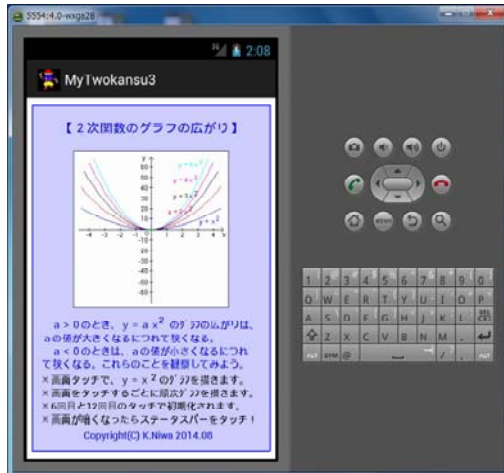
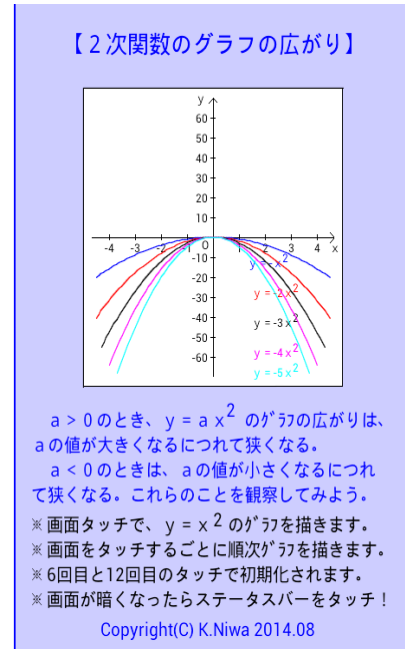


```
//-----
//
//          2次関数のグラフの広がり
//           $y = a x^2$  のグラフ
//          Ver3
//          Copyright (C) K.Niwa 2014. 08. 09
//-----
```



【エミュレータ画面例】

【スマートフォン画面例】
(Android 4.0)

【アプリの概要】

2次関数 $y = a x^2$ のグラフの広がり、 $a > 0$ のとき a の値が大きくなるにつれて狭くなり、 $a < 0$ のとき a の値が小さくなるにつれて狭くなることを観察してみよう。

【1】 Twokansu3.java

```
package jp.seitoku.twokansu3;    //パッケージを指定

import android.content.Context;  //ライブラリを読み込む
import android.graphics.*;
import android.util.AttributeSet;
import android.view.*;

public class Twokansu3 extends View {    //Viewクラスを継承したTwokan1クラス

    int flag=0;    //グラフの移動(1)、グラフの停止(2)、グラフの初期化(0)初期化 識別子
    double x, y;    //グラフ描写に利用
    int px, py, oldpx, oldpy;    //グラフ描写に利用
    int fg;    //グラフ描写に利用
    double a=0, b=0;    //グラフ描写に利用

    public Twokansu3(Context context, AttributeSet attrs, int defStyle) {    //コンストラクタ (クラス名と同じメソッドで最初に読まれる)
        super(context, attrs, defStyle);
    }

    public Twokansu3(Context context, AttributeSet attrs) {    //コンストラクタ (クラス名と同じメソッドで最初に読まれる)
        super(context, attrs);
    }

    public Twokansu3(Context context) {    //コンストラクタ (クラス名と同じメソッドで最初に読まれる)
        super(context);
    }
}
```

```

//onDrawメソッド-----
@Override
protected void onDraw(Canvas canvas) {

    super.onDraw(canvas);
    canvas.drawColor(Color. WHITE);
    Paint paint = new Paint();
    paint.setColor(Color. BLUE);
    paint.setAlpha(50);
    canvas.drawRect((getWidth()/2-240)+10, (getHeight()/2-343)+10, (getWidth()/2-240)+470, (getHeight()/2-343)+675, paint);
    paint.setAlpha(10000);
    paint.setColor(Color. BLUE);

    for (int i=0;i<2;i++) { //額縁を付ける
        canvas.drawLine((getWidth()/2-240)+10+i, (getHeight()/2-343)+10+i, (getWidth()/2-240)+10+i, (getHeight()/2-343)+675-i, paint);
        canvas.drawLine((getWidth()/2-240)+10+i, (getHeight()/2-343)+675-i, (getWidth()/2-240)+470-i, (getHeight()/2-343)+675-i, paint);
        canvas.drawLine((getWidth()/2-240)+470-i, (getHeight()/2-343)+675-i, (getWidth()/2-240)+470-i, (getHeight()/2-343)+10+i, paint);
        canvas.drawLine((getWidth()/2-240)+470-i, (getHeight()/2-343)+10+i, (getWidth()/2-240)+10+i, (getHeight()/2-343)+10+i, paint);
    }

    paint.setColor(Color. BLACK); //実験枠の描画
    canvas.drawRect((getWidth()/2-240)+90, (getHeight()/2-343)+100, (getWidth()/2-240)+390, (getHeight()/2-343)+400, paint);
    paint.setColor(Color. WHITE);
    canvas.drawRect((getWidth()/2-240)+91, (getHeight()/2-343)+101, (getWidth()/2-240)+389, (getHeight()/2-343)+399, paint);

    paint.setColor(Color. BLACK); //座標軸の描画
    canvas.drawLine((getWidth()/2-240)+100, (getHeight()/2-343)+250, (getWidth()/2-240)+380, (getHeight()/2-343)+250, paint);
    canvas.drawLine((getWidth()/2-240)+380, (getHeight()/2-343)+250, (getWidth()/2-240)+380-5, (getHeight()/2-343)+250-5, paint);
    canvas.drawLine((getWidth()/2-240)+380, (getHeight()/2-343)+250, (getWidth()/2-240)+380-5, (getHeight()/2-343)+250+5, paint);
    canvas.drawLine((getWidth()/2-240)+240, (getHeight()/2-343)+110, (getWidth()/2-240)+240, (getHeight()/2-343)+390, paint);
    canvas.drawLine((getWidth()/2-240)+240, (getHeight()/2-343)+110, (getWidth()/2-240)+240+5, (getHeight()/2-343)+110+5, paint);
    canvas.drawLine((getWidth()/2-240)+240, (getHeight()/2-343)+110, (getWidth()/2-240)+240-5, (getHeight()/2-343)+110+5, paint);

    for (int xx=120;xx<=360;xx=xx+30) { // x 軸メモリの描写
        canvas.drawLine((getWidth()/2-240)+xx, (getHeight()/2-343)+250-3, (getWidth()/2-240)+xx, (getHeight()/2-343)+250+3, paint);
    }
    for (int yy=130;yy<=370;yy=yy+20) { // y 軸メモリの描写
        canvas.drawLine((getWidth()/2-240)+240-3, (getHeight()/2-343)+yy, (getWidth()/2-240)+240+3, (getHeight()/2-343)+yy, paint);
    }

    // x 軸メモリの描写
    canvas.drawText("1", (getWidth()/2-240)+270-3, (getHeight()/2-343)+260+5, paint);
    canvas.drawText("2", (getWidth()/2-240)+300-3, (getHeight()/2-343)+260+5, paint);
    canvas.drawText("3", (getWidth()/2-240)+330-3, (getHeight()/2-343)+260+5, paint);
    canvas.drawText("4", (getWidth()/2-240)+360-3, (getHeight()/2-343)+260+5, paint);
    canvas.drawText("x", (getWidth()/2-240)+375, (getHeight()/2-343)+260+5, paint);
    canvas.drawText("0", (getWidth()/2-240)+230-3, (getHeight()/2-343)+260+2, paint);
    canvas.drawText("-1", (getWidth()/2-240)+210-6, (getHeight()/2-343)+260+5, paint);
    canvas.drawText("-2", (getWidth()/2-240)+180-6, (getHeight()/2-343)+260+5, paint);
    canvas.drawText("-3", (getWidth()/2-240)+150-6, (getHeight()/2-343)+260+5, paint);
    canvas.drawText("-4", (getWidth()/2-240)+120-6, (getHeight()/2-343)+260+5, paint);
}

