# lektion2

January 29, 2018

Table of Contents



## 1 Lektion 2

```
In [103]: from sympy import *
          init_printing()
```

### 1.1 Zahlen, Symbole

```
In [104]: 1+1
```

Out[104]:

$$2$$

```
In [105]: 2*3
```

Out[105]:

$$6$$

```
In [106]: 1
```

Out[106]:

$$1$$

```
In [107]: type(1)
```

```
Out[107]: int
```

```
In [108]: 1.0
```

Out[108]:

$$1.0$$

```
In [109]: type(1.0)

Out[109]: float

In [110]: 1+1j

Out[110]: (1+1j)

In [111]: type(1+1j)

Out[111]: complex

In [112]: drittel = Rational(1,3)

In [113]: type(drittel)

Out[113]: sympy.core.numbers.Rational

In [114]: N(pi,20) # Kreiszahl
```

Out[114]:

$$3.1415926535897932385$$

```
In [115]: N(EulerGamma,40) # Euler Gamma
```

Out[115]:

$$0.5772156649015328606065120900824024310422$$

```
In [116]: N(Catalan,30) # Catalans Konstante
```

Out[116]:

$$0.915965594177219015054603514932$$

```
In [117]: N(E,20) # Basis des nat. Logarithmus
```

Out[117]:

$$2.7182818284590452354$$

```
In [118]: s1 = 1

In [119]: type(s1)
```

```
Out[119]: int
```

```
In [120]: a = Symbol('a')
          a
```

Out[120]:

$$a$$

```
In [121]: x,y,z = symbols('x y z')
          f = y*x**z+1
          f
```

Out[121]:

$$x^z y + 1$$

```
In [122]: a,b = symbols('b a') # nicht nachmachen
          print('a ist :', a, 'und b ist', b)
```

```
a ist : b und b ist a
```

```
In [123]: xx, yy = symbols('xx yy',commutative=False)
          xx*yy == yy*xx
```

```
Out[123]: False
```

```
In [124]: cos(m*pi)
```

Out[124]:

$$(-1)^m$$

```
In [125]: m = symbols('m',integer=True)
          m.assumptions0
```

```
Out[125]: {'algebraic': True,
           'commutative': True,
           'complex': True,
           'hermitian': True,
           'imaginary': False,
           'integer': True,
           'irrational': False,
           'noninteger': False,
           'rational': True,
           'real': True,
           'transcendental': False}
```

```
In [126]: a = Symbol('a',positive=True)
          sqrt(a**2), sqrt(b**2)
```

Out[126]:

$$\left(a, \quad \sqrt{a^2}\right)$$

In [127]:
```
n = Symbol('n',positive=True,integer=True)
n.assumptions0
```

Out[127]:
```
{'algebraic': True,
 'commutative': True,
 'complex': True,
 'hermitian': True,
 'imaginary': False,
 'integer': True,
 'irrational': False,
 'negative': False,
 'noninteger': False,
 'nonnegative': True,
 'nonpositive': False,
 'nonzero': True,
 'positive': True,
 'rational': True,
 'real': True,
 'transcendental': False,
 'zero': False}
```

## 1.2 Tupel, Listen, Mengen, Dictionaries

In [128]:
```
tupel = (1,2,'p',pi,2,cos(x),(3,4))
tupel
```

Out[128]:
```
(1, 2, 'p', pi, 2, cos(x), (3, 4))
```

In [129]:
```
tupel[0]
```

Out[129]:

$$1$$

In [130]:
```
tupel
```

Out[130]:
```
(1, 2, 'p', pi, 2, cos(x), (3, 4))
```

In [131]:
```
liste = [1,2,'p',pi,2,cos(x),(3,4)]
liste
```

Out[131]:
```
[1, 2, 'p', pi, 2, cos(x), (3, 4)]
```

In [132]:
```
liste[3]
```

Out[132]:

