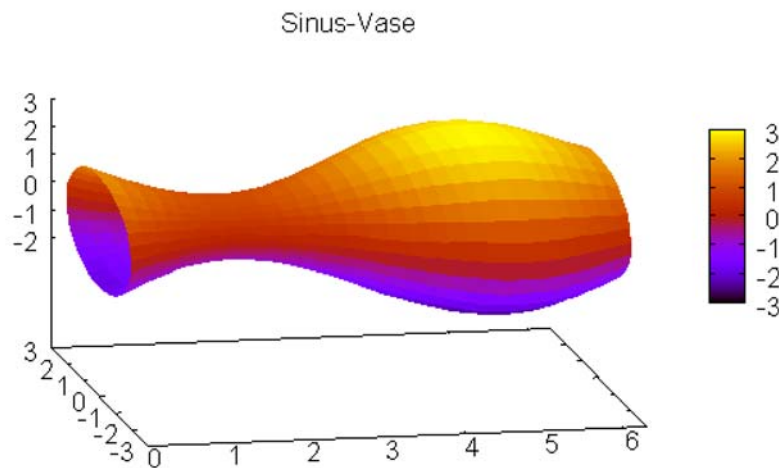


Figure 1:



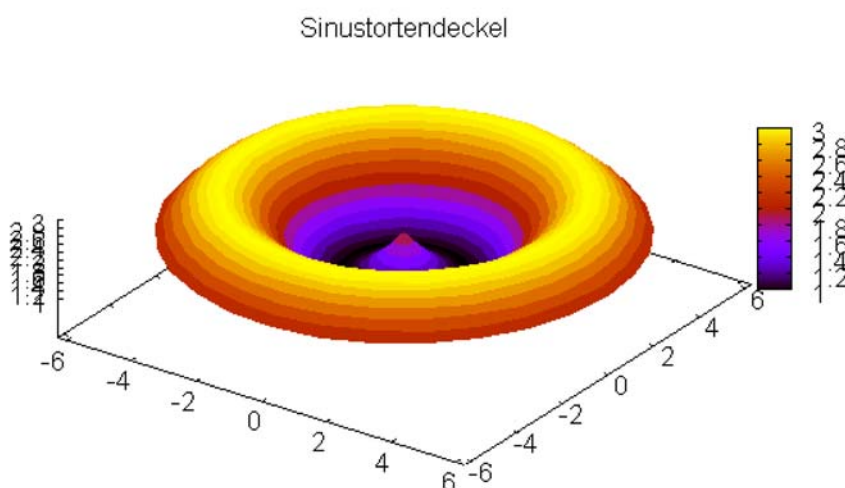
view: 68.0000, 349.000 scale: 1.00000, 1.00000

1.2 Schnelle Zeichenmethode, Rotation um die z-Achse

Es ist sinnvoll, "um z-Achse" zu sagen, da sich das Bild dann in die übliche Raumdarstellung einfügt.

```
--> draw3d(title      = "Sinustortendeckel",
            surface_hide = true,
            enhanced3d=true,
            parametric_surface(x *cos(t),
                               x*sin(t),
                               f(x),
                               t, 0, 2*%pi, x, 0, 2*%pi) )$
```

Figure 2:



view: 28.0000, 35.0000 scale: 1.00000, 1.00000

2 Erweiterte Methoden für mehrere Objekte

2.1 Erzeugung von Grafik-Objekten

Mit `gr3d(optionen, objekttyp)` werden sie erzeugt.
Die Erzeugung bewirkt noch keine Darstellung.

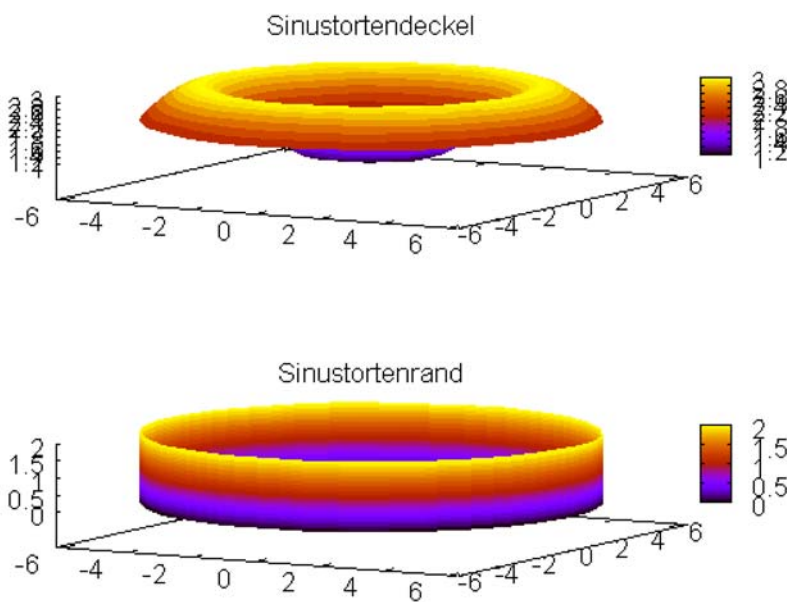
```
--> deckel:gr3d(title      = "Sinustortendeckel",
  surface_hide = true,
  enhanced3d=true,
  parametric_surface(x *cos(t),
    x*sin(t),
    f(x),
    t, 0, 2*%pi, x, 0, 2*%pi))$
```

```
--> rand:gr3d(title      = "Sinustortenrand",
  surface_hide = true,
  enhanced3d=true,
  parametric_surface(2*%pi *cos(t),
    2*%pi*sin(t),
    z,
    t, 0, 2*%pi, z, 0, 2))$
```