```

```

// y 軸メモリの描写
canvas.drawText("60", (getWidth()/2-240)+220, (getHeight()/2-343)+130+5, paint);
canvas.drawText("50", (getWidth()/2-240)+220, (getHeight()/2-343)+150+5, paint);
canvas.drawText("40", (getWidth()/2-240)+220, (getHeight()/2-343)+170+5, paint);
canvas.drawText("30", (getWidth()/2-240)+220, (getHeight()/2-343)+190+5, paint);
canvas.drawText("20", (getWidth()/2-240)+220, (getHeight()/2-343)+210+5, paint);
canvas.drawText("10", (getWidth()/2-240)+220, (getHeight()/2-343)+230+5, paint);
canvas.drawText("y", (getWidth()/2-240)+220, (getHeight()/2-343)+110+5, paint);
canvas.drawText("-10", (getWidth()/2-240)+220-5, (getHeight()/2-343)+270+5, paint);
canvas.drawText("-20", (getWidth()/2-240)+220-5, (getHeight()/2-343)+290+5, paint);
canvas.drawText("-30", (getWidth()/2-240)+220-5, (getHeight()/2-343)+310+5, paint);
canvas.drawText("-40", (getWidth()/2-240)+220-5, (getHeight()/2-343)+330+5, paint);
canvas.drawText("-50", (getWidth()/2-240)+220-5, (getHeight()/2-343)+350+5, paint);
canvas.drawText("-60", (getWidth()/2-240)+220-5, (getHeight()/2-343)+370+5, paint);

paint.setColor(Color.BLACK); //実験枠の描画
canvas.drawLine((getWidth()/2-240)+90, (getHeight()/2-343)+100, (getWidth()/2-240)+90, (get
Height()/2-343)+400, paint);
canvas.drawLine((getWidth()/2-240)+90, (getHeight()/2-343)+100, (getWidth()/2-240)+390, (ge
tHeight()/2-343)+100, paint);

paint.setColor(Color.BLUE); //表題の表示
paint.setTextSize(25.0f);
canvas.drawText("", (getWidth()/2-240)+60, (getHeight()/2-343)+65, paint);
canvas.drawText("【2次関数のグラフの広がり】", (getWidth()/2-240)+60+30-24, (getHeight()
/2-343)+65, paint);
//canvas.drawText("2", (getWidth()/2-240)+170, (getHeight()/2-343)+50, paint);

paint.setColor(Color.BLUE); //目標の提示
paint.setTextSize(19.0f);
canvas.drawText("a > 0 のとき、y = a x のグラフの広がりは、", (getWidth()/2-240)+30+18, (g
etHeight()/2-343)+440, paint);
canvas.drawText("2", (getWidth()/2-240)+255, (getHeight()/2-343)+440-5-5, paint);
canvas.drawText("a の値が大きくなるにつれて狭くなる。", (getWidth()/2-240)+30, (getHeight
()/2-343)+465, paint);
canvas.drawText("a < 0 のときは、a の値が小さくなるにつれ", (getWidth()/2-240)+30+18, (ge
tHeight()/2-343)+490, paint);
canvas.drawText("て狭くなる。これらのことを観察してみよう。", (getWidth()/2-240)+30, (get
Height()/2-343)+515, paint);

paint.setColor(Color.BLACK); //説明の表示
paint.setTextSize(19.0f);
canvas.drawText("※ 画面タッチで、y = x のグラフを描きます。", (getWidth()/2-240)+50-20, (g
etHeight()/2-343)+550-5, paint);
canvas.drawText("2", (getWidth()/2-240)+290-60+10, (getHeight()/2-343)+545-5, paint);
canvas.drawText("※ 画面をタッチするごとに順次グラフを描きます。", (getWidth()/2-240)+50-20,
(getHeight()/2-343)+575-5, paint);
canvas.drawText("※ 6回目と12回目のタッチで初期化されます。", (getWidth()/2-240)+50-20, (g
etHeight()/2-343)+600-5, paint);
canvas.drawText("※ 画面が暗くなったらステータスバーをタッチ！", (getWidth()/2-240)+50-2
0, (getHeight()/2-343)+625-5, paint);
paint.setColor(Color.BLUE);
paint.setTextSize(19.0f);
canvas.drawText("Copyright (C) K.Niwa 2014.08", (getWidth()/2-240)+110, (getHeight()/2-34
3)+650, paint); //作者・作成年月の表示

paint.setTextSize(12.0f);

if (flag==1) {
//y=x^2 のグラフの描画
paint.setColor(Color.BLUE);
canvas.drawText("y = x", (getWidth()/2-240)+300-20+50, (getHeight()/2-343)+230+
10, paint);

canvas.drawText("2", (getWidth()/2-240)+315+50, (getHeight()/2-343)+230-5+10, pai
nt);