$$\pi$$

In [133]:
```
liste[3] = E
liste
```

Out[133]:
```
[1, 2, 'p', E, 2, cos(x), (3, 4)]
```

In [134]:
```
menge = {1,2,pi,sin(x),cos(x),cos(-x),1,2}
menge
```

Out[134]:

$$\{1, 2, \pi, \sin(x), \cos(x)\}$$

In [135]:
```
leereMenge = set() # nicht {}
leereMenge
```

Out[135]:

$$\{\}$$

In [136]:
```
pi in menge
```

Out[136]:
```
True
```

In [137]:
```
dictionary = {'Alice': 1234, 'Bob': 1224, 'Eve' : 2346}
dictionary['Alice']
```

Out[137]:

$$1234$$

In [138]:
```
list(dictionary.keys())
```

Out[138]:
```
['Alice', 'Bob', 'Eve']
```

### 1.3 Sympifizierung

In [139]:
```
## Sympifizierung

f = sympify("a**2*sin(x*pi)+.09*I*10")
f
```

Out[139]:

$$a^2 \sin(\pi x) + 0.9i$$

In [140]:
```
f = sympify("a**2*sin(x*pi)+.09*I*10")
f
```

Out[140]:

$$a^2 \sin(\pi x) + 0.9i$$

In [141]:
```
g = S('(x**2-3)/((x-2)*(x+1))')
g
```

Out[141]:

$$\frac{x^2 - 3}{(x-2)(x+1)}$$

## 1.4 Zerlegen von Ausdrücken

In [142]:
```
f.args
```

Out[142]:

$$\left(0.9i, \quad a^2 \sin(\pi x)\right)$$

In [143]:
```
f.args[0] # Hilft nicht recht weiter
```

Out[143]:

$$0.9i$$

In [144]:
```
f.atoms()
```

Out[144]:

$$\{0.9, 2, i, \pi, a, x\}$$

In [145]:
```
f.atoms(Symbol)
```

Out[145]:

$$\{a, x\}$$

In [146]:
```
f.atoms(Number)
```

Out[146]:

$$\{0.9, 2\}$$

In [147]:
```
f.atoms(NumberSymbol)
```

Out[147]:

$$\{\pi\}$$

In [148]:
```
f.atoms(I)
```

Out[148]:

$$\{i\}$$

In [149]: f

Out[149]:

$$a^2 \sin(\pi x) + 0.9i$$

In [150]: f.as_coeff_add()[1][0]

Out[150]:

$$0.9i$$

In [151]: g = f.as_coeff_add()[1][1]
g

Out[151]:

$$a^2 \sin(\pi x)$$

In [152]: g.as_coeff_mul()

Out[152]:

$$\left(1, \quad \left(a^2, \quad \sin(\pi x)\right)\right)$$

## 1.5 Auswertung von Ausdrücken

In [153]: g

Out[153]:

$$a^2 \sin(\pi x)$$

In [154]: g.subs(x,1) # Ersetzung von x durch 1 in g

Out[154]:

$$0$$

In [155]: h = 2*x*y+x-y
h

Out[155]:

$$2xy + x - y$$

In [156]:
```
h.subs(y,2).subs(x,1) # mehrfache Ersetzung
```

Out[156]:

$$3$$

In [157]:
```
h.subs({x:1,y:2}) # mehrfache Ersetzung mit Dictinary
```

Out[157]:

$$3$$

In [158]:
```
h.subs([(x,1),(y,2)]) # noch eine Alternative mit Liste von Tupeln
```

