Einsatz von MuPAD für Schüler der gymnasialen Oberstufe - eine Kurzanleitung

Andreas Romeyke

10. Februar 1999

Inhaltsverzeichnis

1 Vorwort **3**

1.1 Stilvereinbarung . . . 3

1.2 Warum diese Anleitung? . . . 3

2 Vorbereitung **4**

2.1 Warum MuPAD? . . . 4

2.2 Woher bekommt man MuPAD? . . . 4

2.3 Wie wird MuPAD installiert? . . . 4

2.3.1 Linux . . . 5

2.3.2 Windows und Macintosh . . . 5

3 Tutorial **6**

3.1 Einfach loslegen . . . 6

3.1.1 Mathematische Konstanten . . . 7

3.1.2 Allgemeine arithmetische Funktionen . . . 7

3.1.3 Allgemeine Befehle . . . 7

3.2 Einfache Beispielrechnungen zum Verständnis . . . 8

3.2.1 Absolutbetrag . . . 8

3.2.2 Erweitern, Ausmultiplizieren und Co. . . . 9

3.2.3 Faktorisierung ganzer Zahlen . . . 9

3.2.4 Primzahlen . . . 10

3.2.5 Größter gemeinsamer Teiler und kleinstes gemeinsames Vielfaches . . . 10

3.3 Kurvendiskussion . . . 12

3.3.1 Grenzwerte . . . 12

3.3.2 Differenzieren . . . 13

3.3.3 Integrieren . . . 13

3.4 Vektorrechnung, lineare Algebra . . . 14

3.4.1 Matrizen . . . 14

3.4.2 Vektorrechnung . . . 16

4 Grafische Ausgabe mit MuPAD **17**

4.1 Zweidimensionale Darstellung . . . 17

4.2 Dreidimensionale Darstellung . . . 18

5 Wenn es mal nicht weitergeht... 20

5.1 Fehler in der Berechnung, keine Fehlermeldung 20

5.2 Fehlermeldung bei einem Befehl 20

5.3 Ich komme gleich drauf 21

5.4 Ich verstehe gar nicht, was diese sch... Aufgabe hier von mir will? 21

Kapitel 1

Vorwort

1.1 Stilvereinbarung

Um die folgende Anleitung besser zu verstehen, verwende ich bestimmte Fonteinstellungen. Normaler Text, wird so dargestellt, wie dieser, den Du gerade liest. Die Styles sehen im Wesentlichen wie folgt aus:

email/URL:

> *testuser@l.test.de*

MuPads-Shell:[1]

```
2+3*4;
```

Dieses Dokument bezieht sich auf die MuPAD-Version 1.2.2a und wurde unter Zuhilfenahme von Lyx, TeTex, Ghostscript unter Linux erstellt und muß deshalb gut aussehen. :)

1.2 Warum diese Anleitung?

Ich habe diese Anleitung im Wesentlichen aus folgenden Beweggründen heraus geschrieben. Ich bin Schüler am Leipzig-Kolleg, und mache derzeit mein Abitur. Zur Lösung von Aufgaben kann man MuPAD wunderbar nutzen.

Leider ist es nicht jedem gegeben, englischsprachige Manuals zu verstehen, genausowenig, wie es jedem gegeben ist, aus diesen das wichtige von unwichtigen zu trennen. Und ein ganzes Buch im Umfang von 400 Seiten zu lesen..., was sich eigentlich nur[2]mit der Programmierung von MuPAD beschäftigt, muß nicht sein, zumal es vielleicht für einen Schüler auch etwas zu teuer ist. Mich nervt die kommerzielle Verwertung von elementaren Wissen, was eigentlich frei sein sollte: es ist der pure Wahnsinn, was ein Schüler so alles zu Hause haben muß, um einigermaßen arbeiten zu können. Ein gutes Schulbuch und Nachschlagewerk kosten zwischen 35,- und 65,-DM, wenn man dann noch Software für Mathe, Physik und Co. kauft, und die Kosten der Internetrecherchen berücksichtigt...[3] Ein weiterer Grund ist, daß heutzutage ein Matheass in der gymnasialen Oberstufe nicht mehr ausreicht[4], und es vernünftige Software neben MuPAD im kostenlosen/unteren Preisbereich für Schüler nicht mehr gibt. Also, kurz gesagt, warum sollte ich keine Anleitung zu MuPAD speziell für Gymnasiasten und Erstsemestler schreiben? Ein weiterer Grund ist, daß ich für ein paar Freunde[5] und für mich eben genau so eine kurze Anleitung brauchte, es aber keine gab und ich mich dann einfach mal hingesetzt habe. Und last but not least wollte ich mal lyx[6] ausprobieren.