```

```

fg=0;
for (x=-4.5;x<=4.6;x=x+0.1) {
    y=x*x;
    px=(int) (240+30*x);
    py=(int) (250-2*y);
    if (fg==0) {
        canvas.drawLine((int) (getWidth()/2-240)+px, (int) (getHeight()/2-
343)+py, (int) (getWidth()/2-240)+px, (int) (getHeight()/2-343)+py, paint);
    }
    else {
        canvas.drawLine((int) (getWidth()/2-240)+oldpx, (int) (getHeight()
/2-343)+oldpy, (int) (getWidth()/2-240)+px, (int) (getHeight()/2-343)+py, paint);
    }
    oldpx=px;oldpy=py;
    fg++;
}

}

else if (flag==2) {
//y=x^2 のグラフの描画
paint.setColor(Color. BLUE);
canvas.drawText("y = x ", (getWidth()/2-240)+330, (getHeight()/2-343)+240, pain
t);

canvas.drawText("2", (getWidth()/2-240)+365, (getHeight()/2-343)+235, paint);
fg=0;
for (x=-4.5;x<=4.6;x=x+0.1) {
    y=x*x;
    px=(int) (240+30*x);
    py=(int) (250-2*y);
    if (fg==0) {
        canvas.drawLine((int) (getWidth()/2-240)+px, (int) (getHeight()/2-
343)+py, (int) (getWidth()/2-240)+px, (int) (getHeight()/2-343)+py, paint);
    }
    else {
        canvas.drawLine((int) (getWidth()/2-240)+oldpx, (int) (getHeight()
/2-343)+oldpy, (int) (getWidth()/2-240)+px, (int) (getHeight()/2-343)+py, paint);
    }
    oldpx=px;oldpy=py;
    fg++;
}

//y=2x^2 のグラフの描画
paint.setColor(Color. RED);
canvas.drawText("y = 2 x ", (getWidth()/2-240)+270, (getHeight()/2-343)+220, pain
t);

canvas.drawText("2", (getWidth()/2-240)+313, (getHeight()/2-343)+215, paint);
fg=0;
for (x=-4.5;x<=4.6;x=x+0.1) {
    y=2*x*x;
    px=(int) (240+30*x);
    py=(int) (250-2*y);
    if (fg==0) {
        canvas.drawLine((int) (getWidth()/2-240)+px, (int) (getHeight()/2-
343)+py, (int) (getWidth()/2-240)+px, (int) (getHeight()/2-343)+py, paint);
    }
    else {
        canvas.drawLine((int) (getWidth()/2-240)+oldpx, (int) (getHeight()
/2-343)+oldpy, (int) (getWidth()/2-240)+px, (int) (getHeight()/2-343)+py, paint);
    }
    oldpx=px;oldpy=py;
    fg++;
}

}
}

