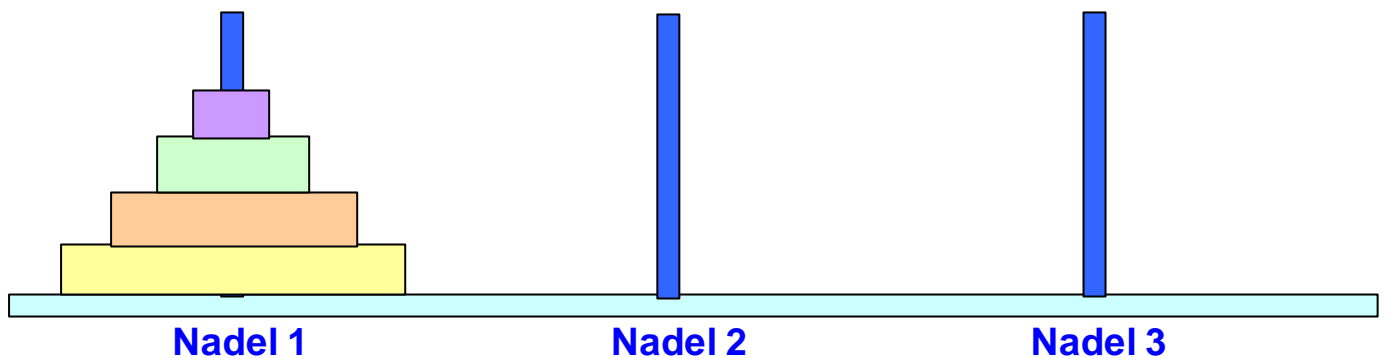


## Die Türme von Hanoi



Der französische Mathematiker Edouard Lucas hat 1883 eine kleine Geschichte erfunden, die unter dem Namen **Die Türme von Hanoi** weltberühmt wurde :

Im Großen Tempel von **Benares** , unter dem Dom, der die Mitte der Welt markiert, ruht eine **Messingplatte** , in der **drei Diamantnadeln** befestigt sind, jede eine Elle hoch und so stark wie der Körper einer Biene. Bei der Erschaffung der Welt hat Gott **vierundsechzig Scheiben aus purem Gold** auf eine der Nadeln gesteckt, wobei die größte Scheibe auf der Messingplatte ruht, und die übrigen, immer kleiner werdend, eine auf der anderen. Das ist der **Turm von Brahma**.

Tag und Nacht sind die Priester unablässig damit beschäftigt, den festgeschriebenen und unveränderlichen **Gesetzen von Brahma** folgend, die Scheiben von einer Diamantnadel auf eine andere zu setzen, **wobei die Priester nur jeweils eine Scheibe auf einmal umsetzen dürfen, und zwar so, dass sich nie eine kleinere Scheibe auf einer größeren zu liegen kommt.**

Sobald dereinst alle vierundsechzig Scheiben von der Nadel 1, auf die Gott sie bei der Erschaffung der Welt gesetzt hat, auf eine der anderen Nadeln gebracht worden sind, werden der Turm samt dem Tempel und allen Brahmanen zu Staub zerfallen, und die Welt wird mit einem Donnerschlag untergehen.

**Varanasi (Benares)**, liegt inmitten des Ganges-Beckens und ist einer der heiligsten Plätze des Landes und von daher die Bühne vieler historischer Epen und Mythen.

**Brahma** ist einer der hinduistischen Hauptgötter. Er wurde aus einem goldenen Ei geboren und ist der Schöpfer der Erde. Er hat vier Köpfe, die seinen vollständigen Überblick als Weltenschöpfer symbolisieren.



