**实 验 报 告**

**课程名称**  移 动 应 用 系统

**实验项目**  实验四：函数图形绘制器

**实验仪器** 计算机一台、Android手机一部

**系 别**  软件工程

**专 业** 软件工程

**班级/学号**  软工1801 /2018011192

**学生姓名** 刘宏远

**实验日期**  2020/11/15

**成 绩**

**指导教师**  郝 保 水

# 实验四 函数图形绘制器

1. 实验目的
2. 了解Android环境的搭建；
3. 理解并掌握layout;
4. 理解并掌握常见的控件使用方法；
5. 理解并掌握菜单；
6. 理解并掌握事件处理机制；
7. 理解并掌握Intent使用方法；

2实验要求

1. 课前预习实验内容，并查找相关资料。
2. 按照实验步骤完成各个相关内容。
3. 撰写实验报告。
   1. 实验报告格式必须符合学校要求（例如必须采用学校规定的实验封面）；
   2. 写出实验详细步骤，包括主要采用的技术方案、相关分析和核心代码。注意：不要简单地近包括截屏和代码，完整代码可以作为附录放在实验报告结尾；
   3. 总结实验中遇到的问题、分析和解决方法。
   4. 写出心得体会与收获等。

3实验步骤

主要分为几个大步骤进行实现。

1. 需求分析，确定界面。
2. 系统设计，对函数绘图进行模块划分，确定技术方案。
3. 编码，编写代码实现各项功能。
4. 测试。

3.1需求分析

图形界面绘制，用户自定义函数，坐标轴，函数曲线

3.2系统设计

Expression：表达式求值

支持+、-、\*、/、%、^(幂)

支持一元加和一元减

支持括号

支持PI

支持sin、cos、tg、ctg等三角函数

例如：sin(2)(0.03489949670250097164599518162533)

ExpressionWithVars

继承自Expression，支持变量，对变量给定能够计算

sin(x)(x=2,首先将表达式中所有的x替换成2，变成sin(2)，然后对sin(2)求值

Axis:坐标轴

坐标轴范围

x:minX,maxX

y:minY,maxY

坐标轴颜色

坐标轴上的数字

原点

x轴两端

y轴两端

Plot：图形绘制

对于指定表达式(sin(x))绘制

绘制的颜色

绘制的范围

custom：函数绘制

Axis:坐标轴

Plot:函数曲线绘制

MainActivity：主界面

获取用户输入的函数

绘制函数

3.3编码实现

**public class** Axis {  
 *//逻辑范围* **private int nMinX**;  
 **private int nMaxX**;  
 **private int nMinY**;  
 **private int nMaxY**;  
  
 *//物理范围* **private** Rect **mRect**;  
  
 **public** Axis(Rect rect) {  
 **nMinX** = -10;  
 **nMaxX** = 10;  
 **nMinY** = -10;  
 **nMaxY** = 10;  
 **mRect** = rect;  
 }  
  
 *//将逻辑坐标转换为物理坐标* **public int** convertXLP2DP(**double** x) {  
 **return mRect**.**left** + (**int**) (**mRect**.width() / (**double**) (**nMaxX** - **nMinX**) \* (x - **nMinX**));  
 }  
  
 *//将逻辑坐标转换为物理坐标* **public int** convertYLP2DP(**double** y) {  
 **return mRect**.**bottom** - (**int**) (**mRect**.height() / (**double**) (**nMaxY** - **nMinY**) \* (y - **nMinX**));  
 }

//绘制坐标轴  
 public void draw(Canvas canvas) {  
 //画笔  
 Paint paint = new Paint();  
 paint.setAntiAlias(true);  
 paint.setColor(Color.BLUE);  
 paint.setStyle(Paint.Style.STROKE);  
 paint.setStrokeWidth(5);  
  
 double nX=(double)(nMaxX-nMinX)/20;  
 double nY=(double)(nMaxY-nMinY)/20;  
  
 //绘制坐标轴  
 canvas.drawLine(convertXLP2DP(nMinX), convertYLP2DP(0), convertXLP2DP(nMaxX), convertYLP2DP(0), paint);//x轴  
 canvas.drawLine(convertXLP2DP(0), convertYLP2DP(nMaxY), convertXLP2DP(0), convertYLP2DP(nMinY), paint);//y轴  
  
  
  
 //绘制坐标轴上的坐标（数字）  
 paint.setStrokeWidth(1);  
 paint.setTextSize(9);  
 canvas.drawText("0", convertXLP2DP(nX), convertYLP2DP(-nY), paint);//原点  
 canvas.drawText(nMinX + "", convertXLP2DP(nMinX+nX), convertYLP2DP(-nY), paint);//x最小  
 canvas.drawText(nMinX+"",convertXLP2DP(nMaxX-nX), convertYLP2DP(-nY),paint);//x最大  
 canvas.drawText(nMinY+"",convertXLP2DP(-nX), convertYLP2DP(nMinY+nY),paint);//y最小  
 canvas.drawText(nMinY+"",convertXLP2DP(-nX), convertYLP2DP(nMaxY-nY),paint);//y最大  
 }  
}