Der draw-Befehl stellt die Objekte dar.

```
--> draw(deckel,rand)$
```

Figure 3:



view: 60.0000, 30.0000 scale: 1.00000, 1.00000

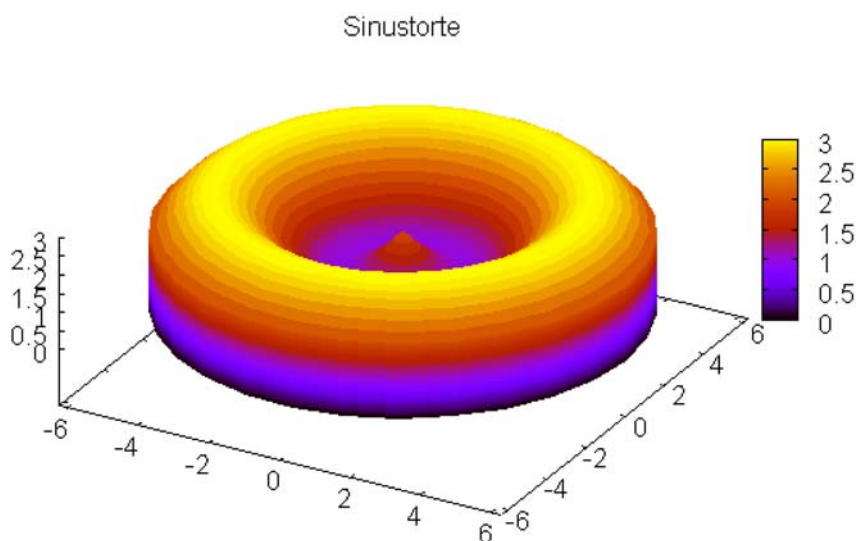
2.2 Mehrere Objekte in demselben Bild

Wenn man mehrere Objekte in ein und demselben Bild darstellen will, schreibt man sie alle in ein einziges Grafikobjekt.

```
--> torte:gr3d(title      = "Sinustorte",
  surface_hide = true,
  enhanced3d=true,
  parametric_surface(x *cos(t),
    x*sin(t),
    f(x),
    t, 0, 2*%pi, x, 0, 2*%pi),
  parametric_surface(2*%pi *cos(t),
    2*%pi*sin(t),
    z,
    t, 0, 2*%pi, z, 0, 2))$
```

```
--> draw(torte)$
```

Figure 4:



view: 37.0000, 29.0000 scale: 1.00000, 1.00000

2.3 Mehrere Objekttypen in demselben Bild

Das sieht schon gut aus. Nun wäre noch das erzeugende Kurvenstück als Raumkurve hilfreich. Es wird noch eingefügt und die dafür benötigten Optionen stehen davor.

```

--> torte_plus:gr3d(title      = "Sinustorte",
    surface_hide = true,
    enhanced3d=true,
    parametric_surface(x *cos(t),
        x*sin(t),
        f(x),
        t, 0, 2*%pi, x, 0, 2*%pi),
    parametric_surface(2*%pi *cos(t),
        2*%pi*sin(t),
        z,
        t, 0, 2*%pi, z, 0, 2),
    surface_hide=false,
    enhanced3d=false,
    line_width=3,
    color=black,
    parametric(x ,
        0,
        f(x),
        x, 0, 2*%pi))$

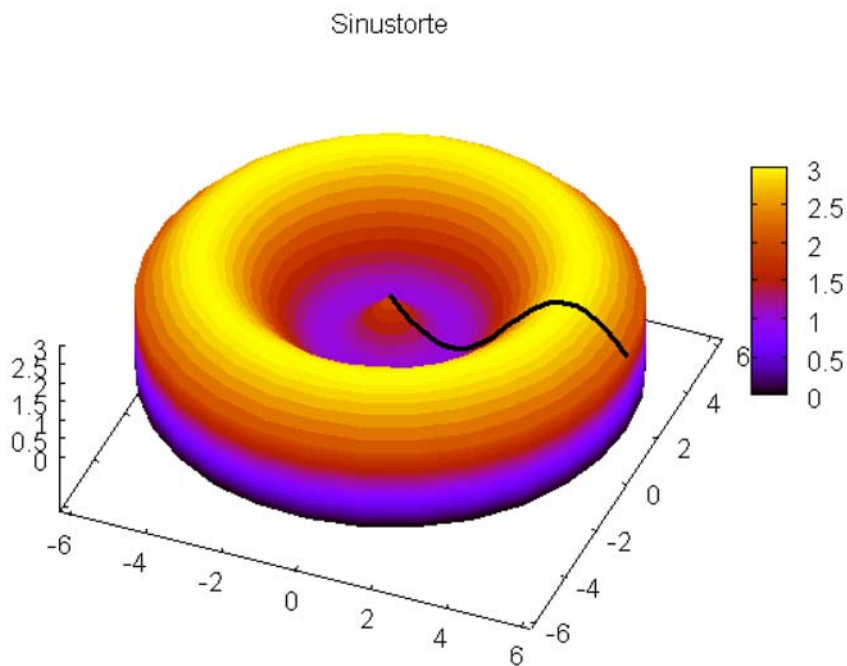
```

```

--> draw(torte_plus)$

```

Figure 5:



view: 28.0000, 22.0000 scale: 1.00000, 1.00000

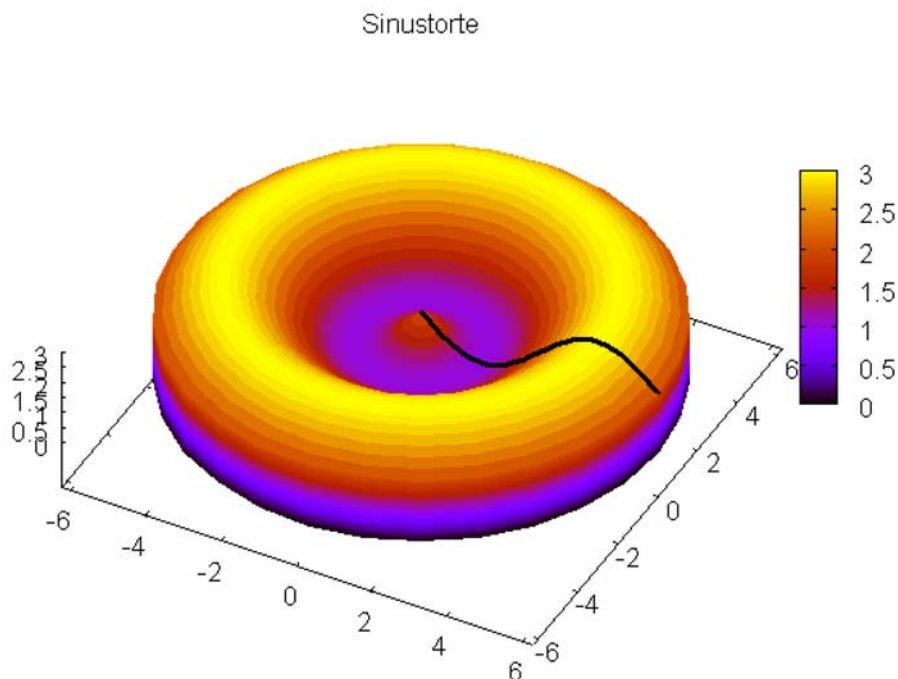
Damit kann man zufrieden sein. Im gnuplot kann man die Grafik bewegen, aber nach dem Transport über das Clippbord natürlich nicht mehr. Dann ist es nur ein Bild.

