

```
package turm_von_hanoi ;

import java.awt.*;
import java.awt.event.*;
import java.applet.*;

public class hanoi extends Applet implements Runnable {

    public boolean lauf;
    Label tmp=new Label();
    int[] xPosBalken ,yPosBalken ,belegung ;
    int scheinenzahl ,pos;
    String loes=new String();
    Thread anima;
    Image image;
    boolean isStandalone = false;
    Button button1 = new Button();
    Button button2 = new Button();
    Button button3 = new Button();
    Button button4 = new Button();
    Button button5 = new Button();
    Panel panell = new Panel();
    Choice choice1 = new Choice();
    Label labell = new Label();

    public String getParameter (String key, String def) {
        return isStandalone ? System.getProperty (key, def) :
            (getParameter (key) != null ? getParameter (key) : def);
    }

    public hanoi(){

    }

    public void run()
    {
        Thread thisThread=Thread.currentThread ();
        for (;thisThread==anima ;)
        {
            if(pos<(int)Math.pow(2,scheinenzahl )-1 & lauf)
            {
                demo (true);
                tmp.setText ("+++"+pos+"+++");
                try{anima.sleep (500);}
                catch (Exception E){}
            }
        }
    }

    //Initialisierung des Applets
    public void init() {
        try {
            jbInit();
            minit();
        }
        catch (Exception e) {
            e.printStackTrace ();
        }
    }

    //Initialisierung der Komponenten
    private void jbInit() throws Exception {
        button1.setLabel ("Reset");
        button1.setBounds (new Rectangle (5, 5, 90, 27));
        button1.addActionListener (new java.awt.event.ActionListener () {
            public void actionPerformed (ActionEvent e) {
                button1_actionPerformed (e);
            }
        });
        button2.setLabel ("Animation");
        button2.setBounds (new Rectangle (5, 35, 90, 27));
        button2.addActionListener (new java.awt.event.ActionListener () {
            public void actionPerformed (ActionEvent e) {
                button2_actionPerformed (e);
            }
        });
        button3.setLabel ("Back");
        button3.setBounds (new Rectangle (5, 65, 90, 27));
```

```

        button3.addActionListener (new java.awt.event.ActionListener () {
            public void actionPerformed (ActionEvent e) {
                button3_actionPerformed (e);
            }
        });
        button4.setLabel ("Go");
        button4.setBounds (new Rectangle (5, 95, 90, 27));
        button4.addActionListener (new java.awt.event.ActionListener () {
            public void actionPerformed (ActionEvent e) {
                button4_actionPerformed (e);
            }
        });
        button5.setLabel ("Stop");
        button5.setBounds (new Rectangle (5, 125, 90, 27));
        button5.addActionListener (new java.awt.event.ActionListener () {
            public void actionPerformed (ActionEvent e) {
                button5_actionPerformed (e);
            }
        });
        panell.setBounds (new Rectangle (345, 0, 100, 300));
        panell.setLayout (null);
        choice1.setBounds (new Rectangle (15, 192, 66, 28));
        labell.setText ("Scheibenzahl");
        labell.setBounds (new Rectangle (11, 167, 90, 22));
        tmp.setText ("*0*");
        tmp.setBounds (new Rectangle (11, 11, 190, 22));
        panell.add(button1, null);
        panell.add(button2, null);
        panell.add(button3, null);
        panell.add(button4, null);
        panell.add(button5, null);
        panell.add(choice1, null);
        panell.add(labell, null);
        this.setBackground (Color.black);
        this.setForeground (Color.gray);
        this.setLayout (null);
        this.add(panell, null);
