SAGE-Tutorium 05 im SoSe 2009

Lars Fischer*

27.05.2009

Inhaltsverzeichnis

1 Wiederholung

- mrange und Iteratoren

*WWW: `http://w3.countnumber.de/fischer`, EMail: vorname.nachname (bei der) uni-siegen.de

- Variablen und Referenzen
- Funktionen höherer Ordnung am Beispiel der Arithmetischen Funktionen

2 LaTeX und SAGE

Es lassen sich in SAGE einfache LaTeX Anweisungen verwenden. Damit kann man den erläuternden Text zwischen den einzelnen Zellen mit korrekten mathematischen Symbolen anreichern.

(Das eignet sich z.B. hervorragend für die Abgabe der theoretischen Aufgaben.)

```
%latex
Die ganzen Zahlen $\mathbb{Z}$.

$\alpha, \beta, \gamma$

$\frac{2}{3}, \frac{3}{4}, \binom{n}{k}$

$$ \left\{ k \in \mathbb{Z} \mid k \equiv{0} \pmod{7} \right\} = 7 \mathbb{Z} $$
```

Mit einem `%hide` wird sogar der Zellinhalt versteckt.

Eine andere Möglichkeit ist es, den Text außerhalb einer Zelle einzugeben.

Dazu klickt man die dicke blaue Linie über eine Zellen an und hält dabei die Shift-Taste gedrückt.

3 Polynome

Als Standardeinstellung ist `x` als symbolische Variable definiert.

```
type(x)
```

```
reset()
# x=RR[x].0
f= 3*x^3-5*x+3
print f
print f.roots()
plot(f)
```

Wenn wir ein Polynom über einem bestimmten Grundring betrachten wollen, so müssen wir den Polynomring genauer angeben:

```
R=IntegerModRing(13)
P.<x>= PolynomialRing(R)
print P
print type(x)
```

```
f= 3*x^3-5*x+3
print f
print f.roots()
plot(f)
```

Den Polynomring über $\mathbb{Q}$, also $\mathbb{Q}[x]$, bzw. den Polynomring über $\mathbb{R}$, also $\mathbb{R}[x]$ bekommen wir so:

```
PQ.<x>=QQ[x]
PR.<x> = RR[x]
```

Die Schreibweise mit den eckigen Klammern ist nur eine Kurzschreibweise für:

```
P.<x>= PolynomialRing(QQ, "x")
```

Und die spitzen Klammern sind eine Kurzschreibweise für:

```
# Definition von P und x in zwei Schritten
P = PolynomialRing(QQ, "x") # x als Zeichenkette
x=P.0                       # P.0 ist der erste Erzeuger

print P== QQ[x], P==RR[x]
```

Dabei taucht das x jeweils auf der linken und rechten Seite auf. Wir haben zum einen die Variable auf der linken Seite des Gleichheitszeichens, damit können wir die Variable in Ausdrücken verwenden und damit arbeiten.

Zum anderen gibt es die Variable auf der rechten Seite des Gleichheitszeichens. Sie wird für Ausgaben verwendet. Beide dürfen unterschiedlich sein. Auf der rechten Seite darf auch eine Zeichenkette stehen:

```
T= PolynomialRing(QQ, "zeta" )
z=T.0
print T
print 3*z^3 - 5 *z + 3
```

Wir sind nicht auf eine Variable festgelegt.

```
S=PolynomialRing( CC, "xy", 2)
x=S.0 # erster
y=S.1 # zweiter Erzeuger
print S
print x+y
print S==CC[x,y] # Kurzschreibweise
```

```
print z in T
print z in S
```

Bei der Schreibweise mit den spitzen Klammern, können wir die Variable auf der rechten Seite weglassen.

```
R.<b>=PolynomialRing(GF(2) ); print R
```

4 Grafik

Es stehen mehrere Befehle zum Plotten von Funktionen zur Verfügung. Die Dokumentation enthält jeweils viele Beispiele.

```
plot
plot3d
parametric_plot
parametric_plot3d
line
polygon
cirlce
list_plot
```

4.1 Funktionsgraphen

```
reset()
x=QQ[x].0
f=5*x^3-3*x+1
print f,type(x)



P=plot(f, -3,3 )
P+=plot(f.derivative(x), -3,3, color="red")
P+=plot(f.derivative(x,2), -3,3, color="green")

P.show()
```