[1] Ein- und Ausgabe
[2] in Anführungsstrichen!
[3] soll ich jetzt eigentlich ein Spendenkonto einrichten?
[4] zumindest in Sachsen
[5] Ein herzlicher Gruß sei an dieser Stelle Maren Mielke gewidmet: Bleib so wie Du bist: Dir ehrlich und treu, und ganz wichtig, denk an dein Studium :)
[6] lyx ist ein whatyouseeiswhatyoumean-editor, mit dem man einwandfreie TEX-Dokumente schreiben kann

Kapitel 2

Vorbereitung

2.1 Warum MuPAD?

MuPAD ist frei zur nichtkommerziellen[1] Verwendung, nähere Details findet man im File „Licence“ in der MuPAD-Distribution. Desweiteren kann MuPAD einem viel Arbeit in der Mathematik abnehmen und erleichtert das Lösen und Überprüfen von mathematischen Problemen. Dabei rechnet MuPAD weitgehend symbolisch, das heißt, es rechnet so, wie man es in der Schule lernt[2] und ist durch die Möglichkeiten der grafischen Ausgabe universeller als so manche teueren WinMathe-Programme[3]. MuPAD ist für viele Plattformen verfügbar und läuft unter Windows und Macintosh genauso, wie unter Linux.

2.2 Woher bekommt man MuPAD?

MuPAD kann man zB. aus dem Internet beziehen:

ftp://ftp.uni-paderborn.de/local/MuPAD

oder per eMail über:

MuPAD-distribution@uni-paderborn.de

oder per Brief über:

> Benno Fuchssteiner
> MathPAD-Group Automath
> Institut für Automatisierung und Instrumentelle Mathematik
> Universität Paderborn
> Warburger Straße 100
> D-33095 Paderborn
> Deutschland

2.3 Wie wird MuPAD installiert?

Dies ist von Betriebssystem zu Betriebssystem unterschiedlich. MuPAD benötigt unter allen Betriebssystemen zwischen 13 und 50 Mb freien Festplattenplatz. Empfehlenswert ist ein 486-Rechner (oder höher) mit viel Speicher (Minimum 16 MB)[4], besonders, wenn man extensiv das Grafikpaket oder die Millionste Stelle von π berechnen will.

---

[1] man sollte sich aber registrieren lassen, sh. „Registration“ der MuPAD-Distribution
[2] lernen sollte, ähem, gelehrt bekommen sollte
[3] „Winfunktion Mathematik“ kostet knapp 70 DM, „Mathematica“ ca. 340 DM
[4] „je mehr, desto besser“ trifft auch hier zu, notfalls wird geswappt :)

2.3.1 Linux

An Softwarevoraussetzung unter Linux ist eine „libc“ und „libm“ Version 5 nötig (mupad), sowie libc5-gelinkte xview-, olgx-, x11-, xext-Libraries (xmupad). Notfalls ist es erforderlich bei glibc-basierten Distributionen (Suse6.0, Debian 2.0/2.1) die oldlibs-Packages nachzuinstallieren.

Unter Linux erhält man ein "*.tar.gz", "*.tgz", "*.tar.bz2"[5] oder "*.zip"[6] Archiv. In diesen Archiven gibt es meist einen Installscript, wenn nicht, dann sollte man nach einer „INSTALL“ Datei ausschauhalten und sich diese anschauen.

Bei der Installation ist noch zu beachten, daß die Umgebungsvariablen und Pfade stimmen. Unter Linux könnte ein Eintrag in „/etc/profile“[7] so aussehen:

```
# /etc/profile: system-wide .profile file for bash(1).

PATH="/usr/local/bin:/usr/bin:/bin:/usr/bin/X11:/usr/games"
PS1="\\$ "

export PATH PS1

umask 002
/usr/bin/check-sendfile

#MuPAD 1999 by Andreas added
MuPAD_ROOT_PATH="/usr/local/MuPAD"
export MuPAD_ROOT_PATH
FONT_PATH=$MuPAD_ROOT_PATH/share/doc/hytex/fonts
MANPATH=‘manpath‘
export MANPATH
if test -n "$MANPATH" ; then
MANPATH=$MANPATH:$MuPAD_ROOT_PATH/share/doc/man
else
MANPATH=$MANPATH:$MuPAD_ROOT_PATH/share/doc/man
fi
export MANPATH
PATH=$PATH:$MuPAD_ROOT_PATH/share/bin
export PATH
export FONT_PATH
#end MuPAD
```

2.3.2 Windows und Macintosh

Erfahrungen zur Windows- oder Macintoshversion bezüglich der Installation liegen mir nicht vor[8]. Im MuPAD-FAQ steht aber, daß für Windows die Dateien „winmup1.exe“ und „winmup2.exe“ als selbstextrahierende Archive zur Verfügung stehen. Die Installation unter Windows9x ist unproblematisch[9]. Man legt ein Verzeichnis an, zB. „c:\winmupad“ und kopiert dahinein diese zwei Archive. Dann sollte man mit „winmup1.exe /PATH“ bzw. „winmup2.exe /PATH“ MuPAD auspacken können. Für Windows 3.x ist es nötig, den 32-bit Loader WIN32S zu installieren.

