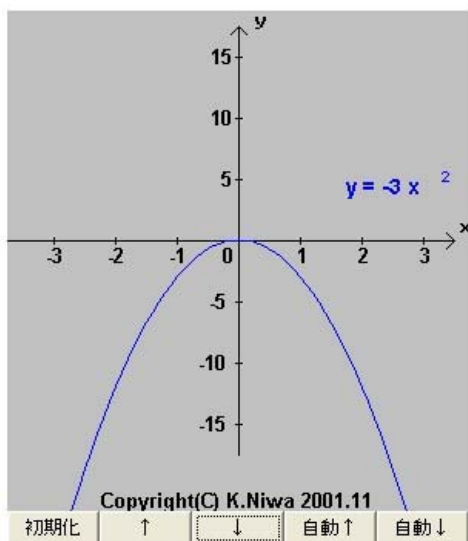
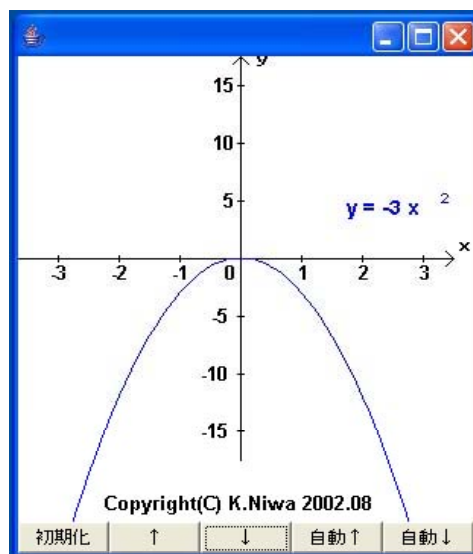


## 【2次関数のグラフの向きと広がり】



【J a v a アプレット】



【J a v a アプリケーション】

### 1. はじめに

$y = a x^2$  のグラフについて、 $x^2$  の係数  $a$  が正のときと負のときでは、グラフにどのような違いがあるでしょうか。

$y = a x^2$  のグラフについて、 $x^2$  の係数  $a$  が正のとき、係数  $a$  の値が大きくなるにつれて、グラフの広がりはどうなるでしょうか。

$y = a x^2$  のグラフについて、 $x^2$  の係数  $a$  が負のとき、係数  $a$  の値が小さくなるにつれて、グラフの広がりはどうなるでしょうか。

シミュレーションソフト「2次関数のグラフの向きと広がり」を使って、 $y = a x^2$  のグラフを観察してみてください。

### 2. J a v a アプレット

#### (1) J a v a プログラムリスト

```

//////////////////////////////////////
//
//                               「2次関数のグラフの向きと広がり」
//                               y = ax^2
//                               Copyright (C) K.Niwa 2001.11.17
//                               (J a v a アプレット)
//
//////////////////////////////////////

```

```

import java.applet.Applet;
import java.awt.*;
import java.awt.event.*;
import java.lang.Math;

```

```

public class Kansuu3 extends Applet{
    Button[] myBtn;
    Panel myPanel;
    int flag=0;
    double x,y;
    int px,py,oldpx,oldpy;
    int N;