4.2 Histogramme

Häufigkeits-Histogramme, können über die Funktion `rpy.histogram` erstellt werden. Leider liefert das nur die Einteilung und Auszählung der Klassen. Den Plot als Balkendiagramm müssen wir selber implementieren.

```
# Zufallsdaten erzeugen
L=[ randint(1,100) for i in range(100) ]
print L
```

```
import rpy
H= rpy.histogram(L)
# H= rpy.histogram(L, bins=4) # vier gleiche Klassen
# H= rpy.histogram(L, bins=[0,10,30,60,100]) # 0-10, 10-30,30-60, 60-100


# **args nimmt überzählige Argumente auf und reicht sie an Polygon weiter
def plot_histogram( H, drawLines=False, **args ):
    G= Graphics()
    for i in range(len(H[0])):
        links=  H[1][i]
        rechts= H[1][i+1]

        Ecken = [ ( links , 0    )  , (links  , H[0][i]),
                  ( rechts, H[0][i]), (rechts , 0 )       ]

        B= polygon( Ecken, **args)
        if drawLines:
            B+= line(Ecken + [Ecken[0]], rgbcolor="black", thickness=2 )
        G+= B
    return G


H= rpy.histogram(L, bins=[0,10,30,60,100])
P= plot_histogram( H )
#P= plot_histogram( H, rgbcolor="red", drawLines=True )
P.show()
```

4.3 3D-Plots

(Falls jmol im CIP-Pool funktioniert.)

```
reset()
u, v = var("u,v")

P1=plot3d(u^2 + v^2, (-2,2), (-2,2) )
P2=plot3d(sin(u*v), (u, -pi, pi), (v, -pi, pi))
P1.show()
#P2.show()
```

Möbius Band:

```
reset()
r,a=var("r,a2")
fx= cos(a)*(1+r/2*cos(a/2) )
fy= sin(a)*(1+r/2*cos(a/2) )
fz= r/2*sin(a/2)

M=parametric_plot3d([fx, fy, fz], (r, -1.1, 1.1), (a,0,2*pi) ,
                    frame=False, color="blue")
M.show()
```

Kleinsche Flasche:

```
reset()

u, v = var("u,v")

fx = (3*(1+sin(v)) + 2*(1-cos(v)/2)*cos(u))*cos(v)
fy = (4+2*(1-cos(v)/2)*cos(u))*sin(v)
fz = -2*(1-cos(v)/2) * sin(u)

K=parametric_plot3d([fx, fy, fz], (u,0, 2*pi), (v,0, 2*pi),
                    frame=False, color="green")
K.show()
```

5 Fragen zur Vorlesung

5.1 Quadratische Reste

Ein $a \in (\mathbb{Z}/m\mathbb{Z})^*$ heißt ein quadratischer Rest, falls es ein b gibt, mit $b^2 \equiv a \pmod{m}$.

Als Kurzschreibweise für eine ungerade Primzahl p wird das Legendresymbol definiert als:

$$\left(\frac{a}{p}\right) := \begin{cases} 1, & \text{falls eine Lösung } b^2 \equiv a \pmod{m} \text{ existiert} \\ 0, & \text{falls } p|a \\ -1, & \text{sonst} \end{cases}$$

Damit kann man die Tatsache, dass a ein quadratischer Rest mod p ist, ausdrücken als $\left(\frac{a}{p}\right) = 1$, bzw. ein quadratischer Nichtrest mod p ist, als $\left(\frac{a}{p}\right) = -1$. Kennt man die Primfaktorzerlegung von a, so lässt sich das Legendresymbol schnell berechnen (Multiplikativität ausnutzen, Quadratsche Reziprozität ausnutzen, Division mit Rest, um die Zahlen immer kleiner zu bekommmen, ...). Die Primfaktorzerlegung wird gebraucht, da das Legendre-Symbol nur über Primzahlen definiert ist.

Mit dem Jacobi-Symbol, welches die Erweiterung des Legendre-Symbols über zusammengesetzten Zahlen ist, umgeht man dieses Manko. Durch mehrmaliges Anwenden der quadratischen Reziprozität und der Division mit Rest, lässt sich das Jacobi Symbol effektiv berechnen, ohne die Faktorisierung von a zu kennen. Einzig muss der »Nenner« eines Jacobi-Symbols ungerade und positiv sein.