Out[158]:

$$3$$

In [159]:
```
h.subs(x,1)
```

Out[159]:

$$y + 1$$

In [160]:
```
s = sqrt(8)
s
```

Out[160]:

$$2\sqrt{2}$$

In [161]:
```
s.evalf() # numerische Auswertung
```

Out[161]:

$$2.82842712474619$$

In [162]:
```
cosc = (cos(x)-1)/x
cosc
```

Out[162]:

$$\frac{1}{x}\left(\cos\left(x\right) - 1\right)$$

In [163]:
```
cosc.evalf(subs={x: 1})
```

Out[163]:

$$-0.45969769413186$$

```
In [164]: cosc.subs(x,0) # Achtung Falle
```

Out[164]:

$$\text{NaN}$$

```
In [165]: cosc.evalf(subs={x:0})
```

Out[165]:

$$0$$

## 1.6 Grenzwerte

```
In [166]: cosc.limit(x,0)
```

Out[166]:

$$0$$

```
In [167]: limit(cosc,x,0)
```

Out[167]:

$$0$$

```
In [168]: Limit(cosc,x,0) # träger Operator
```

Out[168]:

$$\lim_{x \to 0^+} \left( \frac{1}{x} \left( \cos\left(x\right) - 1 \right) \right)$$

```
In [169]: L = Limit(1/x,x,0,'+')
          L
```

Out[169]:

$$\lim_{x \to 0^+} \frac{1}{x}$$

```
In [170]: L.doit()
```

Out[170]:

$$\infty$$

```
In [171]: limit(1/x,x,0,'-')
```

Out[171]:

$$-\infty$$

```
In [172]: b = factorial(n)/sqrt(n)*(E/n)**n
          b
```

Out[172]:

$$\frac{\left(\frac{e}{n}\right)^n n!}{\sqrt{n}}$$

```
In [173]: L = Limit(b,n,oo)
          L
```

Out[173]:

$$\lim_{n\to\infty}\left(\frac{\left(\frac{e}{n}\right)^n n!}{\sqrt{n}}\right)$$

```
In [174]: L.doit() # das ist leider falsch
```

Out[174]:

$$0$$

```
In [175]: bb = b.subs(n,10000)
          bb.evalf()
```

Out[175]:

$$2.50664916328698$$

```
In [176]: bb = b.subs(n,10000) - sqrt(2*pi)
          bb.evalf()
```

Out[176]:

$$2.08886559842412 \cdot 10^{-5}$$

## 1.7 Summen

```
In [177]: s = Sum(m,(m,0,n))
          s
```

Out[177]:

$$\sum_{m=0}^{n} m$$

```
In [178]: s.doit()
```

Out[178]:

$$\frac{n^2}{2} + \frac{n}{2}$$

In [179]:
```
factor(s.doit())
```

Out[179]:

$$\frac{n}{2}\left(n+1\right)$$

In [180]:
```
s = Sum(x**m,(m,0,n))
s
```

Out[180]:

$$\sum_{m=0}^{n} x^{m}$$

In [181]:
```
s.doit()
```

Out[181]:

$$\begin{cases} n+1 & \text{for } x = 1 \\ \frac{-x^{n+1}+1}{-x+1} & \text{otherwise} \end{cases}$$

In [182]:
```
s = Sum(x**m,(m,0,oo))
s
```

Out[182]:

$$\sum_{m=0}^{\infty} x^{m}$$

In [183]:
```
s.doit()
```

Out[183]:

$$\begin{cases} \frac{1}{-x+1} & \text{for } |x| < 1 \\ \sum_{m=0}^{\infty} x^{m} & \text{otherwise} \end{cases}$$

In [184]:
```
s.doit().args
```

Out[184]:

$$\left(\left(\frac{1}{-x+1}, \quad |x| < 1\right), \quad \left(\sum_{m=0}^{\infty} x^{m}, \quad \text{True}\right)\right)$$

In [185]:
```
s.doit().args[0]
```

Out[185]:

$$\left(\frac{1}{-x+1}, \quad |x| < 1\right)$$

In [186]:
```
s.doit().args[0][0]
```

Out[186]:

$$\frac{1}{-x+1}$$

In [187]:
```
s = Sum((-1)**(m+1)/m,(m,1,oo))
s
```

Out[187]:

$$\sum_{m=1}^{\infty} \frac{1}{m} (-1)^{m+1}$$

In [188]:
```
s.doit()
```

Out[188]:

$$\log(2)$$

In [189]:
```
s = Sum(1/m**2,(m,1,oo))
s
```

Out[189]:

$$\sum_{m=1}^{\infty} \frac{1}{m^2}$$

In [190]:
```
s.doit()
```

Out[190]:

$$\frac{\pi^2}{6}$$

## 1.8 Integrale

In [191]:
```
f = x**2/sqrt(4-x**2)
f
```

Out[191]:

$$\frac{x^2}{\sqrt{-x^2+4}}$$

In [192]:
```
F = Integral(f,(x,0,2))
F
```

Out[192]:

$$\int_0^2 \frac{x^2}{\sqrt{-x^2+4}}\, dx$$

In [193]:
```
F.doit()
```

Out[193]:

$$\pi$$

In [194]:
```
f.integrate((x,0,2))
```

Out[194]:

$$\pi$$

In [195]:
```
F = Integral(f,x)
F
```

Out[195]:

$$\int \frac{x^2}{\sqrt{-x^2+4}}\,dx$$

In [196]:
```
FF = F.doit()
FF
```

Out[196]:

$$-\frac{x}{2}\sqrt{-x^2+4} + 2\,\mathrm{asin}\left(\frac{x}{2}\right)$$

In [197]:
```
limit(FF,x,0), limit(FF,x,2)
```

Out[197]:

$$(0, \quad \pi)$$

In [198]:
```
ff = diff(FF,x)
ff
```

Out[198]:

$$\frac{x^2}{2\sqrt{-x^2+4}} - \frac{1}{2}\sqrt{-x^2+4} + \frac{1}{\sqrt{-\frac{x^2}{4}+1}}$$

In [199]:
```
simplify(ff)
```

Out[199]:

$$\frac{x^2}{\sqrt{-x^2+4}}$$

In [200]:
```
integrate(exp(-x**2),x)
```

Out[200]:

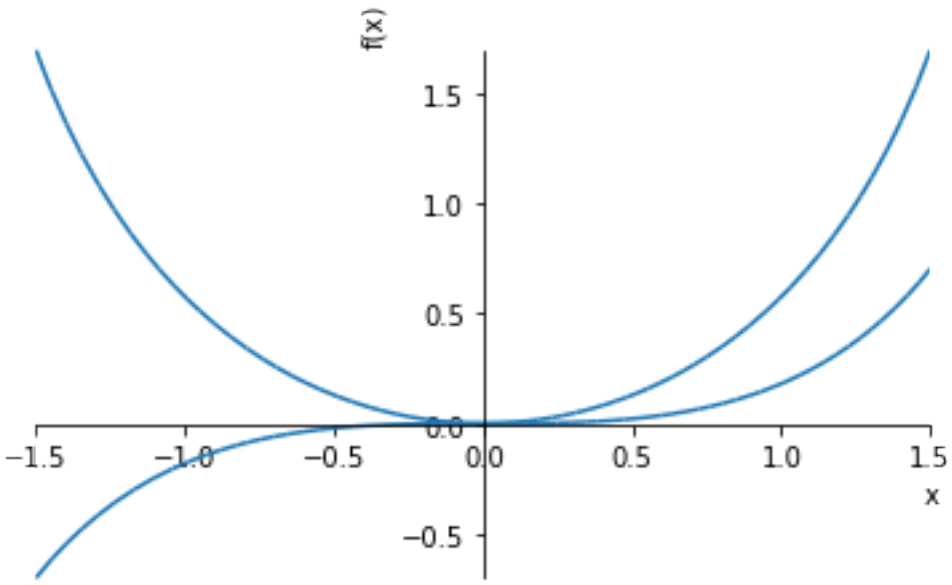
$$\frac{\sqrt{\pi}}{2}\operatorname{erf}(x)$$

In [201]: integrate(exp(-x**2),(x,-oo,oo))

Out[201]:

$$\sqrt{\pi}$$

In [202]: plot(f,FF,(x,-1.5,1.5))




Out[202]: <sympy.plotting.plot.Plot at 0x7f8c634b9080>

In [203]: ?plot