**Einst wurde der oberste Priester gefragt, ob er und seine untergebenen Priester (Schüler) genau wüssten, was sie zu tun hätten.**

**Daraufhin hat er die folgende Lösung für dieses Problem angegeben:**

1. Falls der Turm aus mehr als einer Scheibe besteht, bitte deinen ältesten Schüler, den darüber liegenden Turm von  $(n - 1)$  Scheiben von der ersten zur zweiten Nadel unter Verwendung der dritten Nadel als Zwischenablage umzusetzen.
2. Trage selbst die unterste Scheibe von der ersten zur dritten Nadel
3. Falls der Turm aus mehr als einer Scheibe besteht, bitte deinen ältesten Schüler, den Turm aus  $(n - 1)$  Scheiben von der zweiten zur dritten Nadel unter Verwendung der ersten Nadel als Zwischenablage umzusetzen.

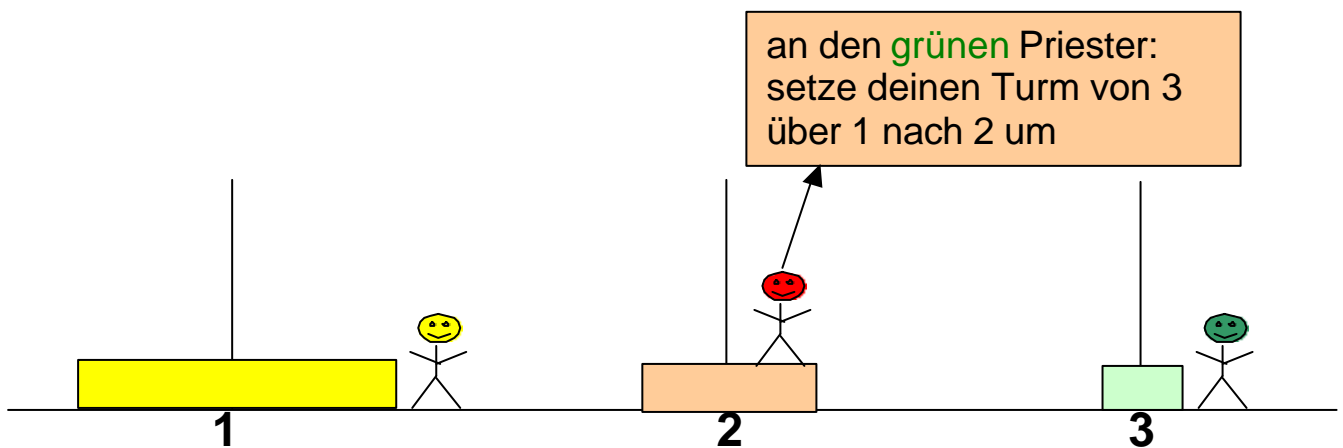
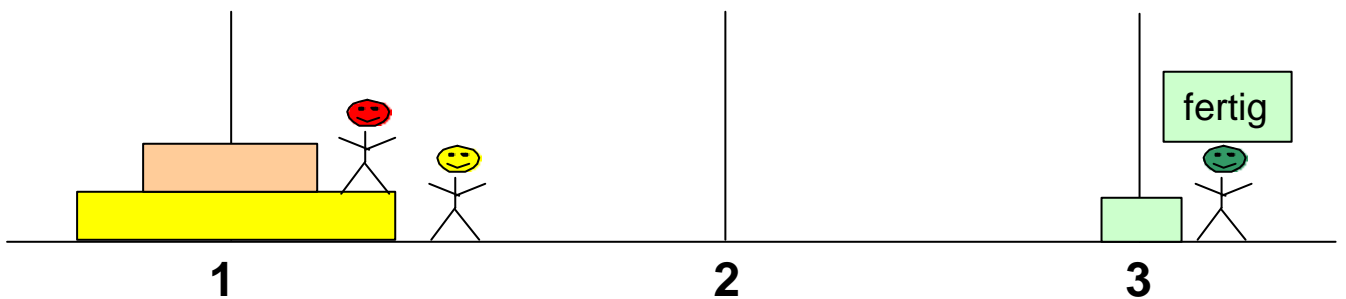
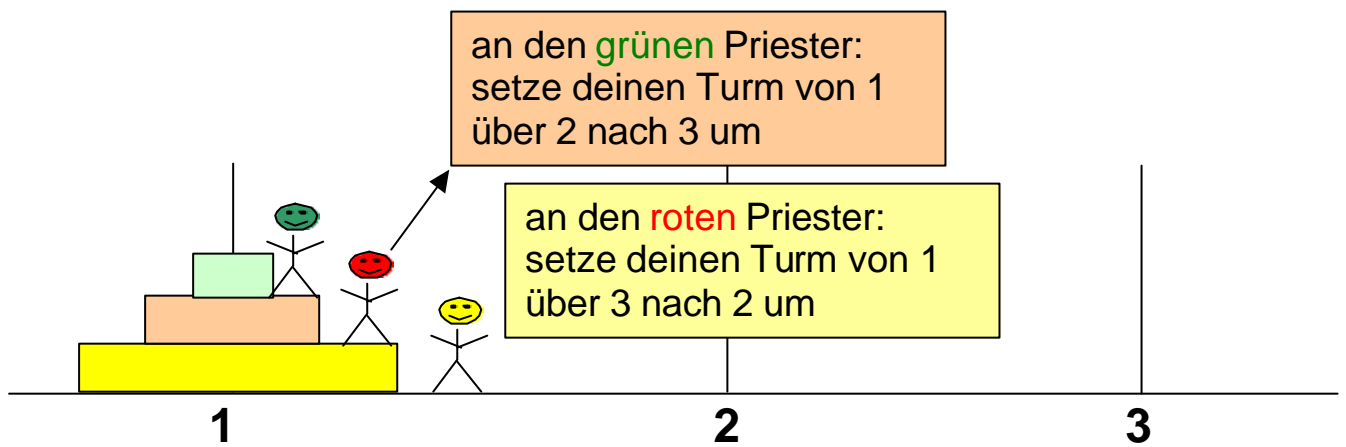
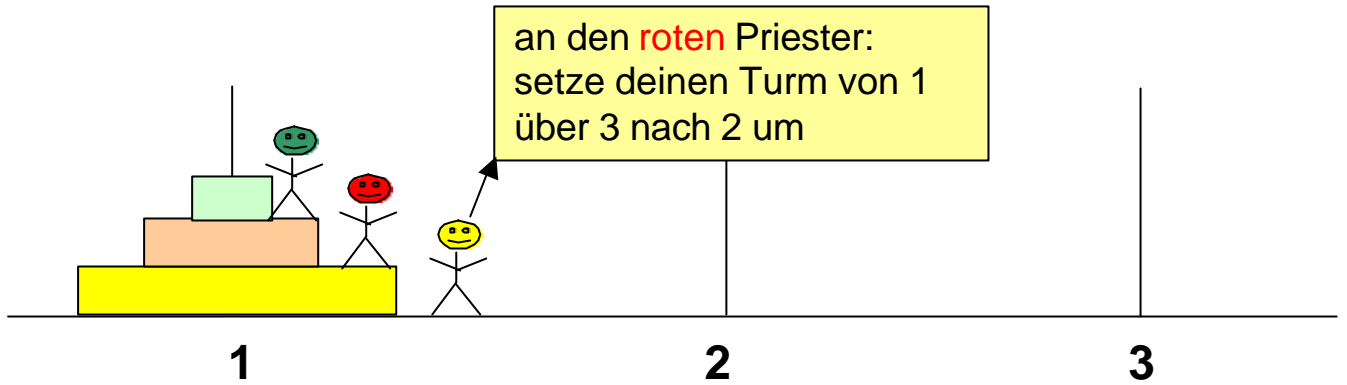
**Dies ist ein sehr elegantes Beispiel für eine rekursive Lösung, die für  $n=3$  wie folgt aussieht:**

