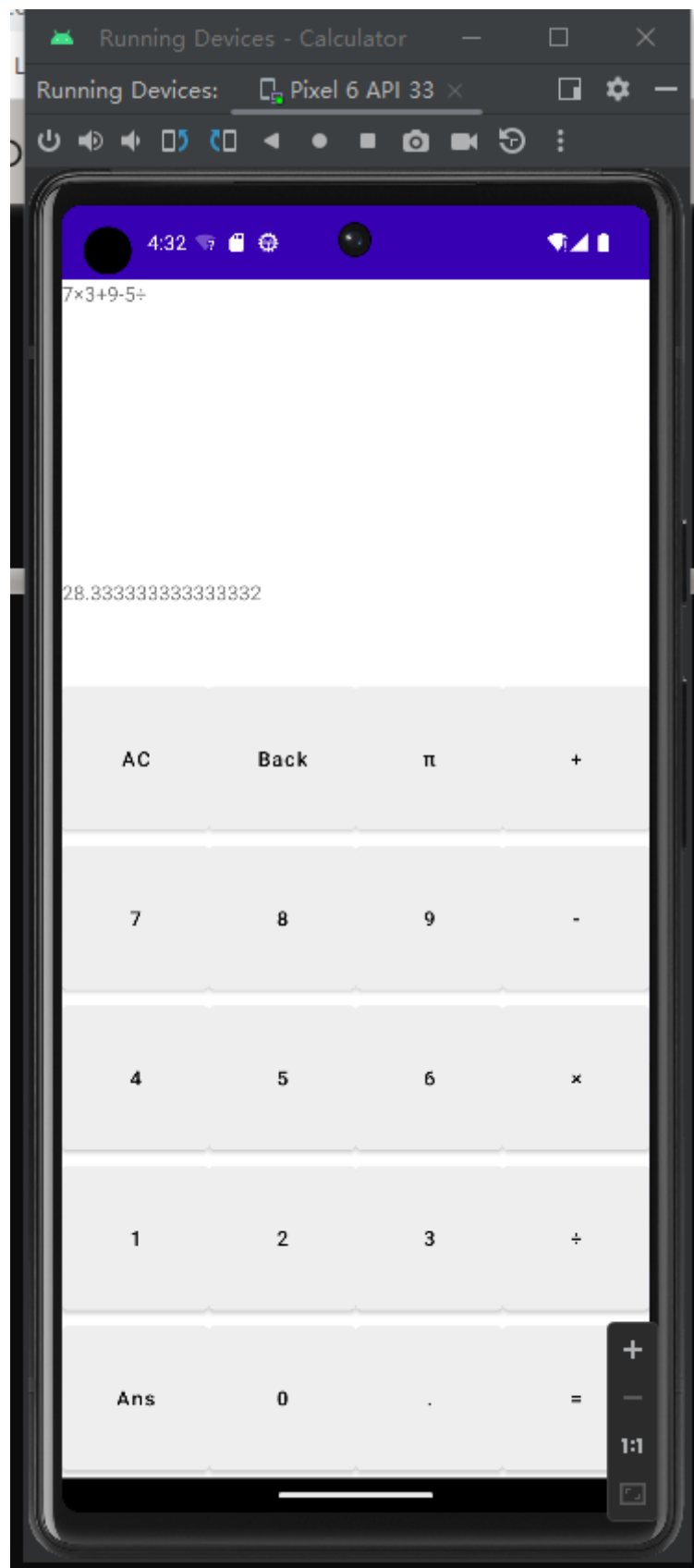
(二) 普通四则运算



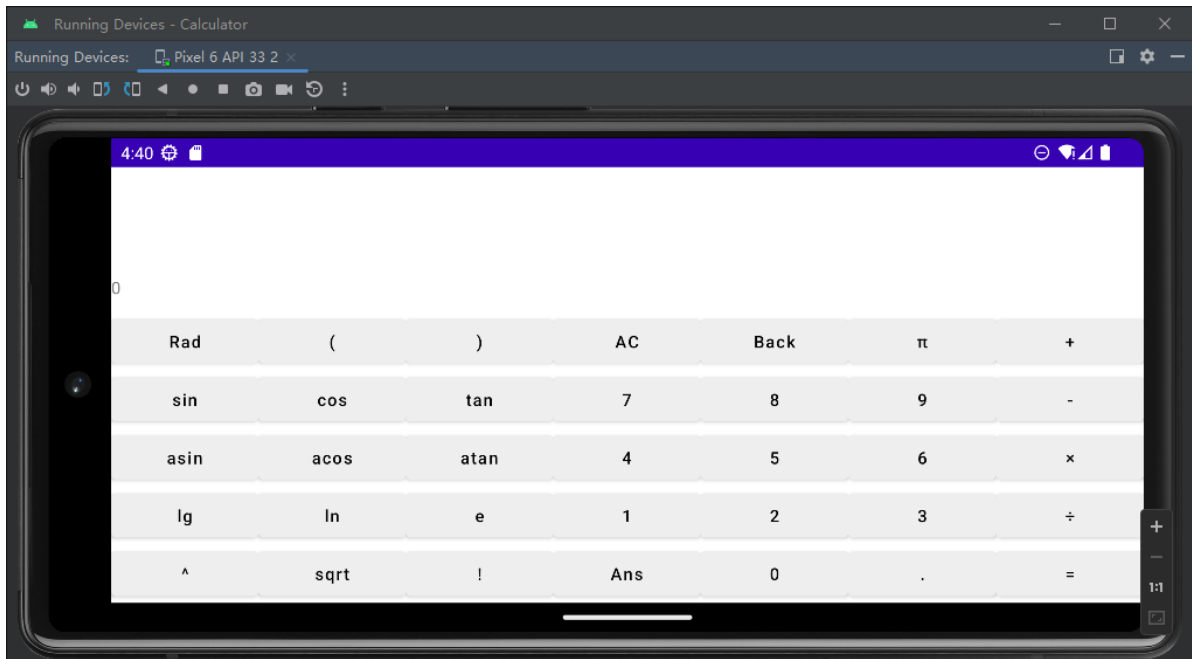
(三) AC键



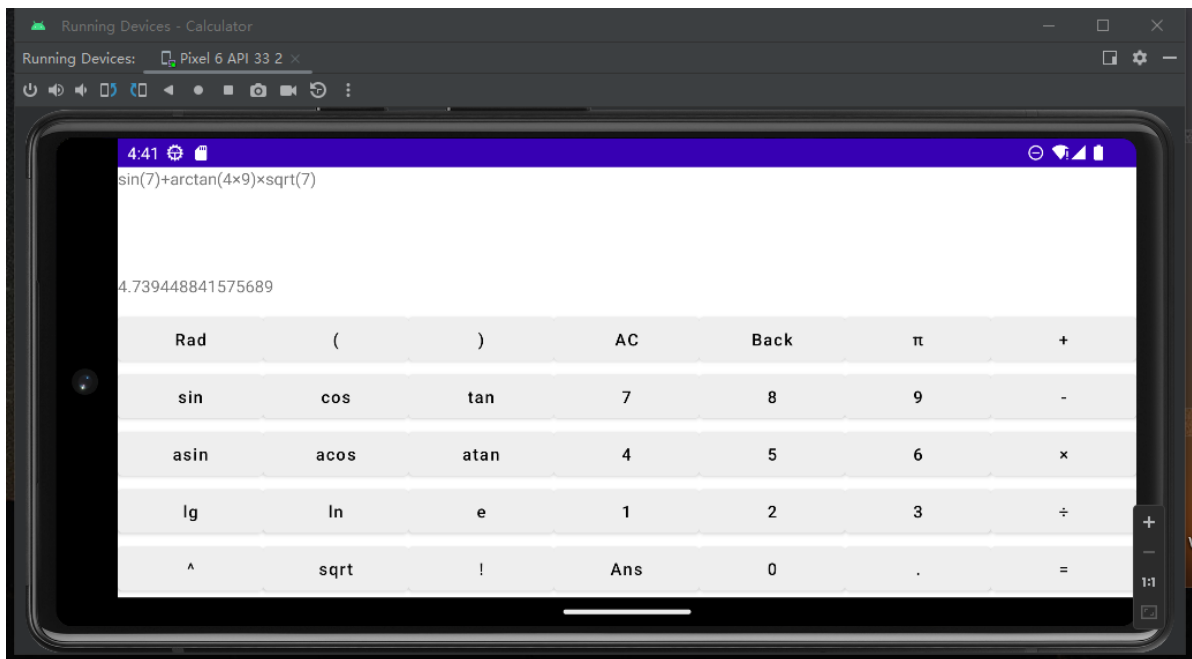
(四) Back键



(五) 横板显示页面



(六) 复杂计算



代码设计

(一) 主界面布局

```
<?xml version="1.0" encoding="utf-8"?>
<LinearLayout xmlns:android="http://schemas.android.com/apk/res/android"
    android:layout_width="match_parent"
    android:layout_height="match_parent"
    android:orientation="vertical">

    <LinearLayout
        android:layout_width="match_parent"
        android:layout_height="0dp"
        android:layout_weight="1"
        android:orientation="vertical">
```

```

<TextView
    android:id="@+id/tv_input"
    android:layout_width="match_parent"
    android:layout_height="0dp"
    android:layout_weight="3" />

<TextView
    android:id="@+id/tv_answer"
    android:layout_width="match_parent"
    android:layout_height="0dp"
    android:layout_weight="1" />
</LinearLayout>

<TableLayout
    android:layout_width="match_parent"
    android:layout_height="0dp"
    android:layout_weight="2">

    <TableRow
        android:layout_width="match_parent"
        android:layout_height="0dp"
        android:layout_weight="1">

        <Button
            android:id="@+id/btn_ac"
            android:layout_width="0dp"
            android:layout_height="match_parent"