2.4 Weiteres zu den Optionen

surface_hide sorgt dafür, dass man Verdecktes nicht sieht.
 enhanced=true sorgt für die Farben entsprechend der Höhenlage.
 Man kann die Farben auch durch einen Term eines Parameters steuern.
 rot_horizontal und rot_vertical beziehen sich auf die Raumrichtung,
 aus der man die Grafik betrachtet. Hat man interaktiv eine gute Richtung
 gefunden (wie oben bei View zu sehen), kann man sie dauerhaft eintragen.

```
--> torte_plus:gr3d(title      = "Sinustorte",
    rot_horizontal=28, rot_vertical=22,
    surface_hide  = true, enhanced3d=true,
    parametric_surface(x *cos(t),
        x*sin(t),
        f(x),
        t, 0, 2*%pi, x, 0, 2*%pi),
    parametric_surface(2*%pi *cos(t),
        2*%pi*sin(t),
        z,
        t, 0, 2*%pi, z, 0, 2),
    surface_hide=false, enhanced3d=false,
    line_width=3, color=black,
    parametric(x ,
        0,
        f(x),
        x, 0, 2*%pi))$
```

```
--> draw(torte_plus)$
```



view: 22.0000, 28.0000 scale: 1.00000, 1.00000

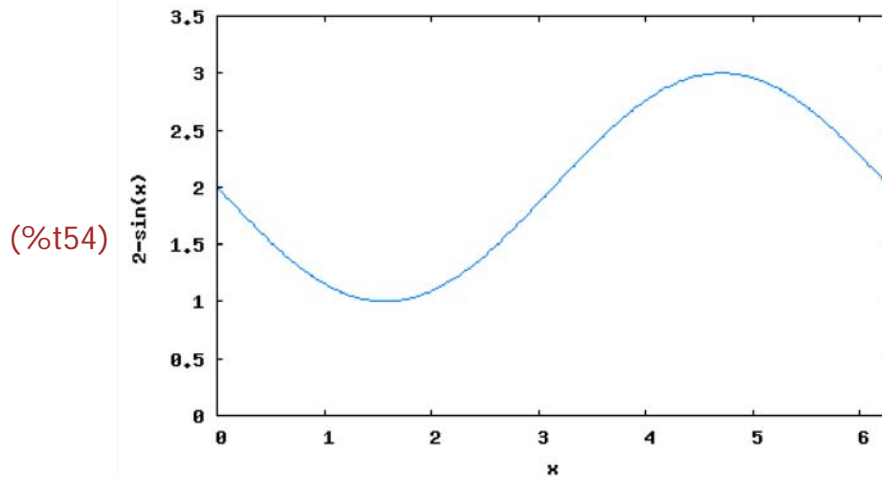
Die Zeichnung erscheint in einem Extrafenster "gnuplot graph".
 Oben links ist ein Icon mit Pulldown-Menü unter Options gibt es
 "ins Clipboard", vorher am besten kleiner ziehen.
 Dann kann man die zeichnung hier einfügen.
 Achtung, später unbedingt als wxmx speichern, sonst verschwinden
 die Zeichnungen.

3 Berechnung der Volumina

3.1 Rotation um die x-Achse

```
--> f(x);
(%o51) 2-sin(x)
```

```
--> wxplot2d([f(x)], [x,0,2*%pi], [y,0,3.5])$
```



```
--> Vx(x_term,x,a,b):=%pi*integrate(x_term^2,x,a,b)$
```

```
--> integrate(%pi*f(x)^2, x, 0, 2*%pi);
(%o11) 9 %pi^2
```

```
--> Vx(f(x),x,0,2*%pi);
(%o14) 9 %pi^2
```

3.2 Rotation um die z-Achse, Zylinderscheibchen

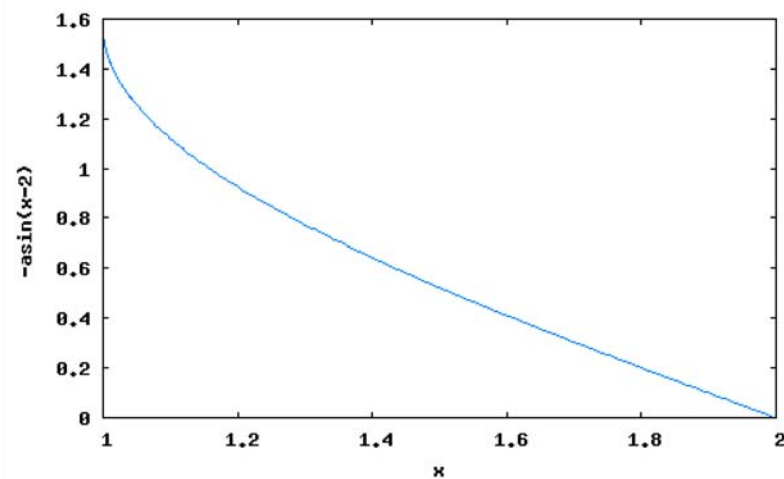
Diese Methode ist hier nicht gut anwendbar, da die Funktion nicht umkehrbar ist.

```
--> Vy(y_term,y,a,b):=%pi*integrate(y_term^2,y,y_a,y_b)$
```