```

```

else if (flag==3) {
    //y=x^2 のグラフの描画
    paint.setColor(Color.BLUE);
    canvas.drawText("y = x ", (getWidth()/2-240)+330, (getHeight()/2-343)+240, paint);

    canvas.drawText("2", (getWidth()/2-240)+365, (getHeight()/2-343)+235, paint);
    fg=0;
    for (x=-4.5;x<=4.6;x=x+0.1) {
        y=x*x;
        px=(int)(240+30*x);
        py=(int)(250-2*y);
        if (fg==0) {
            canvas.drawLine((int)(getWidth()/2-240)+px, (int)(getHeight()/2-343)+py, (int)(getWidth()/2-240)+px, (int)(getHeight()/2-343)+py, paint);
        }
        else {
            canvas.drawLine((int)(getWidth()/2-240)+oldpx, (int)(getHeight()/2-343)+oldpy, (int)(getWidth()/2-240)+px, (int)(getHeight()/2-343)+py, paint);
        }
        oldpx=px;oldpy=py;
        fg++;
    }

    //y=2x^2 のグラフの描画
    paint.setColor(Color.RED);
    canvas.drawText("y = 2 x ", (getWidth()/2-240)+270, (getHeight()/2-343)+220, paint);

    canvas.drawText("2", (getWidth()/2-240)+313, (getHeight()/2-343)+215, paint);
    fg=0;
    for (x=-4.5;x<=4.6;x=x+0.1) {
        y=2*x*x;
        px=(int)(240+30*x);
        py=(int)(250-2*y);
        if (fg==0) {
            canvas.drawLine((int)(getWidth()/2-240)+px, (int)(getHeight()/2-343)+py, (int)(getWidth()/2-240)+px, (int)(getHeight()/2-343)+py, paint);
        }
        else {
            canvas.drawLine((int)(getWidth()/2-240)+oldpx, (int)(getHeight()/2-343)+oldpy, (int)(getWidth()/2-240)+px, (int)(getHeight()/2-343)+py, paint);
        }
        oldpx=px;oldpy=py;
        fg++;
    }

    //y=3x^2 のグラフの描画
    paint.setColor(Color.BLACK);
    canvas.drawText("y = 3 x ", (getWidth()/2-240)+280, (getHeight()/2-343)+190, paint);

    canvas.drawText("2", (getWidth()/2-240)+323, (getHeight()/2-343)+185, paint);
    fg=0;
    for (x=-4.3;x<=4.4;x=x+0.1) {
        y=3*x*x;
        px=(int)(240+30*x);
        py=(int)(250-2*y);
        if (fg==0) {
            canvas.drawLine((int)(getWidth()/2-240)+px, (int)(getHeight()/2-343)+py, (int)(getWidth()/2-240)+px, (int)(getHeight()/2-343)+py, paint);
        }
        else {
            canvas.drawLine((int)(getWidth()/2-240)+oldpx, (int)(getHeight()/2-343)+oldpy, (int)(getWidth()/2-240)+px, (int)(getHeight()/2-343)+py, paint);
        }
        oldpx=px;oldpy=py;
        fg++;
    }
}
}

```

```

else if (flag==4) {
//y=x^2 のグラフの描画
paint.setColor(Color.BLUE);
canvas.drawText("y = x ", (getWidth()/2-240)+330, (getHeight()/2-343)+240, paint);
t);

canvas.drawText("2", (getWidth()/2-240)+365, (getHeight()/2-343)+235, paint);
fg=0;
for (x=-4.5;x<=4.6;x=x+0.1) {
y=x*x;
px=(int) (240+30*x);
py=(int) (250-2*y);
if (fg==0) {
canvas.drawLine((int) (getWidth()/2-240)+px, (int) (getHeight()/2-343)+py, (int) (getWidth()/2-240)+px, (int) (getHeight()/2-343)+py, paint);
}
else {
canvas.drawLine((int) (getWidth()/2-240)+oldpx, (int) (getHeight()/2-343)+oldpy, (int) (getWidth()/2-240)+px, (int) (getHeight()/2-343)+py, paint);
}
oldpx=px;oldpy=py;
fg++;
}

//y=2x^2 のグラフの描画
paint.setColor(Color.RED);
canvas.drawText("y = 2 x ", (getWidth()/2-240)+270, (getHeight()/2-343)+220, paint);
t);

canvas.drawText("2", (getWidth()/2-240)+313, (getHeight()/2-343)+215, paint);
fg=0;
for (x=-4.5;x<=4.6;x=x+0.1) {
y=2*x*x;
px=(int) (240+30*x);
py=(int) (250-2*y);
if (fg==0) {
canvas.drawLine((int) (getWidth()/2-240)+px, (int) (getHeight()/2-343)+py, (int) (getWidth()/2-240)+px, (int) (getHeight()/2-343)+py, paint);
}
else {
canvas.drawLine((int) (getWidth()/2-240)+oldpx, (int) (getHeight()/2-343)+oldpy, (int) (getWidth()/2-240)+px, (int) (getHeight()/2-343)+py, paint);
}
oldpx=px;oldpy=py;
fg++;
}

//y=3x^2 のグラフの描画
paint.setColor(Color.BLACK);
canvas.drawText("y = 3 x ", (getWidth()/2-240)+280, (getHeight()/2-343)+190, paint);
t);

canvas.drawText("2", (getWidth()/2-240)+323, (getHeight()/2-343)+185, paint);
fg=0;
for (x=-4.3;x<=4.4;x=x+0.1) {
y=3*x*x;
px=(int) (240+30*x);
py=(int) (250-2*y);
if (fg==0) {
canvas.drawLine((int) (getWidth()/2-240)+px, (int) (getHeight()/2-343)+py, (int) (getWidth()/2-240)+px, (int) (getHeight()/2-343)+py, paint);
}
else {
canvas.drawLine((int) (getWidth()/2-240)+oldpx, (int) (getHeight()/2-343)+oldpy, (int) (getWidth()/2-240)+px, (int) (getHeight()/2-343)+py, paint);
}
oldpx=px;oldpy=py;
fg++;
}