Für weitere Information, insbesondere für die Installation unter Windows und Macintosh lest die entsprechenden Dateien.

---

[5] "*.tar.gz" bzw. "*.tgz" packt man mit "tar xfz file.tar.gz" aus, "*.tar.bz2" ist ein bzip2- gepacktes Archiv, man kann es mit "cat file.tar.bz2 | bunzip2 | tar xf" auspacken

[6] Naja, da sollte wohl klar sein, wie man ein File auspackt? Wenn nicht, gib unter DOS/Windows mal "pkunzip /?" unter Unix "man unzip" ein. :)

[7] gültig für bash, für andere Shells sind die Angaben entsprechend anzupassen

[8] Vielleicht könnt Ihr mir mal da weiterhelfen...

[9] Hüstel!

Kapitel 3

Tutorial

3.1 Einfach loslegen

Um sich am schnellsten mit MuPAD vertraut zu machen, hilft nur eines: einfach loslegen.

Das machen wir jetzt auch, wir starten das Programm, unter Linux[1] mit „mupad“ unter der shell, bzw. „xmupad“ unter XWindows. Unter Windows und Macintosh sollte das Programm per Doppelclick zu starten sein. Jetzt erscheint, wenn das Programm richtig installiert ist, das MuPAD-Logo mit einem Prompt[2]. Um mal ganz einfach 10+3*2 zu rechnen gibt man das einfach so ein „10 + 3 * 2;“[3], drückt <RETURN> und als Ergebnis erscheint, richtig, „16“. Was man also eingibt, wird sofort berechnet und vereinfacht (ua. werden auch Brüche und Potenzen gekürzt, wenn man also „32/11 + 1/11;“ eingibt erscheint als Ergebnis nicht „33/11“ sondern „3“). MuPAD kann auch mit Variablen rechnen. Wenn man zB. „$\frac{a+b}{a*b}$“ rechnen will, gibt man das so ein, wie man es auch auf einem normalen Taschenrechner mit Zahlen rechnen würde: „(a+b)/(a*b);“. MuPAD versucht diesen Ausdruck so darzustellen, daß die Ausgabe der mathematischen Schreibweise weitgehend ähnelt[4]:

```
a + b
-----
a * b
```

Bei der Eingabe von Zahlen ist zu beachten, daß MuPAD als Komma einer Zahl nur den Punkt akzeptiert, aus $2,34$muß man bei der Eingabe in Mupad:

```
2.34
```

machen. Bei der Eingabe von mathematischen Ausdrücken, wie $2x$ ist zu beachten, daß man unter MuPAD den Multiplikator wieder einfügt, also

```
2*x
```

schreibt.[5]

[1] oder irgendeinem anderem Unix, ich verwende hier trotzdem Linux, weil sich das Schüler wohl gerade noch leisten können (naja, BSD kann sich so ein armer Schlucker vielleicht auch noch leisten, aber wer hat schon ein SCO-Unix auf seinen Heimrechner am Laufen?)

[2] Ein Prompt ist das, was man sieht, wenn man unter Linux in der Shell und unter Windows im MSDOS-Fenster arbeitet. Bei MuPAD sieht der Prompt so „>>“ aus

[3] Achtung! Jeder Ausdruck der berechnet werden soll muß mit einem Semikolon abgeschlossen werden!

[4] Dieser Modus nennt sich Pretty_Print, soll er abgestellt werden so gibt man unter MuPAD „PRETTY_PRINT:=FALSE;“, andernfalls „PRETTY_PRINT:=TRUE;“ ein

[5] Ich weiß nicht, warum das die Mathematiker aus Paderborn so gemacht haben. Jeder Taschenrechner kann das mittlerweile :(Weil wir gerade dabei sind, nervig ist auch, daß unter Linux die Backspace-Taste die Funktion der Del-Taste hat, und die Del-Taste nur Mist produziert. Wenn es nur ein Fehler bei mir ist, dann habe ich leider noch keine Möglichkeit gefunden, dies umzuändern...