Man kann hier allenfalls von 0 bis $\pi/2$ in der Originalfigur gehen

```
--> wxplot2d([asin(2-x)], [x,1,2])$
```

(%t74)



```
--> %pi*integrate(asin(2-y),y,1,2) /*aber das nützt nichts, Quadrat fehlt*/;
```

(%o87) $\left(\frac{\%pi}{2} - 1\right) \%pi$

Man muss direkt überlegen, dass das gesuchte Volumen ist:

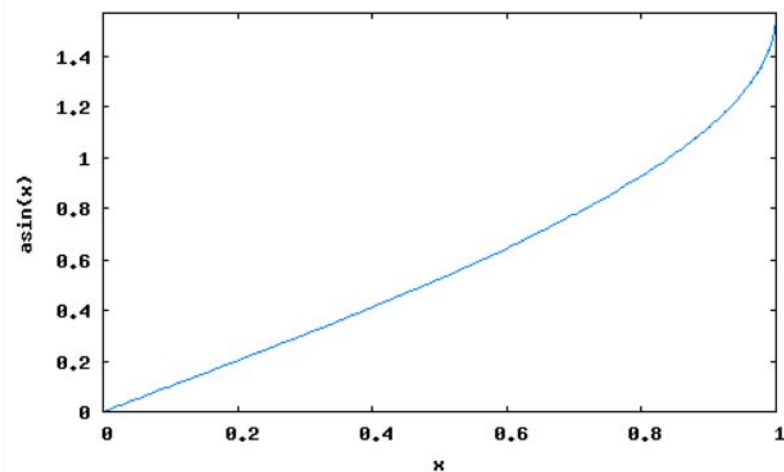
```
--> integrate(asin(x)^2,x);
```

(%o101) $x \operatorname{asin}(x)^2 + 2\sqrt{1-x^2} \operatorname{asin}(x) - 2x$

Bei Rotation um die x-Achse

```
--> wxplot2d([asin(x)], [x,0,1], [y,0,%pi/2])$
```

(%t102)



```
--> vol:%pi*integrate(asin(x)^2,x,0,1);%numer;
```

(%o135) $\frac{\%pi (\%pi^2 - 8)}{4}$

(%o136) 1.468383862895368

Mit Unterteil

```
--> %pi*(%pi^2)/4*1;
```

(%o144) $\frac{\%pi^3}{4}$


```
--> vol +%;expand(%)%;numer;
```

```
(%o145)  $\frac{\%pi^3}{4} + \frac{\%pi (\%pi^2 - 8)}{4}$ 
```

```
(%o146)  $\frac{\%pi^3}{2} - 2 \%pi$ 
```

```
(%o147) 9.219953032970322
```

Dieses Ergebnis ergibt sich mit der unteren Methode 3.3 viel einfacher.

Und mit der Quadrierung geht es nicht in der verschobenen Form.

```
--> Vy(asin(2-y),y,1,2);
```

```
sign: argument cannot be imaginary; found  $\sqrt{y-3}$ 
```

```
#0: Vy(y_term=-asin(y-2),y=y,a=1,b=2)
```

```
-- an error. To debug this try: debugmode(true);
```

3.3 Rotation um die z-Achse, Hohlzylinderchen

```
--> Vz(x_term,x,a,b):=2*%pi*integrate(x*x_term,x,a,b)$
```

```
--> vol:Vz(f(x),x,0,2*%pi); expand(vol);factor(vol);vol,numer;
```

```
(%o16)  $2 \%pi (4 \%pi^2 + 2 \%pi)$ 
```

```
(%o17)  $8 \%pi^3 + 4 \%pi^2$ 
```

```
(%o18)  $4 (2 \%pi + 1) \%pi^2$ 
```

```
(%o19) 287.528631046756
```

Zwei Vergleichszylinder

```
(%i5) %pi*(2*%pi)^2, numer;  
      %*3 ;
```

```
(%o5) 124.0251067211993
```

```
(%o6) 372.0753201635978
```

3.4 Mit Hohlzylindern im umkehrbaren Bereich

```
--> Vz(f(x),x,0,%pi/2); expand(%)factor(%)%;numer;
```

```
(%o111)  $\frac{\%pi (\%pi^2 - 4)}{2}$ 
```

```
(%o112)  $\frac{\%pi^3}{2} - 2 \%pi$ 
```

```
(%o113)  $\frac{(\%pi - 2) \%pi (\%pi + 2)}{2}$ 
```

```
(%o114) 9.219953032970322
```

Das ist genau der oben viel mühsamer erzeugte Wert.