```

```

    }

    //y=4x^2 のグラフの描画
    paint.setColor(Color.MAGENTA);
    canvas.drawText("y = 4x ", (getWidth()/2-240)+280, (getHeight()/2-343)+160, paint);
t);

    canvas.drawText("2", (getWidth()/2-240)+323, (getHeight()/2-343)+155, paint);
    fg=0;
    for (x=-4.0;x<=4.1;x=x+0.1) {
        y=4*x*x;
        px=(int)(240+30*x);
        py=(int)(250-2*y);
        if (fg==0) {
            canvas.drawLine((int)(getWidth()/2-240)+px, (int)(getHeight()/2-
343)+py, (int)(getWidth()/2-240)+px, (int)(getHeight()/2-343)+py, paint);
        }
        else {
            canvas.drawLine((int)(getWidth()/2-240)+oldpx, (int)(getHeight()
/2-343)+oldpy, (int)(getWidth()/2-240)+px, (int)(getHeight()/2-343)+py, paint);
        }
        oldpx=px;oldpy=py;
        fg++;
    }
}

else if (flag==5) {
    //y=x^2 のグラフの描画
    paint.setColor(Color.BLUE);
    canvas.drawText("y = x ", (getWidth()/2-240)+330, (getHeight()/2-343)+240, paint);
t);

    canvas.drawText("2", (getWidth()/2-240)+365, (getHeight()/2-343)+235, paint);
    fg=0;
    for (x=-4.5;x<=4.6;x=x+0.1) {
        y=x*x;
        px=(int)(240+30*x);
        py=(int)(250-2*y);
        if (fg==0) {
            canvas.drawLine((int)(getWidth()/2-240)+px, (int)(getHeight()/2-
343)+py, (int)(getWidth()/2-240)+px, (int)(getHeight()/2-343)+py, paint);
        }
        else {
            canvas.drawLine((int)(getWidth()/2-240)+oldpx, (int)(getHeight()
/2-343)+oldpy, (int)(getWidth()/2-240)+px, (int)(getHeight()/2-343)+py, paint);
        }
        oldpx=px;oldpy=py;
        fg++;
    }
}

//y=2x^2 のグラフの描画
paint.setColor(Color.RED);
canvas.drawText("y = 2x ", (getWidth()/2-240)+270, (getHeight()/2-343)+220, paint);
t);

    canvas.drawText("2", (getWidth()/2-240)+313, (getHeight()/2-343)+215, paint);
    fg=0;
    for (x=-4.5;x<=4.6;x=x+0.1) {
        y=2*x*x;
        px=(int)(240+30*x);
        py=(int)(250-2*y);
        if (fg==0) {
            canvas.drawLine((int)(getWidth()/2-240)+px, (int)(getHeight()/2-
343)+py, (int)(getWidth()/2-240)+px, (int)(getHeight()/2-343)+py, paint);
        }
        else {
            canvas.drawLine((int)(getWidth()/2-240)+oldpx, (int)(getHeight()
/2-343)+oldpy, (int)(getWidth()/2-240)+px, (int)(getHeight()/2-343)+py, paint);
        }
    }
}

```

```

        oldpx=px;oldpy=py;
        fg++;
    }

    //y=3x^2 のグラフの描画
    paint.setColor(Color.BLACK);
    canvas.drawText(" y = 3 x ", (getWidth()/2-240)+280, (getHeight()/2-343)+190, paint);
t);

    canvas.drawText("2", (getWidth()/2-240)+323, (getHeight()/2-343)+185, paint);
    fg=0;
    for (x=-4.3;x<=4.4;x=x+0.1) {
        y=3*x*x;
        px=(int)(240+30*x);
        py=(int)(250-2*y);
        if (fg==0) {
            canvas.drawLine((int)(getWidth()/2-240)+px, (int)(getHeight()/2-
343)+py, (int)(getWidth()/2-240)+px, (int)(getHeight()/2-343)+py, paint);
        }
        else {
            canvas.drawLine((int)(getWidth()/2-240)+oldpx, (int)(getHeight()
/2-343)+oldpy, (int)(getWidth()/2-240)+px, (int)(getHeight()/2-343)+py, paint);
        }
        oldpx=px;oldpy=py;
        fg++;
    }

    //y=4x^2 のグラフの描画
    paint.setColor(Color.MAGENTA);
    canvas.drawText(" y = 4 x ", (getWidth()/2-240)+280, (getHeight()/2-343)+160, paint);
t);

    canvas.drawText("2", (getWidth()/2-240)+323, (getHeight()/2-343)+155, paint);
    fg=0;
    for (x=-4.0;x<=4.1;x=x+0.1) {
        y=4*x*x;
        px=(int)(240+30*x);
        py=(int)(250-2*y);
        if (fg==0) {
            canvas.drawLine((int)(getWidth()/2-240)+px, (int)(getHeight()/2-
343)+py, (int)(getWidth()/2-240)+px, (int)(getHeight()/2-343)+py, paint);
        }
        else {
            canvas.drawLine((int)(getWidth()/2-240)+oldpx, (int)(getHeight()
/2-343)+oldpy, (int)(getWidth()/2-240)+px, (int)(getHeight()/2-343)+py, paint);
        }
        oldpx=px;oldpy=py;
        fg++;
    }

    //y=5x^2 のグラフの描画
    paint.setColor(Color.CYAN);
    canvas.drawText(" y = 5 x ", (getWidth()/2-240)+290, (getHeight()/2-343)+130, paint);
t);

    canvas.drawText("2", (getWidth()/2-240)+333, (getHeight()/2-343)+125, paint);
    fg=0;
    for (x=-3.7;x<=3.8;x=x+0.1) {
        y=5*x*x;
        px=(int)(240+30*x);
        py=(int)(250-2*y);
        if (fg==0) {
            canvas.drawLine((int)(getWidth()/2-240)+px, (int)(getHeight()/2-
343)+py, (int)(getWidth()/2-240)+px, (int)(getHeight()/2-343)+py, paint);
        }
        else {
            canvas.drawLine((int)(getWidth()/2-240)+oldpx, (int)(getHeight()
/2-343)+oldpy, (int)(getWidth()/2-240)+px, (int)(getHeight()/2-343)+py, paint);
        }
        oldpx=px;oldpy=py;
        fg++;
    }