```

```

//ボタン型として宣言する
//パネル型として宣言する
//変数宣言と初期化
//グラフの座標
//グラフの座標
//ゴマの個数

```

```

int count,i; //ループカウンター
int fg; //グラフ描写に使用したフラッグ
int a=1; //x^2の前の係数
int t; //時間かせぎ
int p; //上に凸か下に凸かのフラッグに使用

//public void init() init メソッド (void は値を返さないメソッドの意) -----
public void init() {
    setBackground(Color.lightGray);
    myBtn=new Button[5]; //ボタンの実体化
    myBtn[0]=new Button("初期化");
    myBtn[1]=new Button("↑");
    myBtn[2]=new Button("↓");
    myBtn[3]=new Button("自動↑");
    myBtn[4]=new Button("自動↓");

    myPanel=new Panel();
    myPanel.setLayout(new GridLayout(1,5));
    for (count=0;count<=4;count++)
    {
        myPanel.add(myBtn[count]);
    }
    setLayout(new BorderLayout());
    add("South",myPanel);

//初期化ボタンを押したときのイベント処理
    myBtn[0].addActionListener(new ActionListener() {
        public void actionPerformed(ActionEvent e) {
            flag=0;
            a=1;
            repaint();
        }
    });

//↑ボタンを押したときのイベント処理
    myBtn[1].addActionListener(new ActionListener() {
        public void actionPerformed(ActionEvent e) {
            flag=0;

            if (a<99)
            {
                if (a==1)
                {
                    a=0;
                }
                if (a!=1)
                {
                    a=a+1;
                }
            }
            repaint();
        }
    });

//↓ボタンを押したときのイベント処理
    myBtn[2].addActionListener(new ActionListener() {
        public void actionPerformed(ActionEvent e) {
            flag=0;

            if (a>-99)
            {
                if (a==1)
                {
                    a=0;
                }
            }
        }
    });

```

```

        }
        if (a!=1)
        {
            a=a-1;
        }
        repaint();
    }
});

//自動↑ボタンを押したときのイベント処理
myBtn[3].addActionListener(new ActionListener() {
    public void actionPerformed(ActionEvent e) {
        flag=1;
        //a=1;
        repaint();
    }
});

//自動↓ボタンを押したときのイベント処理
myBtn[4].addActionListener(new ActionListener() {
    public void actionPerformed(ActionEvent e) {
        flag=2;
        //a=1;
        repaint();
    }
});

} //public void init()

//public void paint(Graphics g) //paint メソッド -----
public void paint(Graphics g) { //paint メソッド

    if (flag==0) {
        g.clearRect(0,0,300,350); //全体のクリア
        Font f0=new Font((g.getFont()).getName(),Font.BOLD,13);
        g.setFont(f0);

        //座標軸
        g.drawLine(150,10,150,290);
        g.drawLine(150,10,150-5,10+5);
        g.drawLine(150,10,150+5,10+5); //y 軸
        g.drawString("y",155+3,10);
        for (i=0;i<=120;i=i+40) {
            g.drawLine(150-2,150-i,150+2,150-i); //y 軸メモリ
        }
        for (i=0;i<=120;i=i+40) {
            g.drawLine(150-2,150+i,150+2,150+i); //y 軸メモリ
        }
        g.drawString("5",150-15-3+3,110+5); //y 軸メモリ値
        g.drawString("10",150-15-3,70+5);
        g.drawString("15",150-15-3,30+5);
        g.drawString("-5",150-20-6+3,190+5);
        g.drawString("-10",150-20-6,230+5);
        g.drawString("-15",150-20-6,270+5); //y 軸メモリ値

        g.drawLine(0,150,290,150);
        g.drawLine(290,150,290-5,150-5);
        g.drawLine(290,150,290-5,150+5); //x 軸
        g.drawString("x",290,145);
        for (i=0;i<=120;i=i+40) {
            g.drawLine(150+i,150-2,150+i,150+2); //x 軸メモリ
        }
    }
}