//        this.add(tmp, null); // Debug-Meldung
    }
    public void start() {
    }
    public void stop() {
    }
    public void destroy() {
    }
    public String getAppletInfo () {
        return "Applet Information" ;
    }
    public String[][] getParameterInfo () {
        return null;
    }
    public void paint(Graphics g)
    {
//waagerechte Linie
        g.drawRect (10,192,90,1);
        g.drawRect (110,192,90,1);
        g.drawRect (210,192,90,1);

//eine Senkrechte im Zwischenlager
        g.drawRect ( 55,130,1,61);
        g.drawRect (155,130,1,61);
        g.drawRect (255,130,1,61);
        int a=0;

//setzt die Balken
        for(int i=0;i<scheibenzahl ;i++)
        {
            g.drawImage
                (image,30-i*7+xPosBalken [i]*100,173-yPosBalken [i]*20,48+i*15,20 ,this );
        }
    }
//Loesung
    String Loesung (int anzahl ,String quelle ,String lager ,String ziel)

```

```

{
String loesung=new String();
loesung = quelle+ziel;
if (anzahl>1)
    loesung = Loesung (anzahl-1,quelle ,ziel ,lager)
        +loesung+Loesung (anzahl-1,lager ,quelle ,ziel);
return loesung ;
}
//manuelle init-funktion
private void minit()
{
scheibenzahl =3;
xPosBalken = new int[scheibenzahl ];
yPosBalken = new int[scheibenzahl ];
belegung=new int [3];
belegung [0]=1230;
belegung [1]=0;
belegung [2]=0;
pos=0;
loes=Loesung (3, "0", "2", "1" );

image=getImage (getCodeBase (), "at56b.gif" );
choicel .addItem ( "1" );
choicel .addItem ( "2" );
choicel .addItem ( "3" );
choicel .addItem ( "4" );
choicel .addItem ( "5" );
choicel .select (2);
xPosBalken [0]=0;
xPosBalken [1]=0;
xPosBalken [2]=0;
yPosBalken [0]=2;
yPosBalken [1]=1;
yPosBalken [2]=0;
}
//button-aktionen
void button1_actionPerformed (ActionEvent e) {
    lauf=false;
    scheibenzahl =new Integer
        (Integer.parseInt (choicel .getSelectedItem ())).intValue ();
    xPosBalken = new int[scheibenzahl ];
    yPosBalken = new int[scheibenzahl ];
    belegung =new int [3];
    String Sbelegung ="";
    for(int i=1;i<scheibenzahl +1;i++)
    {
        Sbelegung =Sbelegung +i;
    }
    Sbelegung =Sbelegung + "0";
    belegung [0]=new Integer (Integer.parseInt (Sbelegung)).intValue ();
    belegung [1]=0;
    belegung [2]=0;
    pos=0;
    loes=Loesung (scheibenzahl , "0", "2", "1" );
    for(int i=0;i<scheibenzahl ;i++)
    {
        xPosBalken [i]=0;
    }
    for(int i=0;i<scheibenzahl ;i++)
    {
        yPosBalken [i]=scheibenzahl -i-1;
    }
    tmp.setText ( "+++"+belegung [0]+"+++");
    repaint ();
}

void button2_actionPerformed (ActionEvent e) {
    if(anima==null)
    {
        tmp.setText ( "****"+pos+"****");
        anima=new Thread (this);
        anima.start ();
    }
    lauf=true;
}

```

```
}

void button3_actionPerformed (ActionEvent e) {
    if(pos>0)
        demo(false);
}

void button4_actionPerformed (ActionEvent e) {
    if(pos<(int)Math.pow(2,scheibenzahl)-1)
        demo(true);
}

void button5_actionPerformed (ActionEvent e) {
    lauf=false;
}
//Einzelschritt der Animation
void demo(boolean go) {
    int i=go?pos++:--pos;
    int q,z,j,temp;
    q=Integer.valueOf(String.valueOf(loes.charAt((i*2))).intValue());
    z=Integer.valueOf(String.valueOf(loes.charAt((i*2)+1)).intValue());
    if(!go)
    {
        temp=q;q=z;z=temp;
    }
    temp=0;
    for(j=scheibenzahl;j>0;j--)
    {
        temp=belegung[q]/(int)Math.pow(10,j);
        if(temp>0)
        {
            belegung[q]-=temp*(int)Math.pow(10,j);
            if(belegung[z]==0)
                j=1;
            else
                j=(int)(Math.log(belegung[z])/Math.log(10))+1;
            belegung[z]+=temp*(int)Math.pow(10,j);
            break;
        }
    }
    temp--;
    xPosBalken[temp]=z;
    yPosBalken[temp]=--j;
    repaint();
}
}
```