```



```

    }
}

else if (flag==6) {
}

if (flag==7) {
    //y=-x^2 のグラフの描画
    paint.setColor(Color. BLUE);
    canvas.drawText("y = -x ", (getWidth()/2-240)+280, (getHeight()/2-343)+280, paint);

    canvas.drawText("2", (getWidth()/2-240)+320, (getHeight()/2-343)+275, paint);
    fg=0;
    for (x=-4.5;x<=4.6;x=x+0.1) {
        y=-1*x*x;
        px=(int) (240+30*x);
        py=(int) (250-2*y);
        if (fg==0) {
            canvas.drawLine((int) (getWidth()/2-240)+px, (int) (getHeight()/2-343)+py, (int) (getWidth()/2-240)+px, (int) (getHeight()/2-343)+py, paint);
        }
        else {
            canvas.drawLine((int) (getWidth()/2-240)+oldpx, (int) (getHeight()/2-343)+oldpy, (int) (getWidth()/2-240)+px, (int) (getHeight()/2-343)+py, paint);
        }
        oldpx=px;oldpy=py;
        fg++;
    }
}

else if (flag==8) {
    //y=-x^2 のグラフの描画
    paint.setColor(Color. BLUE);
    canvas.drawText("y = -x ", (getWidth()/2-240)+280, (getHeight()/2-343)+280, paint);

    canvas.drawText("2", (getWidth()/2-240)+320, (getHeight()/2-343)+275, paint);
    fg=0;
    for (x=-4.5;x<=4.6;x=x+0.1) {
        y=-1*x*x;
        px=(int) (240+30*x);
        py=(int) (250-2*y);
        if (fg==0) {
            canvas.drawLine((int) (getWidth()/2-240)+px, (int) (getHeight()/2-343)+py, (int) (getWidth()/2-240)+px, (int) (getHeight()/2-343)+py, paint);
        }
        else {
            canvas.drawLine((int) (getWidth()/2-240)+oldpx, (int) (getHeight()/2-343)+oldpy, (int) (getWidth()/2-240)+px, (int) (getHeight()/2-343)+py, paint);
        }
        oldpx=px;oldpy=py;
        fg++;
    }
}

//y=-2x^2 のグラフの描画
paint.setColor(Color. RED);
canvas.drawText("y = -2x ", (getWidth()/2-240)+285, (getHeight()/2-343)+310, paint);

canvas.drawText("2", (getWidth()/2-240)+332, (getHeight()/2-343)+305, paint);
fg=0;
for (x=-4.5;x<=4.6;x=x+0.1) {
    y=-2*x*x;
    px=(int) (240+30*x);
    py=(int) (250-2*y);
}
}
}

```

```

        if (fg==0) {
            canvas.drawLine((int)(getWidth()/2-240)+px, (int)(getHeight()/2-
343)+py, (int)(getWidth()/2-240)+px, (int)(getHeight()/2-343)+py, paint);
        }
        else {
            canvas.drawLine((int)(getWidth()/2-240)+oldpx, (int)(getHeight()
/2-343)+oldpy, (int)(getWidth()/2-240)+px, (int)(getHeight()/2-343)+py, paint);
        }
        oldpx=px;oldpy=py;
        fg++;
    }
}

else if (flag==9) {
    //y=-x^2 のグラフの描画
    paint.setColor(Color.BLUE);
    canvas.drawText("y = -x ", (getWidth()/2-240)+280, (getHeight()/2-343)+280, pain
t);

    canvas.drawText("2", (getWidth()/2-240)+320, (getHeight()/2-343)+275, paint);
    fg=0;
    for (x=-4.5;x<=4.6;x=x+0.1) {
        y=-1*x*x;
        px=(int)(240+30*x);
        py=(int)(250-2*y);
        if (fg==0) {
            canvas.drawLine((int)(getWidth()/2-240)+px, (int)(getHeight()/2-
343)+py, (int)(getWidth()/2-240)+px, (int)(getHeight()/2-343)+py, paint);
        }
        else {
            canvas.drawLine((int)(getWidth()/2-240)+oldpx, (int)(getHeight()
/2-343)+oldpy, (int)(getWidth()/2-240)+px, (int)(getHeight()/2-343)+py, paint);
        }
        oldpx=px;oldpy=py;
        fg++;
    }

    //y=-2x^2 のグラフの描画
    paint.setColor(Color.RED);
    canvas.drawText("y = -2x ", (getWidth()/2-240)+285, (getHeight()/2-343)+310, pai
nt);

    canvas.drawText("2", (getWidth()/2-240)+332, (getHeight()/2-343)+305, paint);
    fg=0;
    for (x=-4.5;x<=4.6;x=x+0.1) {
        y=-2*x*x;
        px=(int)(240+30*x);
        py=(int)(250-2*y);
        if (fg==0) {
            canvas.drawLine((int)(getWidth()/2-240)+px, (int)(getHeight()/2-
343)+py, (int)(getWidth()/2-240)+px, (int)(getHeight()/2-343)+py, paint);
        }
        else {
            canvas.drawLine((int)(getWidth()/2-240)+oldpx, (int)(getHeight()
/2-343)+oldpy, (int)(getWidth()/2-240)+px, (int)(getHeight()/2-343)+py, paint);
        }
        oldpx=px;oldpy=py;
        fg++;
    }

    //y=-3x^2 のグラフの描画
    paint.setColor(Color.BLACK);
    canvas.drawText("y = -3x ", (getWidth()/2-240)+285, (getHeight()/2-343)+340, pai
nt);

    canvas.drawText("2", (getWidth()/2-240)+332, (getHeight()/2-343)+335, paint);
    fg=0;
    for (x=-4.3;x<=4.4;x=x+0.1) {