            android:layout_weight="1"
            android:backgroundTint="#FFEEEEEE"
            android:text="@string/btn_AC_text"
            android:textAllCaps="false"
            android:textColor="@color/black" />

        <Button
            android:id="@+id/btn_back"
            android:layout_width="0dp"
            android:layout_height="match_parent"
            android:layout_weight="1"
            android:backgroundTint="#FFEEEEEE"
            android:text="@string/btn_back_text"
            android:textAllCaps="false"
            android:textColor="@color/black" />

        <Button
            android:id="@+id/btn_pi"
            android:layout_width="0dp"
            android:layout_height="match_parent"
            android:layout_weight="1"
            android:backgroundTint="#FFEEEEEE"
            android:text="@string/btn_pi_text"
            android:textAllCaps="false"
            android:textColor="@color/black" />

        <Button
            android:id="@+id/btn_add"

```

```

        android:layout_width="0dp"
        android:layout_height="match_parent"
        android:layout_weight="1"
        android:backgroundTint="#FFEEEEEE"
        android:text="@string/btn_plus_text"
        android:textColor="@color/black" />
</TableRow>

<TableRow
    android:layout_width="match_parent"
    android:layout_height="0dp"
    android:layout_weight="1">

    <Button
        android:id="@+id/btn_7"
        android:layout_width="0dp"
        android:layout_height="match_parent"
        android:layout_weight="1"
        android:backgroundTint="#FFEEEEEE"
        android:text="@string/btn_7_text"
        android:textColor="@color/black" />

    <Button
        android:id="@+id/btn_8"
        android:layout_width="0dp"
        android:layout_height="match_parent"
        android:layout_weight="1"
        android:backgroundTint="#FFEEEEEE"
        android:text="@string/btn_8_text"
        android:textColor="@color/black" />

    <Button
        android:id="@+id/btn_9"
        android:layout_width="0dp"
        android:layout_height="match_parent"
        android:layout_weight="1"
        android:backgroundTint="#FFEEEEEE"
        android:text="9"
        android:textColor="@color/black" />

    <Button
        android:id="@+id/btn_minus"
        android:layout_width="0dp"
        android:layout_height="match_parent"
        android:layout_weight="1"
        android:backgroundTint="#FFEEEEEE"
        android:text="-"
        android:textColor="@color/black" />
</TableRow>

<TableRow
    android:layout_width="match_parent"
    android:layout_height="0dp"
    android:layout_weight="1">