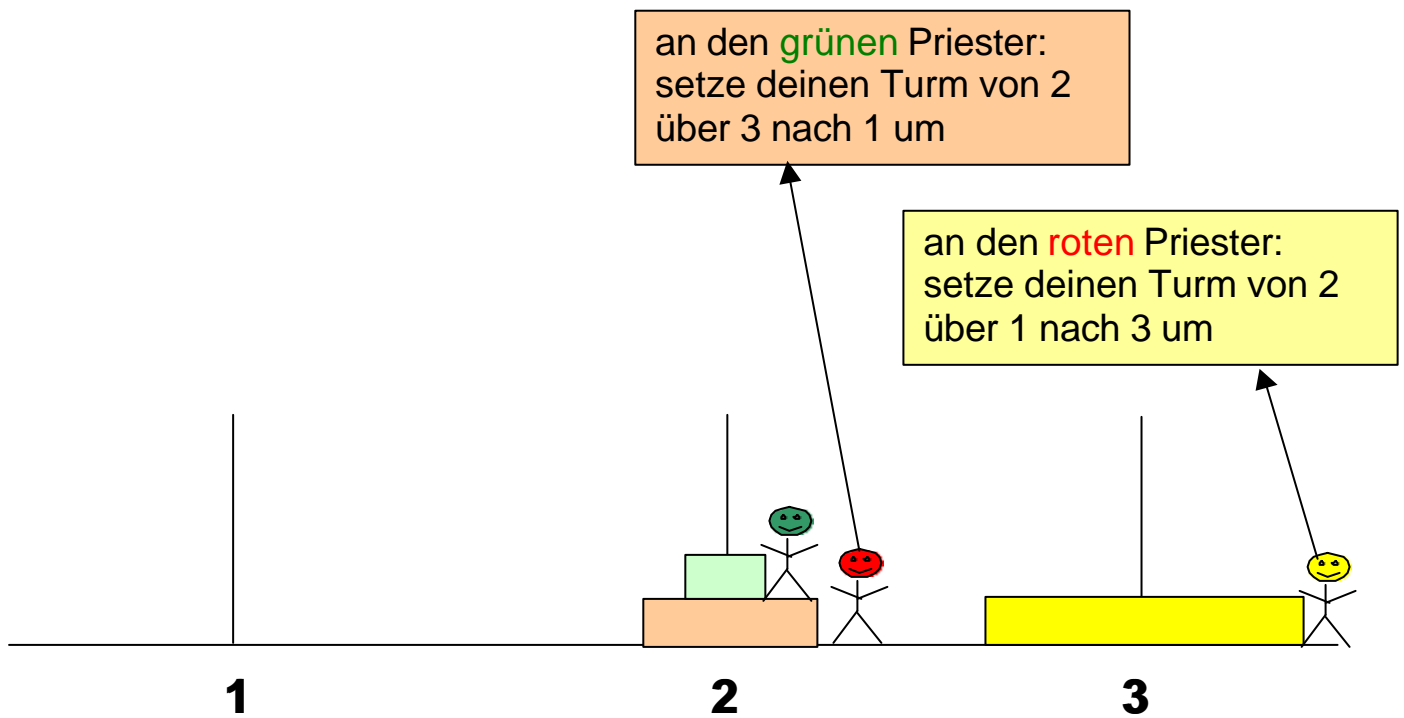
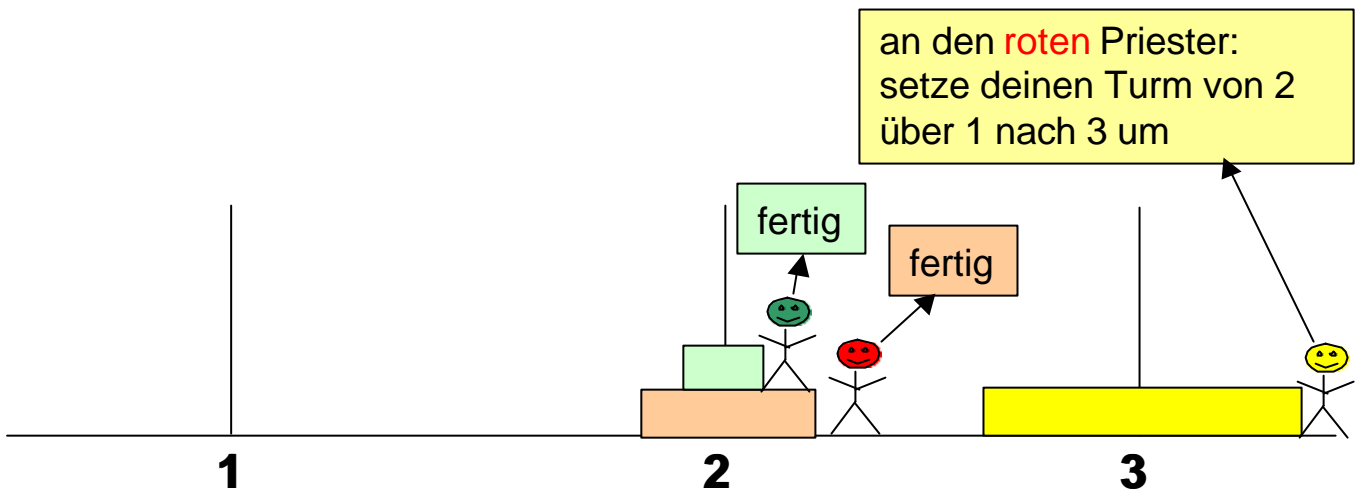
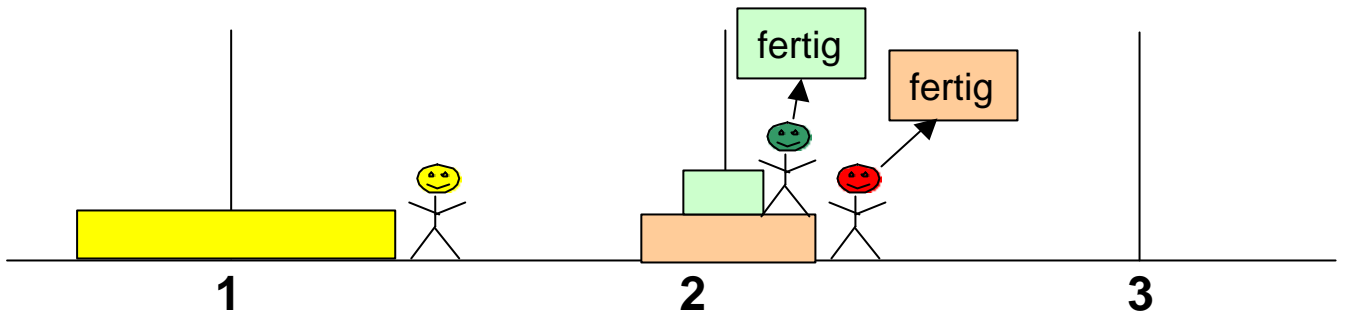
**Der oberste Priester (gelb) ist für die gesamte Versetzung des Turms mit 3 Scheiben von Nadel 1 über Nadel 2 nach Nadel 3 verantwortlich.**