```

```

        y=-3*x*x;
        px=(int) (240+30*x);
        py=(int) (250-2*y);
        if (fg==0) {
            canvas.drawLine((int) (getWidth()/2-240)+px, (int) (getHeight()/2-
343)+py, (int) (getWidth()/2-240)+px, (int) (getHeight()/2-343)+py, paint);
        }
        else {
            canvas.drawLine((int) (getWidth()/2-240)+oldpx, (int) (getHeight()
/2-343)+oldpy, (int) (getWidth()/2-240)+px, (int) (getHeight()/2-343)+py, paint);
        }
        oldpx=px;oldpy=py;
        fg++;
    }
}

else if (flag==10) {
    //y=-x^2 のグラフの描画
    paint.setColor(Color. BLUE);
    canvas.drawText(" y = - x ", (getWidth()/2-240)+280, (getHeight()/2-343)+280, pain
t);

    canvas.drawText("2", (getWidth()/2-240)+320, (getHeight()/2-343)+275, paint);
    fg=0;
    for (x=-4.5;x<=4.6;x=x+0.1) {
        y=-1*x*x;
        px=(int) (240+30*x);
        py=(int) (250-2*y);
        if (fg==0) {
            canvas.drawLine((int) (getWidth()/2-240)+px, (int) (getHeight()/2-
343)+py, (int) (getWidth()/2-240)+px, (int) (getHeight()/2-343)+py, paint);
        }
        else {
            canvas.drawLine((int) (getWidth()/2-240)+oldpx, (int) (getHeight()
/2-343)+oldpy, (int) (getWidth()/2-240)+px, (int) (getHeight()/2-343)+py, paint);
        }
        oldpx=px;oldpy=py;
        fg++;
    }

    //y=-2x^2 のグラフの描画
    paint.setColor(Color. RED);
    canvas.drawText(" y = - 2 x ", (getWidth()/2-240)+285, (getHeight()/2-343)+310, pai
nt);

    canvas.drawText("2", (getWidth()/2-240)+332, (getHeight()/2-343)+305, paint);
    fg=0;
    for (x=-4.5;x<=4.6;x=x+0.1) {
        y=-2*x*x;
        px=(int) (240+30*x);
        py=(int) (250-2*y);
        if (fg==0) {
            canvas.drawLine((int) (getWidth()/2-240)+px, (int) (getHeight()/2-
343)+py, (int) (getWidth()/2-240)+px, (int) (getHeight()/2-343)+py, paint);
        }
        else {
            canvas.drawLine((int) (getWidth()/2-240)+oldpx, (int) (getHeight()
/2-343)+oldpy, (int) (getWidth()/2-240)+px, (int) (getHeight()/2-343)+py, paint);
        }
        oldpx=px;oldpy=py;
        fg++;
    }

    //y=-3x^2 のグラフの描画
    paint.setColor(Color. BLACK);
    canvas.drawText(" y = - 3 x ", (getWidth()/2-240)+285, (getHeight()/2-343)+340, pai
nt);
}
}

```

```

        canvas.drawText("2", (getWidth()/2-240)+332, (getHeight()/2-343)+335, paint);
        fg=0;
        for (x=-4.3;x<=4.4;x=x+0.1) {
            y=-3*x*x;
            px=(int) (240+30*x);
            py=(int) (250-2*y);
            if (fg==0) {
                canvas.drawLine((int) (getWidth()/2-240)+px, (int) (getHeight()/2-
343)+py, (int) (getWidth()/2-240)+px, (int) (getHeight()/2-343)+py, paint);
            }
            else {
                canvas.drawLine((int) (getWidth()/2-240)+oldpx, (int) (getHeight()
/2-343)+oldpy, (int) (getWidth()/2-240)+px, (int) (getHeight()/2-343)+py, paint);
            }
            oldpx=px;oldpy=py;
            fg++;
        }

        //y=-4x^2 のグラフの描画
        paint.setColor(Color.MAGENTA);
        canvas.drawText("y = -4x^2", (getWidth()/2-240)+285, (getHeight()/2-343)+370, pai
nt);

        canvas.drawText("2", (getWidth()/2-240)+332, (getHeight()/2-343)+365, paint);
        fg=0;
        for (x=-4.0;x<=4.1;x=x+0.1) {
            y=-4*x*x;
            px=(int) (240+30*x);
            py=(int) (250-2*y);
            if (fg==0) {
                canvas.drawLine((int) (getWidth()/2-240)+px, (int) (getHeight()/2-
343)+py, (int) (getWidth()/2-240)+px, (int) (getHeight()/2-343)+py, paint);
            }
            else {
                canvas.drawLine((int) (getWidth()/2-240)+oldpx, (int) (getHeight()
/2-343)+oldpy, (int) (getWidth()/2-240)+px, (int) (getHeight()/2-343)+py, paint);
            }
            oldpx=px;oldpy=py;
            fg++;
        }
    }

    else if (flag==11) {
        //y=-x^2 のグラフの描画
        paint.setColor(Color.BLUE);
        canvas.drawText("y = -x^2", (getWidth()/2-240)+280, (getHeight()/2-343)+280, pain
t);

        canvas.drawText("2", (getWidth()/2-240)+320, (getHeight()/2-343)+275, paint);
        fg=0;
        for (x=-4.5;x<=4.6;x=x+0.1) {
            y=-1*x*x;
            px=(int) (240+30*x);
            py=(int) (250-2*y);
            if (fg==0) {
                canvas.drawLine((int) (getWidth()/2-240)+px, (int) (getHeight()/2-
343)+py, (int) (getWidth()/2-240)+px, (int) (getHeight()/2-343)+py, paint);
            }
            else {
                canvas.drawLine((int) (getWidth()/2-240)+oldpx, (int) (getHeight()
/2-343)+oldpy, (int) (getWidth()/2-240)+px, (int) (getHeight()/2-343)+py, paint);
            }
            oldpx=px;oldpy=py;
            fg++;
        }

        //y=-2x^2 のグラフの描画
        paint.setColor(Color.RED);