```



```
<Button
    android:id="@+id/btn_4"
    android:layout_width="0dp"
    android:layout_height="match_parent"
    android:layout_weight="1"
    android:backgroundTint="#FFEEEEEE"
    android:text="4"
    android:textColor="@color/black" />
```

```
<Button
    android:id="@+id/btn_5"
    android:layout_width="0dp"
    android:layout_height="match_parent"
    android:layout_weight="1"
    android:backgroundTint="#FFEEEEEE"
    android:text="5"
    android:textColor="@color/black" />
```

```
<Button
    android:id="@+id/btn_6"
    android:layout_width="0dp"
    android:layout_height="match_parent"
    android:layout_weight="1"
    android:backgroundTint="#FFEEEEEE"
    android:text="6"
    android:textColor="@color/black" />
```

```
<Button
    android:id="@+id/btn_multiply"
    android:layout_width="0dp"
    android:layout_height="match_parent"
    android:layout_weight="1"
    android:backgroundTint="#FFEEEEEE"
    android:text="@string/btn_multiply_text"
    android:textColor="@color/black" />
```

```
</TableRow>
```

```
<TableRow
    android:layout_width="match_parent"
    android:layout_height="0dp"
    android:layout_weight="1">
```

```
<Button
    android:id="@+id/btn_1"
    android:layout_width="0dp"
    android:layout_height="match_parent"
    android:layout_weight="1"
    android:backgroundTint="#FFEEEEEE"
    android:text="1"
    android:textColor="@color/black" />
```

```
<Button
    android:id="@+id/btn_2"
    android:layout_width="0dp"
    android:layout_height="match_parent"
```

```
        android:layout_weight="1"
        android:backgroundTint="#FFEEEEEE"
        android:text="2"
        android:textColor="@color/black" />
```

<Button

```
        android:id="@+id/btn_3"
        android:layout_width="0dp"
        android:layout_height="match_parent"
        android:layout_weight="1"
        android:backgroundTint="#FFEEEEEE"
        android:text="3"
        android:textColor="@color/black" />
```

<Button

```
        android:id="@+id/btn_divide"
        android:layout_width="0dp"
        android:layout_height="match_parent"
        android:layout_weight="1"
        android:backgroundTint="#FFEEEEEE"
        android:text="@string/btn_divide_text"
        android:textColor="@color/black" />
```

</TableRow>

<TableRow

```
        android:layout_width="match_parent"
        android:layout_height="0dp"
        android:layout_weight="1">
```

<Button

```
        android:id="@+id/btn_ans"
        android:layout_width="0dp"
        android:layout_height="match_parent"
        android:layout_weight="1"
        android:backgroundTint="#FFEEEEEE"
        android:text="Ans"
        android:textAllCaps="false"
        android:textColor="@color/black" />
```

<Button

```
        android:id="@+id/btn_0"
        android:layout_width="0dp"
        android:layout_height="match_parent"
        android:layout_weight="1"
        android:backgroundTint="#FFEEEEEE"
        android:text="0"
        android:textColor="@color/black" />
```

<Button

```
        android:id="@+id/btn_point"
        android:layout_width="0dp"
        android:layout_height="match_parent"
        android:layout_weight="1"
        android:backgroundTint="#FFEEEEEE"
        android:text="."
```

```

        android:textColor="@color/black" />

        <Button
            android:id="@+id/btn_equal"
            android:layout_width="0dp"
            android:layout_height="match_parent"
            android:layout_weight="1"
            android:backgroundTint="#FFEEEEEE"
            android:text="="
            android:textColor="@color/black" />
    </TableRow>
</TableLayout>

</LinearLayout>