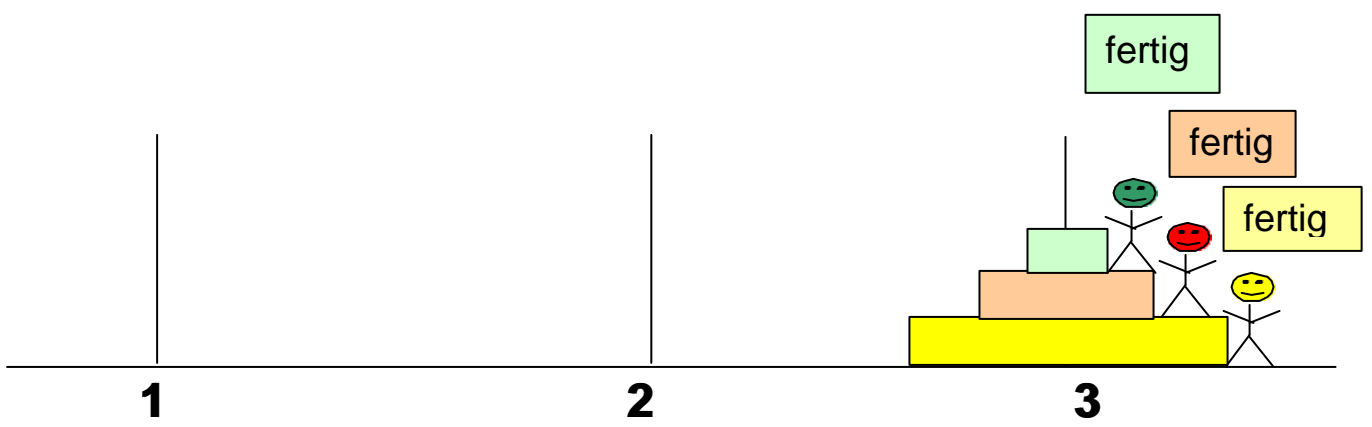
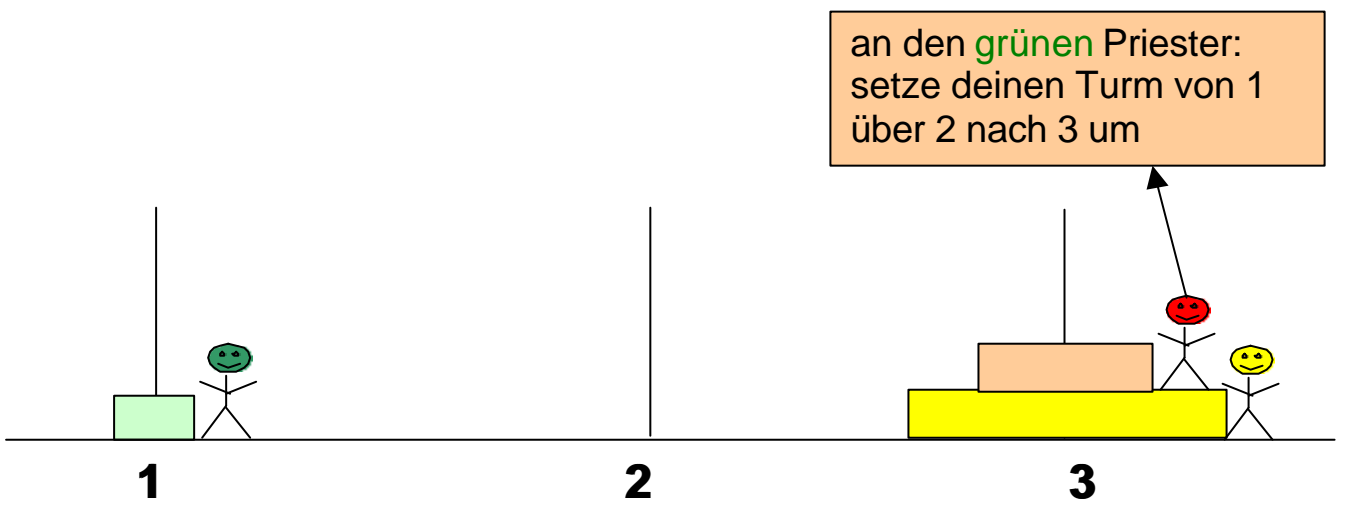
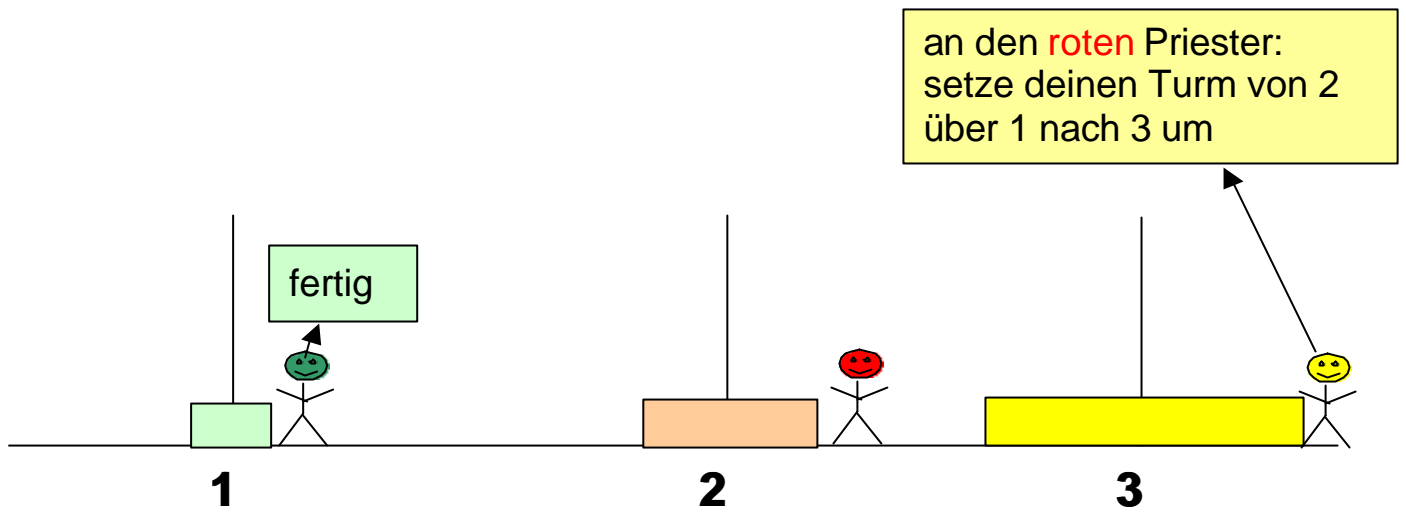
**Er selbst delegiert die Versetzung des Turms aus 2 Scheiben an seinen ältesten Schüler (rot).**