```

```

for (i=0;i<=120;i=i+40) {
    g.drawLine(150-i,150-2,150-i,150+2);/          /x 軸メモリ
}
g.drawString("1",190-2,165);/          /x 軸メモリ値
g.drawString("2",230-2,165);
g.drawString("3",270-2,165);
g.drawString("0",150-10,165);
g.drawString("-1",110-5,165);
g.drawString("-2",70-5,165);
g.drawString("-3",30-5,165);          //x 軸メモリ値

//グラフ描写
//y=ax^2
g.setColor(Color.blue);
Font f4=new Font((g.getFont()).getName(),Font.BOLD,14);
g.setFont(f4);
g.drawString("y = "+a+" x",220,120);

Font f5=new Font((g.getFont()).getName(),Font.BOLD,10);
g.setFont(f5);
g.drawString("2",250+2+30,110+2);

for (x=-4;x<=4.1;x=x+0.1) {
    y=a*x*x;
    px=(int)(150+40*x);
    py=(int)(150-8*y);
    if (fg==0) {
        g.drawLine(px,py,px,py);
    }
    else {
        g.drawLine(oldpx,oldpy,px,py);
    }
    oldpx=px;oldpy=py;
    fg++;
}
fg=0;

g.setColor(Color.black);

Font f6=new Font((g.getFont()).getName(),Font.BOLD,13);
g.setFont(f6);
g.drawString("Copyright (C) K.Niwa 2001.11",60,325);          //作者名
} //if (flag==0)

else if (flag==1) { //下に凸
    for (a=1;a<=20;a++) {
        g.clearRect(0,0,300,350); //全体のクリア
        Font f0=new Font((g.getFont()).getName(),Font.BOLD,13);
        g.setFont(f0);

//座標軸
        g.drawLine(150,10,150,290);
        g.drawLine(150,10,150-5,10+5);
        g.drawLine(150,10,150+5,10+5);          //y 軸
        g.drawString("y ",155+3,10);
        for (i=0;i<=120;i=i+40) {
            g.drawLine(150-2,150-i,150+2,150-i); //y 軸メモリ
        }
        for (i=0;i<=120;i=i+40) {
            g.drawLine(150-2,150+i,150+2,150+i); //y 軸メモリ
        }
        g.drawString(" 5",150-15-3+3,110+5);          //y 軸メモリ値
    }
}

```

```

g.drawString("10",150-15-3,70+5);
g.drawString("15",150-15-3,30+5);
g.drawString(" -5",150-20-6+3,190+5);
g.drawString("-10",150-20-6,230+5);
g.drawString("-15",150-20-6,270+5);           // y 軸メモリ値

g.drawLine(0,150,290,150);
g.drawLine(290,150,290-5,150-5);
g.drawLine(290,150,290-5,150+5);           // x 軸
g.drawString(" x ",290,145);
for (i=0;i<=120;i=i+40) {
    g.drawLine(150+i,150-2,150+i,150+2); // x 軸メモリ
}
for (i=0;i<=120;i=i+40) {
    g.drawLine(150-i,150-2,150-i,150+2); // x 軸メモリ
}
g.drawString("1",190-2,165);           // x 軸メモリ値
g.drawString("2",230-2,165);
g.drawString("3",270-2,165);
g.drawString("0",150-10,165);
g.drawString("-1",110-5,165);
g.drawString("-2",70-5,165);
g.drawString("-3",30-5,165);           // x 軸メモリ値

```

//グラフ描写

```

//y=ax^2
g.setColor(Color.blue);
Font f4=new Font((g.getFont()).getName(),Font.BOLD,14);
g.setFont(f4);
g.drawString("y = "+a+" x",220,120);

Font f5=new Font((g.getFont()).getName(),Font.BOLD,10);
g.setFont(f5);
g.drawString("2",250+2+30,110+2);

for (x=-4;x<=4.1;x=x+0.1) {
    y=a*x*x;
    px=(int) (150+40*x);
    py=(int) (150-8*y);
    if (fg==0) {
        g.drawLine(px,py,px,py);
    }
    else {
        g.drawLine(oldpx,oldpy,px,py);
    }
    oldpx=px;oldpy=py;
    fg++;
}
fg=0;

g.setColor(Color.black);

Font f6=new Font((g.getFont()).getName(),Font.BOLD,13);
g.setFont(f6);
g.drawString("Copyright(C) K.Niwa 2001.11",60,325); //作者名

for (t=1;t<50000000;t++) {}           //時間稼ぎ

} //for (a=1;a<=20;a++)
a=a-1;

} //else if (flag==1)