```

(=) input

```

package com.blkan.calculator.util;

import android.widget.TextView;

import com.blkan.calculator.Calculator;
import com.blkan.calculator.databinding.ActivityMainBinding;

import java.util.regex.Matcher;
import java.util.regex.Pattern;

public class Input {
    public static StringBuilder input = new StringBuilder("");
    public static String result = "0";
    public static boolean deg = false;

    public static void initButtons(ActivityMainBinding binding, boolean
portrait) {
        updateTextView(binding.tvInput);
        binding.tvAnswer.setText(result);
        binding.btnAc.setOnClickListener(view -> {
            input = new StringBuilder("");
            updateTextView(binding.tvInput);
        });
        binding.btnBack.setOnClickListener(view -> {
            if (input.length() == 0)
                return;
            int cut = backSpace();
            input.delete(input.length() - cut, input.length());
            updateTextView(binding.tvInput);
        });
        binding.btnPi.setOnClickListener(view -> {
            input.append("π");
            updateTextView(binding.tvInput);
        });
        binding.btnAdd.setOnClickListener(view -> {
            addSign("+");
            updateTextView(binding.tvInput);
        });
    }
}

```

```
binding.btn7.setOnClickListener(view -> {
    addNumber(7);
    updateTextView(binding.tvInput);
});
binding.btn8.setOnClickListener(view -> {
    addNumber(8);
    updateTextView(binding.tvInput);
});
binding.btn9.setOnClickListener(view -> {
    addNumber(9);
    updateTextView(binding.tvInput);
});
binding.btnMinus.setOnClickListener(view ->
{
    addSign("-");
    updateTextView(binding.tvInput);
});
binding.btn4.setOnClickListener(view -> {
    addNumber(4);
    updateTextView(binding.tvInput);
});
binding.btn5.setOnClickListener(view -> {
    addNumber(5);
    updateTextView(binding.tvInput);
});
binding.btn6.setOnClickListener(view -> {
    addNumber(6);
    updateTextView(binding.tvInput);
});
binding.btnMultiply.setOnClickListener(view ->
{
    addSign("x");
    updateTextView(binding.tvInput);
});
binding.btn1.setOnClickListener(view -> {
    addNumber(1);
    updateTextView(binding.tvInput);
});
binding.btn2.setOnClickListener(view -> {
    addNumber(2);
    updateTextView(binding.tvInput);
});
binding.btn3.setOnClickListener(view -> {
    addNumber(3);
    updateTextView(binding.tvInput);
});
binding.btnDivide.setOnClickListener(view ->
{
    addSign("÷");
    updateTextView(binding.tvInput);
});
binding.btnAns.setOnClickListener(view -> {
    input.append("Ans");
    updateTextView(binding.tvInput);
});
```

```

binding.btn0.setOnClickListener(view -> {
    addNumber(0);
    updateTextView(binding.tvInput);
});
binding.btnPoint.setOnClickListener(view -> {
    if (input.length() == 0)
        input.append(0);
    char c = input.charAt(input.length() - 1);
    if (c >= '0' && c <= '9') {
        input.append(".");
        updateTextView(binding.tvInput);
    }
});
binding.btnEqual.setOnClickListener(view -> {
    String s = input.toString();
    Pattern pattern = Pattern.compile("x");
    Matcher matcher = pattern.matcher(s);
    s = matcher.replaceAll("*");

    pattern = Pattern.compile("÷");
    matcher = pattern.matcher(s);
    s = matcher.replaceAll("/");

    pattern = Pattern.compile("(?<=\\A|\\+|-|\\*|\\/|\\^)(Ans)");
    matcher = pattern.matcher(s);
    s = matcher.replaceAll('(' + result + ')');

    pattern = Pattern.compile("Ans");
    matcher = pattern.matcher(s);
    s = matcher.replaceAll("*" + '(' + result + ')');

    result = Calculator.getInstance().calExpression(s, deg, false);
    if (result.endsWith(".0"))
        result = result.substring(0, result.length() - 2);
    binding.tvAnswer.setText(result);
});
if (portrait)
    return;
if (deg)
    binding.btnRad.setText("Deg");
binding.btnRad.setOnClickListener(view -> {
    deg = !deg;
    if (deg)
        binding.btnRad.setText("Deg");
    else
        binding.btnRad.setText("Rad");
});
binding.btnLeftP.setOnClickListener(view -> {
    if (input.length() != 0) {
        char c = input.charAt(input.length() - 1);
        if ((c >= '0' && c <= '9') || c == ')' || c == 'e' || c == 'π'
|| c == 's' || c == '!') {
            input.append("x");
        }
    }
}