**Dieser wiederum delegiert die Versetzung des Turms mit einer Scheibe an seinen ältesten Schüler (grün).**

### Rekursive Lösung für n=3 Scheiben :







Für die Stapelhöhe  $n=3$  hat man also 7 Umsetzungen und insgesamt 10 rekursive Aufrufe.

Allgemein hat man für die Stapelhöhe  $n$  insgesamt mindestens  $2^n - 1$  Umsetzungen

**Beweis durch vollständige Induktion:**

**Verankerung :** Die Behauptung ist richtig für  $n=1$  denn für diesen Fall gibt es minimal  $2^1 - 1 = 2 - 1 = 1$  Umsetzung

**Induktionsvoraussetzung:** Die Behauptung sei richtig für ein beliebiges  $k$  mit  $1 < k < n$  d.h. für einen Turm mit  $k$  Scheiben gibt es mindestens  $2^k - 1$  Umsetzungen

**Induktionsschluß:** z.Z. für einen Turm der Höhe  $k+1$  gibt es dann mindestens  $2^{k+1} - 1$  Umsetzungen:  
 Nachweis:  
 Es gibt für den Turm der Höhe  $k+1$  gerade mindestens  $2^k - 1 + 2^k - 1 + 1 = 2 \cdot 2^k - 1 = 2^{k+1} - 1$  Umsetzungen