```

```

else if (flag==2) { //上に凸
    for (a=-1;a>=-20;a--) {
        g.clearRect(0,0,300,350); //全体のクリア

        Font f0=new Font((g.getFont()).getName(),Font.BOLD,13);
        g.setFont(f0);

//座標軸

        g.drawLine(150,10,150,290);
        g.drawLine(150,10,150-5,10+5);
        g.drawLine(150,10,150+5,10+5); //y軸
        g.drawString("y",155+3,10);
        for (i=0;i<=120;i=i+40) {
            g.drawLine(150-2,150-i,150+2,150-i); //y軸メモリ
        }
        for (i=0;i<=120;i=i+40) {
            g.drawLine(150-2,150+i,150+2,150+i); //y軸メモリ
        }
        g.drawString("5",150-15-3+3,110+5); //y軸メモリ値
        g.drawString("10",150-15-3,70+5);
        g.drawString("15",150-15-3,30+5);
        g.drawString("-5",150-20-6+3,190+5);
        g.drawString("-10",150-20-6,230+5);
        g.drawString("-15",150-20-6,270+5); //y軸メモリ値

        g.drawLine(0,150,290,150);
        g.drawLine(290,150,290-5,150-5);
        g.drawLine(290,150,290-5,150+5); //x軸
        g.drawString("x",290,145);
        for (i=0;i<=120;i=i+40) {
            g.drawLine(150+i,150-2,150+i,150+2); //x軸メモリ
        }
        for (i=0;i<=120;i=i+40) {
            g.drawLine(150-i,150-2,150-i,150+2); //x軸メモリ
        }
        g.drawString("1",190-2,165); //x軸メモリ値
        g.drawString("2",230-2,165);
        g.drawString("3",270-2,165);
        g.drawString("0",150-10,165);
        g.drawString("-1",110-5,165);
        g.drawString("-2",70-5,165);
        g.drawString("-3",30-5,165); //x軸メモリ値

//グラフ描写
//y=ax^2
        g.setColor(Color.blue);
        Font f4=new Font((g.getFont()).getName(),Font.BOLD,14);
        g.setFont(f4);
        g.drawString("y = "+a+" x",220,120);

        Font f5=new Font((g.getFont()).getName(),Font.BOLD,10);
        g.setFont(f5);
        g.drawString("2",250+2+30,110+2);

        for (x=-4;x<=4.1;x=x+0.1) {
            y=a*x*x;
            px=(int)(150+40*x);
            py=(int)(150-8*y);
            if (fg==0) {
                g.drawLine(px,py,px,py);
            }
        }
    }
}

```

```

        else {
            g.drawLine (oldpx,oldpy,px,py);
        }
        oldpx=px;oldpy=py;
        fg++;
    }
    fg=0;

    g.setColor (Color.black);

    Font f6=new Font ((g.getFont ()).getName (),Font.BOLD,13);
    g.setFont (f6);
    g.drawString ("Copyright (C) K.Niwa 2001.11",60,325); //作者名

    for (t=1;t<50000000;t++) {} //時間稼ぎ

} //for (for (a=-1;a>=-20;a--)
a=a+1;

} //else if (flag==2)

} //public void paint (Graphics g) //paint メソッド
} //public class Kansuu extends Applet

```

## (2) HTML リスト

```

<HTML>
<HEAD>
< !-----
                2次関数のグラフの向きと広がり
                y = ax^2
                Copyright (C) K.Niwa 2001.11.17
----->
</HEAD>
<BODY>
<CENTER>
<B> 2次関数の平行移動 3 </B>
<BR><BR>
<APPLET CODE="Kansuu3.class" WIDTH="300" HEIGHT="350"></APPLET>
<BR><BR>
<B>※ x^2 の前の係数が正のときと負のときとでは、グラフにどのような違いがありますか。 </B>
<BR>
<B>※ x^2 の前の係数が正のとき、係数が大きくなるにつれて、グラフの広がりはどうなりますか。
</B>
<BR>
<B>※ x^2 の前の係数が負のとき、係数が小さくなるにつれて、グラフの広がりはどうなりますか。
</B>
</CENTER>
</BODY>
</HTML>