5.2 Rechenregeln für das Legendre-Symbol

Es gibt eine Reihe von Regeln (siehe [Niven, Zuckerman]), wie sich das Legendre-Symbol berechnen lässt:

- $\left(\frac{a}{p}\right) = a^{(p-1)/2} \pmod p$
- $\left(\frac{ab}{p}\right) = \left(\frac{a}{p}\right)\left(\frac{b}{p}\right)$
- falls a und p teilerfremd sind: $\left(\frac{a^2}{p}\right) = 1$, $\left(\frac{a^2b}{p}\right) = \left(\frac{b}{p}\right)$
- $a \equiv a' \pmod p \Rightarrow \left(\frac{a}{p}\right) = \left(\frac{a'}{p}\right)$

Für bestimmte Werte von a haben wir:

- $\left(\frac{1}{p}\right) = 1$
- $\left(\frac{0}{p}\right) = 0$
- $\left(\frac{-1}{p}\right) = (-1)^{(p-1)/2}$
- $\left(\frac{2}{p}\right) = (-1)^{(p^2-1)/8}$, hier kommt es nur auf den Rest von $p \pmod 8$ an. Da p eine ungerade Primzahl ist, gibt es nur die Reste $1, 3, 5, 7$.

Das quadratische Reziprozitäts-Gesetz für das Legendre-Symbol lautet: Es seien p, q verschiedene ungerade Primzahlen, dann gilt:

$$\left(\frac{p}{q}\right) = (-1)^{(p-1)(q-1)/4}\left(\frac{q}{p}\right)$$

Im [Niven, Zuckerman] wird das so beschrieben: Wir betrachten die beiden Kongruenzen:

1. $x^2 \equiv p \pmod{q}$
2. $x^2 \equiv q \pmod{p}$

und machen eine Fallunterscheidung:

- Sind die beiden ungeraden Primzahlen p, q beide kongruent 3 (mod 4), dann ist eine der Kongruenzen lösbar und die andere ist nicht lösbar.
- Ist eine der Primzahlen kongruent 1 (mod 4), so sind beide Kongruenzen lösbar oder beide nicht lösbar.

Es gilt durch Faktorisierung (gerade Zahlen sieht man es leicht an, dass sie durch 2 teilbar sind) und quadratische Reziprozität und euklidische Division, die Zahlen so klein zu bekommen, dass man es ihnen ansieht, ob sie ein Quadrat sind, oder auf $-1, 1, 2$ zu kommen.

6 Aufgaben

Aufgabe 1: Welche Elemente in $(\mathbb{Z}/5\mathbb{Z})^*$ sind quadratische Reste, welche nicht. Benutzt die Funktion `legendre_symbol`.

Aufgabe 2: Erstellt eine Tabelle: erste Spalte: p aus den Primzahlen < 100, zweite Spalte: $\left(\frac{5}{p}\right)$, dritte Spalte $5 \pmod{p}$.

Vergleicht mit den Ergebnissen aus der vorigen Aufgabe.

Aufgabe 3: Ergänzt die Tabelle aus der vorigen Aufgaben um eine weitere Spalte mit $\left(\frac{p}{5}\right)$.

Was hat das mit quadratischer Reziprozität zu tun?

Aufgabe 4: Ich hatte oben gesagt: »$\left(\frac{2}{p}\right) = (-1)^{(p^2-1)/8}$, hier kommt es nur auf den Rest von $p \pmod 8$ an«. Überlegt euch, dass diese Behauptung stimmt. (Hinweis: Da p eine ungerade Primzahl ist, gibt es nur die Reste $1, 3, 5, 7$.)

Für welche Werte von $p \pmod 8$ ist 2 quadratischer Rest?

Aufgabe 5: In dem quadratischen Reziprozitäts-Gesetz taucht die Potenz $\frac{(p-1)(q-1)}{4}$ auf.

Listet in eine Tabelle die verschiedenen Möglichkeiten für die Reste von q und q modulo 4 auf. (Hinweis: Beides sind ungerade Primzahlen.)

Berechnet für jede Möglichkeit, ob die Potenz gerade oder ungerade ist.

7 Nächstes Mal

- Vergabe der Projekte.
- Hinweise zur Arbeit mit SAGE außerhalb des Notebooks

8 Quellcode

Das gesamte Worksheet ist als Text-Datei in dem PDF eingebettet.

- Im Acrobat-Reader lässt es sich unter dem Büroklammer-Symbol in der linken Leiste herunterladen.
- Okular zeigt es im File-Menu als Embeded Files an.
- Unter Linux kann man die Text-Datei auch mit
  pdftk Tutorium05.pdf unpack_files
  aus dem PDF herauslösen.

Anschließend lässt sich die Text-Datei mit der Upload-Funktion des SAGE-Notebooks hochladen.