Für  $n=64$  hätte man damit

$$2^{64} - 1 = 18446744073709551615 \approx 1.8 \cdot 10^{19}$$

Umsetzungen

Wenn jede Scheibe innerhalb einer Sekunde umgesetzt wird, macht das ca. 580 Milliarden Jahre, also noch eine beruhigende Zeitspanne bis zum Ende der Welt.

```
import java.io.*;
```

```
public class TuermeVonHanoi_1
```

```
{
```

```
    static long rekursionstiefe=0;           // Klassenvariablen
```

```
    static long umlegungen;
```

```
public static void main(String[ ] args )throws IOException
```

```
{
```

```
    int stapelhoehe=3;
```

```
    verlegeDenTurm(stapelhoehe,1,3,2);       // von 1 nach 3 ueber 2
```

```
    System.out.println("Umlegungen : "+umlegungen);
```

```
    System.out.println("Rekursionstiefe : "+rekursionstiefe);
```

```
}
```

```
public static void verlegeDenTurm
```

```
    (int stapelhoehe,int von,int ueber,int nach)
```

```
{ rekursionstiefe++;
```

```
  if (stapelhoehe==1)
```

```
  {
```

```
    System.out.println("Verlege die oberste Scheibe von Nadel "  
                        + von+ " nach Nadel " + nach);
```

```
    umlegungen++;
```

```
  }
```

```
  else
```

```
  {
```

```
    verlegeDenTurm(stapelhoehe-1,von,nach,ueber);
```

```
    verlegeDenTurm(1,von,ueber,nach);
```

```
    verlegeDenTurm(stapelhoehe-1,ueber,von,nach);
```

```
  }
```

```
}
```

```
}
```

```

public class TuermeVonHanoi_2
{
    static long rekursionstiefe=0;
    static long umlegungen;

    public static void main(String[] args ) throws IOException
    {
        System.out.println("Die Tuerme von Hanoi           :           ");
        System.out.println("Gegeben sind drei Pfosten mit n Scheiben           ");

        System.out.println("Am Anfang liegen alle n Scheiben sortiert auf Pfosten 1   ");
        System.out.println("Am Ende sollen alle Scheiben genau so sortiert auf         ");
        System.out.println("Pfosten3 liegen.                                           ");
        System.out.println("Einschraenkung1: immer nur eine Scheibe wird bewegt      ");
        System.out.println("Einschraenkung2: es darf immer nur eine kleinere Scheibe ");
        System.out.println("                   auf eine groessere Scheibe gelegt werden ");
        System.out.println("                   ");

        String zahlstr;
        BufferedReader eingabe = new BufferedReader(
                                new InputStreamReader(System.in));
        System.out.print("Stapelhoehe auf Pfosten1 : ");
        zahlstr = eingabe.readLine();
        int stapelhoehe=Integer.parseInt(zahlstr);

        verlegeDenTurm(stapelhoehe,1,2,3);                      // von 1 ueber 2 nach 3
        System.out.print("Rekursionstiefe :"+rekursionstiefe+" ");
        System.out.println("Umlegungen      :"+umlegungen);
    }
}

public static void verlegeDenTurm(int stapelhoehe, int von, int ueber, int nach)
{
    rekursionstiefe++;
    if (stapelhoehe==1)
    {
        System.out.println("Verlege die oberste Scheibe von Turm " + von
                            + " nach Turm " + nach);
        umlegungen++;
    }
    else
    {
        System.out.println("Verlege( " +von+" , "+nach+" , "+ueber+" , "
                            +stapelhoehe+" )");
        verlegeDenTurm(stapelhoehe-1,von,nach,ueber);
        System.out.println("Verlege( " +von+" , "+ueber+" , "+nach+" , "
                            +stapelhoehe+" )");
        verlegeDenTurm(1,von,ueber,nach);
        System.out.println("Verlege( " +ueber+" , "+von+" , "+nach+" , "
                            +stapelhoehe+" )");
        verlegeDenTurm(stapelhoehe-1,ueber,von,nach);
    }
    System.out.println("Verlege( " +von+" , "+ueber+" , "+nach+" , "
                        +stapelhoehe+" fertig ! )");
}
}

```