```





```

//↑ ボタンを押したときのイベント処理
myBtn[1].addActionListener(new ActionListener() {
    public void actionPerformed(ActionEvent e) {
        flag=0;

        if (a<99) {
            if (a==1)
            {
                a=0;
            }
            if (a!=1)
            {
                a=a+1;
            }
        }
        repaint();
    }
});

```

```

//↓ ボタンを押したときのイベント処理
myBtn[2].addActionListener(new ActionListener() {
    public void actionPerformed(ActionEvent e) {
        flag=0;

        if (a>-99) {
            if (a==1) {
                a=0;
            }
            if (a!=1) {
                a=a-1;
            }
        }
        repaint();
    }
});

```

```

//自動↑ ボタンを押したときのイベント処理
myBtn[3].addActionListener(new ActionListener() {
    public void actionPerformed(ActionEvent e) {
        flag=1;
        //a=1;
        repaint();
    }
});

```

```

//自動↓ ボタンを押したときのイベント処理
myBtn[4].addActionListener(new ActionListener() {
    public void actionPerformed(ActionEvent e) {
        flag=2;
        //a=1;
        repaint();
    }
});

```

```

} //public FKansuu3()

```

```

//public void paint(Graphics g) //paint メソッド -----
public void paint(Graphics g) { //paint メソッド

    if (flag==0) {
        g.clearRect(0,0,300+10,360+20); //全体のクリア
        Font f0=new Font((g.getFont()).getName(),Font.BOLD,13);
        g.setFont(f0);
    }
}

```

```

//座標軸
g.drawLine(150,10+20,150,290+20);
g.drawLine(150,10+20,150-5,10+5+20);
g.drawLine(150,10+20,150+5,10+5+20); // y 軸
g.drawString(" y ",155+3,10+20+5);
for (i=0;i<=120;i=i+40) {
    g.drawLine(150-2,150-i+20,150+2,150-i+20); // y 軸メモリ
}
for (i=0;i<=120;i=i+40)
{
    g.drawLine(150-2,150+i+20,150+2,150+i+20); // y 軸メモリ
}
g.drawString(" 5",150-15-3+3,110+5+20); // y 軸メモリ値
g.drawString("10",150-15-3,70+5+20);
g.drawString("15",150-15-3,30+5+20);
g.drawString(" -5",150-20-6+3,190+5+20);
g.drawString(" -10",150-20-6,230+5+20);
g.drawString(" -15",150-20-6,270+5+20); // y 軸メモリ値

g.drawLine(0,150+20,290,150+20);
g.drawLine(290,150+20,290-5,150-5+20);
g.drawLine(290,150+20,290-5,150+5+20); // x 軸
g.drawString(" x ",290,145+20);
for (i=0;i<=120;i=i+40) {
    g.drawLine(150+i,150-2+20,150+i,150+2+20); // x 軸メモリ
}
for (i=0;i<=120;i=i+40) {
    g.drawLine(150-i,150-2+20,150-i,150+2+20); // x 軸メモリ
}
g.drawString("1",190-2,165+20); // x 軸メモリ値
g.drawString("2",230-2,165+20);
g.drawString("3",270-2,165+20);
g.drawString("0",150-10,165+20);
g.drawString("-1",110-5,165+20);
g.drawString("-2",70-5,165+20);
g.drawString("-3",30-5,165+20); // x 軸メモリ値

//グラフ描写
//y=ax^2
g.setColor(Color.blue);

Font f4=new Font((g.getFont()).getName(),Font.BOLD,14);
g.setFont(f4);
g.drawString("y = "+a+" x",220,120+20);

Font f5=new Font((g.getFont()).getName(),Font.BOLD,10);
g.setFont(f5);
g.drawString("2",250+2+30,110+2+20);

for (x=-4;x<=4.1;x=x+0.1) {
    y=a*x*x;
    px=(int) (150+40*x);
    py=(int) (150-8*y+20);
    if (fg==0) {
        g.drawLine(px,py,px,py);
    }
    else {
        g.drawLine(oldpx,oldpy,px,py);
    }
    oldpx=px;oldpy=py;
    fg++;
}
fg=0;