```

```
        input.append('(');
        updateTextView(binding.tvInput);
    });
    binding.btnRightP.setOnClickListener(view -> {
        if (input.length() != 0) {
            input.append(')');
            updateTextView(binding.tvInput);
        }
    });
    binding.btnSin.setOnClickListener(view -> {
        input.append("sin(");
        updateTextView(binding.tvInput);
    });
    binding.btnCos.setOnClickListener(view -> {
        input.append("cos(");
        updateTextView(binding.tvInput);
    });
    binding.btnTan.setOnClickListener(view -> {
        input.append("tan(");
        updateTextView(binding.tvInput);
    });
    binding.btnArcsin.setOnClickListener(view -> {
        input.append("arcsin(");
        updateTextView(binding.tvInput);
    });
    binding.btnArccos.setOnClickListener(view -> {
        input.append("arccos(");
        updateTextView(binding.tvInput);
    });
    binding.btnArctan.setOnClickListener(view -> {
        input.append("arctan(");
        updateTextView(binding.tvInput);
    });
    binding.btnLg.setOnClickListener(view -> {
        input.append("lg(");
        updateTextView(binding.tvInput);
    });
    binding.btnLg.setOnClickListener(view -> {
        input.append("lg(");
        updateTextView(binding.tvInput);
    });
    binding.btnLn.setOnClickListener(view -> {
        input.append("ln(");
        updateTextView(binding.tvInput);
    });
    binding.btnE.setOnClickListener(view -> {
        input.append('e');
        updateTextView(binding.tvInput);
    });
    binding.btnPow.setOnClickListener(view -> {
        addSign("^(");
        updateTextView(binding.tvInput);
    });
    binding.btnSqrt.setOnClickListener(view -> {
        input.append("sqrt(");
```

```

        updateTextView(binding.tvInput);
    });
    binding.btnFact.setOnClickListener(view -> {
        if (input.length() == 0)
            return;
        char c = input.charAt(input.length() - 1);
        if (c == '(' || c == '!')
            return;
        if (c == '+' || c == '-' || c == 'x' || c == '÷') {
            input.deleteCharAt(input.length() - 1);
        }
        input.append('!');
        updateTextView(binding.tvInput);
    });
}

private static void addNumber(int number) {
    if (input.length() != 0) {
        char c = input.charAt(input.length() - 1);
        if (c == ')' || c == 'e' || c == 'π' || c == 's' || c == '!') {
            input.append("x");
        }
    }
    input.append(number);
}

private static void addSign(String sign) {
    if (input.length() == 0)
        input.append(0);
    else {
        char c = input.charAt(input.length() - 1);
        if (c == '+' || c == '-' || c == 'x' || c == '÷') {
            input.deleteCharAt(input.length() - 1);
        }
    }
    input.append(sign);
}

private static int backspace() {
    String s = input.toString();
    if (s.endsWith("^("))
        return 2;
    if (s.endsWith("lg(") || s.endsWith("ln(") || s.endsWith("Ans"))
        return 3;
    if (s.endsWith("sin(") || s.endsWith("cos(") || s.endsWith("tan("))
        return 4;
    if (s.endsWith("sqrt("))
        return 5;
    if (s.endsWith("arcsin(") || s.endsWith("arccos(") ||
s.endsWith("arctan("))
        return 7;
    return 1;
}

private static void updateTextView(Textview tv) {