public class Plot {  
 //定义方程图形的表达式、自变量和自变量范围，例如绘制sin(x)曲线，自变量是x，绘制的范围为[-10,10]  
 private String m\_strExp;//表达式，例如sin(x)  
 private String m\_strvar;//表达式中的自变量，例如x  
 private int m\_nMinX;//自变量最小值  
 private int m\_nMaxX;//自变量最大值  
 private int m\_cLine;//图形颜色  
 private Axis mAxis;//坐标轴，曲线在此坐标轴下进行绘制  
  
 public void draw(Canvas canvas)  
 {  
 if(mAxis==null)  
 return;  
 //画笔  
 Paint paint = new Paint();  
 paint.setAntiAlias(true);  
 paint.setColor(m\_cLine);  
 paint.setStyle(Paint.Style.STROKE);  
 paint.setStrokeWidth(1);  
  
 ExpressionWithVars e = new ExpressionWithVars(m\_strExp, m\_strvar);  
 double oldx= m\_nMinX;  
 double oldy=e.evalf(m\_nMinX);  
 double delta = ((double)(m\_nMaxX - m\_nMinX)) / 100;  
  
 //绘制方程图形，将图形分为100点，依次连接起来  
 for (int i = 1; i < 100; i++)  
 {  
 double newx = (m\_nMinX + delta \* i);  
 double newy = e.evalf(newx);  
 canvas.drawLine(mAxis.convertXLP2DP(oldx),mAxis.convertYLP2DP(oldy),

mAxis.convertXLP2DP(newx),mAxis.convertYLP2DP(newy),paint);  
 oldx=newx;  
 oldy=newy;  
 }  
 }  
}

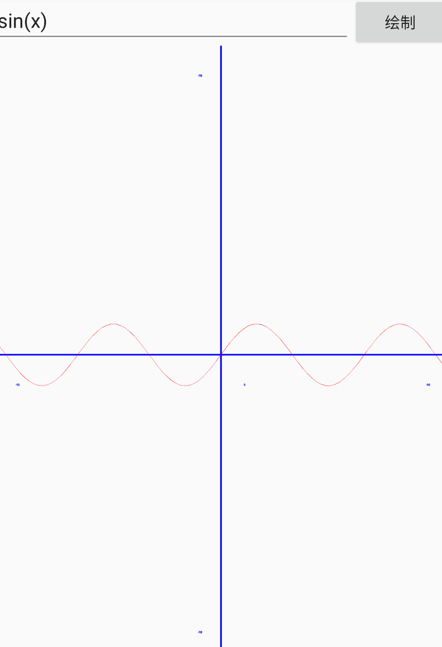
**public class** CustomView **extends** View {  
 String **strFunction**;  
  
 @Override  
 **protected void** onDraw(Canvas canvas) {  
 **if**(**strFunction**==**null** || **strFunction**.length()==0)  
 **return**;

Rect rect=**new** Rect();  
 rect.**left**=0;  
 rect.**top**=0;  
 rect.**right** = getWidth();  
 rect.**bottom**=getHeight();  
  
 Axis axis=**new** Axis(rect);  
 axis.draw(canvas);

Plot plot=**new** Plot(axis,**strFunction**,**"x"**);  
 plot.draw(canvas);  
 }  
}

**public class** MainActivity **extends** AppCompatActivity {  
  
 @Override  
 **protected void** onCreate(Bundle savedInstanceState) {  
 **super**.onCreate(savedInstanceState);  
 setContentView(R.layout.***activity\_main***);  
  
 **final** TextView txtFunction=(TextView)findViewById(R.id.***txtFunction***);  
 **final** CustomView customView=(CustomView)findViewById(R.id.***plotview***);  
 Button buttonPlot=(Button)findViewById(R.id.***buttonPlot***);  
 buttonPlot.setOnClickListener(**new** View.OnClickListener() {  
 @Override  
 **public void** onClick(View v) {  
 **if**(customView!=**null**){  
 String strFunction=txtFunction.getText().toString();  
 customView.setStrFunction(strFunction);  
 customView.invalidate();  
 }  
 }  
 });  
 }  
}

3.4系统测试



4.问题分析（也可以放在实验步骤中写，即写了相关技术或代码后，运行时发现问题，对问题进行分析讨论）

写出问题涉及的相关技术或知识点、问题分析、解决办法等。

运行老师的代码并未发现问题

5.实验心得与体会

收获： 学会了函数绘制

心得： 要学会借鉴他人成果

总结： 要站在巨人的肩膀上

实验报告一定要注意：

1.不要抄袭雷同，一旦发现，不区分抄袭者和被抄袭者，一律0分；

注意，一定不要雷同，不要抄袭！

注意，一定不要雷同，不要抄袭！

注意，一定不要雷同，不要抄袭！

2.格式一定要正确，不要缺少封皮、实验目的、实验内容步骤、心得等；

3，不要代码截屏，我通过github看大家代码，不用为了显得篇幅长而在实验报告中大量粘贴代码；

4.按时提交，过了时间点后系统不允许提交。