```

```

g.setColor(Color.black);

Font f6=new Font((g.getFont()).getName(),Font.BOLD,13);
g.setFont(f6);
g.drawString("Copyright (C) K.Niwa 2002.08",60,325+20);      //作者名
} //if (flag==0)
else if (flag==1) {      //下に凸
    for (a=1;a<=20;a++) {

        g.clearRect(0,0,300+10,360+20);      //全体のクリア

        Font f0=new Font((g.getFont()).getName(),Font.BOLD,13);
        g.setFont(f0);

        //座標軸
        g.drawLine(150,10+20,150,290+20);
        g.drawLine(150,10+20,150-5,10+5+20);
        g.drawLine(150,10+20,150+5,10+5+20); // y 軸
        g.drawString("y",155+3,10+20+5);
        for (i=0;i<=120;i=i+40) {
            g.drawLine(150-2,150-i+20,150+2,150-i+20); // y 軸メモリ
        }
        for (i=0;i<=120;i=i+40) {
            g.drawLine(150-2,150+i+20,150+2,150+i+20); // y 軸メモリ
        }
        g.drawString("5",150-15-3+3,110+5+20);      // y 軸メモリ値
        g.drawString("10",150-15-3,70+5+20);
        g.drawString("15",150-15-3,30+5+20);
        g.drawString("-5",150-20-6+3,190+5+20);
        g.drawString("-10",150-20-6,230+5+20);
        g.drawString("-15",150-20-6,270+5+20);      // y 軸メモリ値

        g.drawLine(0,150+20,290,150+20);
        g.drawLine(290,150+20,290-5,150-5+20);
        g.drawLine(290,150+20,290-5,150+5+20);      // x 軸
        g.drawString("x",290,145+20);
        for (i=0;i<=120;i=i+40) {
            g.drawLine(150+i,150-2+20,150+i,150+2+20); // x 軸メモリ
        }
        for (i=0;i<=120;i=i+40) {
            g.drawLine(150-i,150-2+20,150-i,150+2+20); // x 軸メモリ
        }
        g.drawString("1",190-2,165+20);      // x 軸メモリ値
        g.drawString("2",230-2,165+20);
        g.drawString("3",270-2,165+20);
        g.drawString("0",150-10,165+20);
        g.drawString("-1",110-5,165+20);
        g.drawString("-2",70-5,165+20);
        g.drawString("-3",30-5,165+20);      // x 軸メモリ値

        //グラフ描写
        //y=ax^2

        g.setColor(Color.blue);

        Font f4=new Font((g.getFont()).getName(),Font.BOLD,14);
        g.setFont(f4);
        g.drawString("y = "+a+" x",220,120+20);

        Font f5=new Font((g.getFont()).getName(),Font.BOLD,10);
        g.setFont(f5);
        g.drawString("2",250+2+30,110+2+20);