```

```
        tv.setText(input.toString());
    }
}
```

(三) MainActivity

```
package com.blkan.calculator;

import androidx.appcompat.app.ActionBar;
import androidx.appcompat.app.AppCompatActivity;
import androidx.appcompat.widget.LinearLayoutCompat;

import android.content.res.Configuration;
import android.os.Bundle;
import android.view.View;
import android.widget.Button;
import android.widget.TextView;

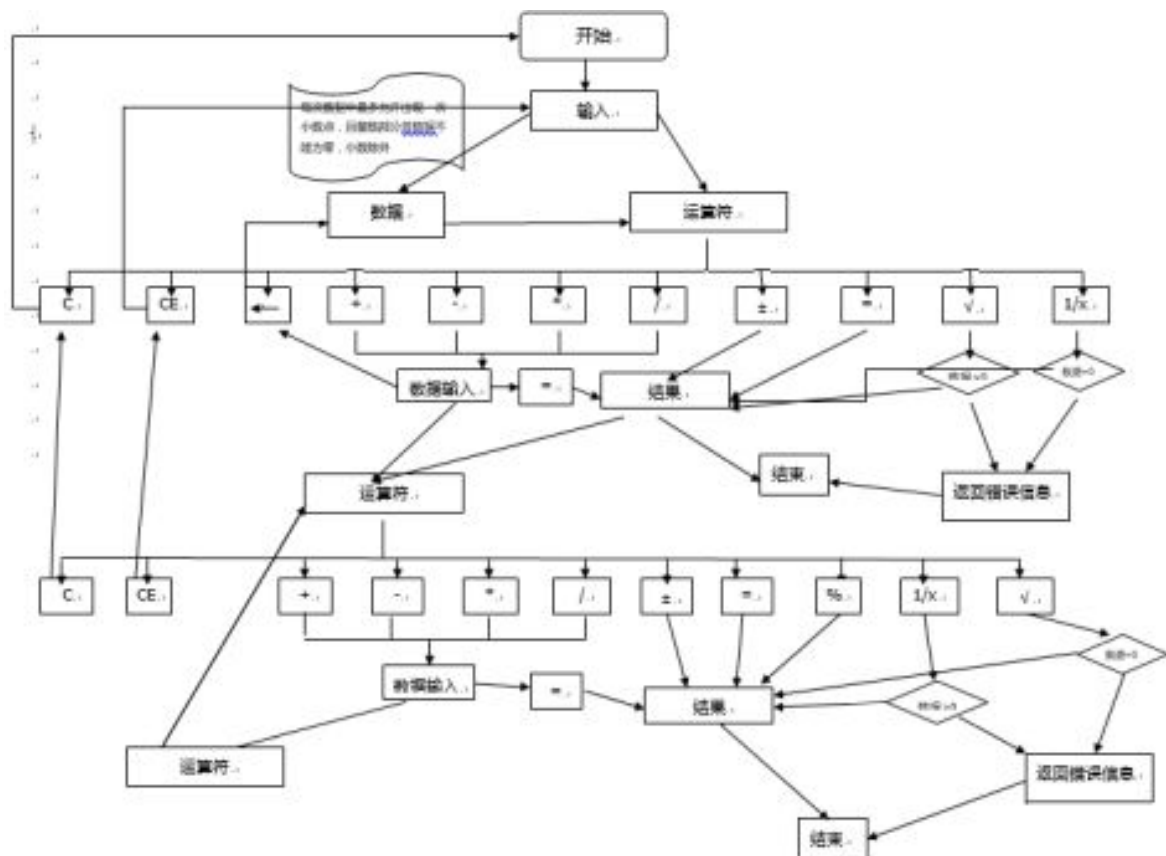
import com.blkan.calculator.databinding.ActivityMainBinding;
import com.blkan.calculator.util.Input;

public class MainActivity extends AppCompatActivity {

    private ActivityMainBinding binding;

    @Override
    protected void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);
        binding = ActivityMainBinding.inflate(getLayoutInflater());
        setContentView(binding.getRoot());
        ActionBar actionBar = getSupportActionBar();
        if (actionBar != null)
            actionBar.hide();
        int orientation = getResources().getConfiguration().orientation;
        Input.initButtons(binding, orientation ==
        Configuration.ORIENTATION_PORTRAIT);
    }
}
```