```

```

    canvas.drawText("y = -2x ", (getWidth()/2-240)+285, (getHeight()/2-343)+310, paint);
    canvas.drawText("2", (getWidth()/2-240)+332, (getHeight()/2-343)+305, paint);
    fg=0;
    for (x=-4.5;x<=4.6;x=x+0.1) {
        y=-2*x*x;
        px=(int)(240+30*x);
        py=(int)(250-2*y);
        if (fg==0) {
            canvas.drawLine((int)(getWidth()/2-240)+px, (int)(getHeight()/2-
343)+py, (int)(getWidth()/2-240)+px, (int)(getHeight()/2-343)+py, paint);
        }
        else {
            canvas.drawLine((int)(getWidth()/2-240)+oldpx, (int)(getHeight()/2-343)+oldpy, (int)(getWidth()/2-240)+px, (int)(getHeight()/2-343)+py, paint);
        }
        oldpx=px;oldpy=py;
        fg++;
    }

    //y=-3x^2 のグラフの描画
    paint.setColor(Color.BLACK);
    canvas.drawText("y = -3x ", (getWidth()/2-240)+285, (getHeight()/2-343)+340, paint);
    canvas.drawText("2", (getWidth()/2-240)+332, (getHeight()/2-343)+335, paint);
    fg=0;
    for (x=-4.3;x<=4.4;x=x+0.1) {
        y=-3*x*x;
        px=(int)(240+30*x);
        py=(int)(250-2*y);
        if (fg==0) {
            canvas.drawLine((int)(getWidth()/2-240)+px, (int)(getHeight()/2-343)+py, (int)(getWidth()/2-240)+px, (int)(getHeight()/2-343)+py, paint);
        }
        else {
            canvas.drawLine((int)(getWidth()/2-240)+oldpx, (int)(getHeight()/2-343)+oldpy, (int)(getWidth()/2-240)+px, (int)(getHeight()/2-343)+py, paint);
        }
        oldpx=px;oldpy=py;
        fg++;
    }

    //y=-4x^2 のグラフの描画
    paint.setColor(Color.MAGENTA);
    canvas.drawText("y = -4x ", (getWidth()/2-240)+285, (getHeight()/2-343)+370, paint);
    canvas.drawText("2", (getWidth()/2-240)+332, (getHeight()/2-343)+365, paint);
    fg=0;
    for (x=-4.0;x<=4.1;x=x+0.1) {
        y=-4*x*x;
        px=(int)(240+30*x);
        py=(int)(250-2*y);
        if (fg==0) {
            canvas.drawLine((int)(getWidth()/2-240)+px, (int)(getHeight()/2-343)+py, (int)(getWidth()/2-240)+px, (int)(getHeight()/2-343)+py, paint);
        }
        else {
            canvas.drawLine((int)(getWidth()/2-240)+oldpx, (int)(getHeight()/2-343)+oldpy, (int)(getWidth()/2-240)+px, (int)(getHeight()/2-343)+py, paint);
        }
        oldpx=px;oldpy=py;
        fg++;
    }

    //y=-5x^2 のグラフの描画
    paint.setColor(Color.CYAN);
    canvas.drawText("y = -5x ", (getWidth()/2-240)+285, (getHeight()/2-343)+390, paint);

```

```

        canvas.drawText("2", (getWidth()/2-240)+332, (getHeight()/2-343)+385, paint);
        fg=0;
        for (x=-3.7;x<=3.8;x=x+0.1) {
            y=-5*x*x;
            px=(int) (240+30*x);
            py=(int) (250-2*y);
            if (fg==0) {
                canvas.drawLine((int) (getWidth()/2-240)+px, (int) (getHeight()/2-
343)+py, (int) (getWidth()/2-240)+px, (int) (getHeight()/2-343)+py, paint);
            }
            else {
                canvas.drawLine((int) (getWidth()/2-240)+oldpx, (int) (getHeight()
/2-343)+oldpy, (int) (getWidth()/2-240)+px, (int) (getHeight()/2-343)+py, paint);
            }
            oldpx=px;oldpy=py;
            fg++;
        }
    }

    else if (flag==0) {
        //a=0; //初期化する
        //b=0; //初期化する
        //invalidate(); //再描画する (clear & goto onDraw) そして、この行へ戻ってくる。
    }

} //protected void onDraw(Canvas canvas)

```

//画面にタッチしたときのイベント処理-----

```

@Override
public boolean onTouchEvent(MotionEvent event) {

    flag=flag+1; //flagに1を加える
    flag=flag % 12; //flagに1、2、・・・、10、11、0を代入する

    invalidate(); //再描画する (clear & goto onDraw)
    return false;
}
}

```

[2] main.xml

```

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

<LinearLayout
    xmlns:android="http://schemas.android.com/apk/res/android"
    android:orientation="vertical"
    android:layout_width="match_parent"
    android:layout_height="match_parent"
    android:weightSum="1" >

    <jp.seitoku.twokansu3.Twokansu3
        android:layout_height="match_parent"
        android:layout_width="match_parent"
        android:id="@+id/myview1">
    </jp.seitoku.twokansu3.Twokansu3>

</LinearLayout>

```

[3] Twokansu3Activity.java

```

package jp.seitoku.twokansu3;

```

```
import android.os.Bundle;
import android.app.Activity;
import android.view.Menu;

public class Twokansu3Activity extends Activity {

    @Override
    public void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);
        setContentView(R.layout.main);
    }

    @Override
    public boolean onCreateOptionsMenu(Menu menu) {
        getMenuInflater().inflate(R.menu.main, menu);
        return true;
    }
}
```