```

```

for (x=-4;x<=4.1;x=x+0.1) {
    y=a*x*x;
    px=(int) (150+40*x);
    py=(int) (150-8*y+20);
    if (fg==0) {
        g.drawLine(px,py,px,py);
    }

    else{
        g.drawLine(oldpx,oldpy,px,py);
    }
    oldpx=px;oldpy=py;
    fg++;
}
fg=0;

g.setColor(Color.black);

Font f6=new Font((g.getFont()).getName(),Font.BOLD,13);
g.setFont(f6);
g.drawString("Copyright(C) K.Niwa 2002.08",60,325+20);//作者名

for (t=1;t<50000000;t++) {} //時間稼ぎ

} //for (a=1;a<=20;a++)
a=a-1;

} //else if (flag==1)

else if (flag==2) { //上に凸
    for (a=-1;a>=-20;a--) {
        g.clearRect(0,0,300+10,360+20); //全体のクリア
        Font f0=new Font((g.getFont()).getName(),Font.BOLD,13);
        g.setFont(f0);

        //座標軸
        g.drawLine(150,10+20,150,290+20);
        g.drawLine(150,10+20,150-5,10+5+20);
        g.drawLine(150,10+20,150+5,10+5+20); // y 軸
        g.drawString(" y ",155+3,10+20+5);
        for (i=0;i<=120;i=i+40) {
            g.drawLine(150-2,150-i+20,150+2,150-i+20); // y 軸メモリ
        }
        for (i=0;i<=120;i=i+40) {
            g.drawLine(150-2,150+i+20,150+2,150+i+20); // y 軸メモリ
        }
        g.drawString(" 5",150-15-3+3,110+5+20); // y 軸メモリ値
        g.drawString("10",150-15-3,70+5+20);
        g.drawString("15",150-15-3,30+5+20);
        g.drawString(" -5",150-20-6+3,190+5+20);
        g.drawString("-10",150-20-6,230+5+20);
        g.drawString("-15",150-20-6,270+5+20); // y 軸メモリ値

        g.drawLine(0,150+20,290,150+20);
        g.drawLine(290,150+20,290-5,150-5+20);
        g.drawLine(290,150+20,290-5,150+5+20); // x 軸
        g.drawString(" x ",290,145+20);
        for (i=0;i<=120;i=i+40) {
            g.drawLine(150+i,150-2+20,150+i,150+2+20); // x 軸メモリ
        }
        for (i=0;i<=120;i=i+40) {
            g.drawLine(150-i,150-2+20,150-i,150+2+20); // x 軸メモリ
        }
        g.drawString("1",190-2,165+20); // x 軸メモリ値
    }
}

```

```

        g.drawString("2",230-2,165+20);
        g.drawString("3",270-2,165+20);
        g.drawString("0",150-10,165+20);
        g.drawString("-1",110-5,165+20);
        g.drawString("-2",70-5,165+20);
        g.drawString("-3",30-5,165+20); // x 軸メモリ値

//グラフ描写
//y=ax^2

        g.setColor(Color.blue);

        Font f4=new Font((g.getFont()).getName(),Font.BOLD,14);
        g.setFont(f4);
        g.drawString("y = "+a+" x",220,120+20);

        Font f5=new Font((g.getFont()).getName(),Font.BOLD,10);
        g.setFont(f5);
        g.drawString("2",250+2+30,110+2+20);

        for (x=-4;x<=4.1;x=x+0.1) {
            y=a*x*x;
            px=(int) (150+40*x);
            py=(int) (150-8*y+20);
            if (fg==0)
            {
                g.drawLine(px,py,px,py);
            }
            else
            {
                g.drawLine(oldpx,oldpy,px,py);
            }
            oldpx=px;oldpy=py;
            fg++;
        }
        fg=0;

        g.setColor(Color.black);

        Font f6=new Font((g.getFont()).getName(),Font.BOLD,13);
        g.setFont(f6);
        g.drawString("Copyright (C) K.Niwa 2002.08",60,325+20); //作者名

        for (t=1;t<50000000;t++) {} //時間稼ぎ

    } //for (for (a=-1;a>=-20;a--)
      a=a+1;

    } //else if (flag==2)

} //public void paint(Graphics g) //paint メソッド

/***** public static void main メソッド *****/
public static void main(String[] args) {
    Frame w=new FKansuu3();
    w.show();
} //public static void main(String[] args)

} //public class FKansuu3 extends Frame

```