软件操作流程



对软件的说明

调用

```
calExpression(String content, boolean printProcess)
```

(一) 表达式组成：

1. e和 π
2. 函数sin() cos() tan()
3. 函数arcsin() arccos() arctan()
4. 函数sqrt()
5. 函数lg() ln()
6. 阶乘!
7. ^
8. +-* /
9. ()

(二) 特殊规定

加[]

计算过程中，每完成一步计算，需将计算出来的数字结果用[]括起来（[]里面的一定是数字）。

[]的作用与()类似，只是不标定运算顺序。

比如，对于表达式：1+(0-2)^2

如果不加[]，那么计算到下一步就是：1-2^2=-3

而如果加了[]，那么就是：1+[-2]^2=5

无圆括号表达式

在加[]的前提下，如果一个表达式没有任何()（可以有[]），那么就规定为**无圆括号表达式**。

（三）计算无圆括号表达式

要计算无圆括号表达式的值，可以按照以下方法：

1. 替换e和 π
2. 计算并替换sin[]、cos[]、tan[]
3. 计算并替换arcsin[]、arccos[]、arctan[]
4. 计算并替换lg[]、ln[]
5. 计算并替换sqrt[]
6. 替换二元运算的减号（注意不替换负数的-号）
7. 计算并替换阶乘!
8. 计算次方^
9. 计算乘除*/
10. 计算加+（-已在第5步被替换）

注：由于数学表达式中存在数字*字母可以省略*号的情况，所以进行前四个替换时，需要判断前一个字符的类型！如果是字符串开头，+*/，则直接替换成结果数字；其余情况需要在前面补上一个*号。

除了1.，每完成一步计算，需将计算出来的数字结果用[]括起来。

1. 替换e和 π

```
public String calEandPI(String content)
```

替换成String.valueOf(Math.E)和String.valueOf(Math.PI)。

由于e和 π 绝不是负数，所以不用加[]。

例：e+2e+ π

2.718281828459045+2*2.718281828459045+3.141592653589793

2. 计算并替换sin[]、cos[]、tan[]

```
public String calSin(String content)
```

利用regex捕获[]里面的数字，提取为double类型，传进Math.sin()计算；

替换成String.valueOf(Math.sin(数))。

例：sin[1]-cos[2]+3tan[5]

[0.8414709848078965]-[-0.4161468365471424]+3*[-3.380515006246586]

3. 计算并替换arcsin[]、arccos[]、arctan[]

```
public String calArcsin(String content)
```

与2.类似。

4. 计算并替换lg[], ln[]

```
public String calLog(String content)
```

原理如上。

例: $\lg[10]+3\lg[10]+5*\ln[5]$

$[1.0]+3[1.0]+5[1.6094379124341003]$

5.计算并替换sqrt[]

```
public String calSqrt(String content)
```

略

6. 替换二元运算的减号

```
public String calMinus(String content)
```

为了防止进行后面的步骤出错，需要将减号（不是负号）替换成负号，即-变成+[-1]*

例如: $1-2^2$

如果不替换，在进行^运算时会出现将1-2识别为一个数，导致运算错误。

如果替换，则变成了 $1+[-1]2^2=1+[-1]4=1+[-4]=[-3]$

7. 计算并替换阶乘!

```
public String calFact(String content)
```

识别!的前面一个数字，如果数字是正整数，则计算阶乘，如果数字包含-或.，则结果为NaN。

8. 计算次方^

```
public String calPow(String content)
```

获取^左边的数a和右边的数b;

计算 $\text{Math.pow}(a, b)$ 。

计算乘除*/

类似8.

10. 计算加+

类似上面。