3.1.1 Mathematische Konstanten

MuPAD	Symbol	Bemerkung
PI	π	3,1415...
E	e	2,7182...
I	i	imaginäre Einheit

3.1.2 Allgemeine arithmetische Funktionen

Funktion	Mathem.	Beschreibung	Bemerkung	Beispiel
+	$+$	Addition		12+24
-	$-$	Subtraktion		12-24
*	$\star$	Multiplikation		12*24
/	$\div$	Division		12/24
ˆ	a^b	Potenzierung		12ˆ24
sqrt()	$\sqrt{a}$	Quadratwurzel	rechnet weitgehend symb.	sqrt(9)
ln()	$\ln a$	natürlicher Logarithmus		ln(3.0)
exp()	$\exp a$	Exponentialfunktion		exp(3.0)
sin()	$\sin\alpha$	Sinus	Achtung! Angabe in RAD	sin(0.2)
cos()	$\cos\alpha$	Cosinus	Achtung! Angabe in RAD	cos(PI)
tan()	$\tan\alpha$	Tangens	Achtung! Angabe in RAD	tan(2.1)
asin()	$\sin^{-1} x$	Arcussinus	Ausgabe in RAD	asin(1)
acos()	$\cos^{-1} x$	Arcuscosinus	Ausgabe in RAD	acos(0)
atan()	$\tan^{-1} x$	Arcustangens	Ausgabe in RAD	atan(-2.44)
sinh()	$\sinh x$	Sinus hyperbolicus		sinh(E)
asinh()	$\sinh^{-1} x$	Areasinus		asinh(E)
random()		gleichvert., ganzz. Zufallszahl	$0...2^{15}-1$, kein Parameter	
sign()		Vorzeichen	gibt -1, 1 zurück	sign(-5)
abs()	$\lvert a\rvert$	Absolutbetrag		abs(-5)
max()		Maximum		max(-2, 3, 0)
min()		Minimum		min(-2,3,0)
fact()	$a!$	Fakultät		fact(24)
round()		Rundung	nach 0,5-Regel	round(24.3)
ceil()		Aufrundung		ceil(24.3)
floor()		Abrundung		floor(24.3)
trunc()		Ganzzahlanteil	analog zu INT in Pascal	trunc(24.3)
float()		Umwandlung in Dezimalzahl	macht aus $\frac{2}{3}$0,666...	float(2/3)
div		Ganzzahlige Division		2 div 3
mod	$a\%b$	Rest bei ganzz. Division		2 mod 3
frac()		Rest		frac(24/23)

3.1.3 Allgemeine Befehle

MuPAD hat bezüglich der Genauigkeit, der History, usw. im allgemeinen gute und nützliche Voreinstellungen. Man kann diese aber über die von MuPAD zur Verfügung gestellten Variablen ändern. Da Variablen bei der Nutzung von MuPAD eine große Rolle spielen, erwähnen wir die Zuweisung von Variablen zuerst. Im allgemeinen hält sich das Programm an die Syntax von PASCAL, dh. wenn ich der Variablen a den Wert $\frac{2}{3}$ zuweisen will, dann sieht das so aus:

```
a:=(2/3);
```

Warum ist das so nützlich? Nun, wir können Variablen als Zwischenspeicher für Ergebnisse, oder aber auch für Formeln benutzen. So kann man zB. folgendes machen:

```
a:=2*x^2;
x:=5;
5*a;
3*a;
```

Hier haben wir also den Term $2x^2$ durch a ersetzt[6], und können diesen Term dann in späteren Rechnungen, wie im obigen Beispiel nutzen. Die Variablenzuweisung hatten wir ja schon kennengelernt. Wenn man zB. die HISTORY-Tiefe (auch als UNDO bekannt) verändern will[7] , so gibt man in der MuPAD-Shell zB. folgendes ein:

```
HISTORY:=[30,10];
```

Hier wird dann der UNDO-Wert auf 30 (interaktiv) beziehungsweise 10 Schritte (innerhalb von Programmen) gesetzt.

Die Genauigkeit von MuPAD beträgt voreingestellt 10 Stellen, will man dies ändern setzt man die Variable DIGITS auf einen neuen Wert:

```
DIGITS:=50;
```

Zu den allgemeinen Befehlen zählen neben der Variablenzuweisung die folgenden Befehle:

Funktion	Beschreibung	Bemerkung	Beispiel
reset()	setzt alles auf Default	Parameterfrei	reset()
help()	Gib Hilfe zu Befehl xy aus		help(„diff")
?...	Gib Hilfe zu Befehl xy aus		?diff
last()	Liefert den n-ten Wert der Historytabelle		last(2)
%	Liefert den n-ten Wert der Historytabelle		%2
$ x	Wiederholt x-Mal		diff(1/x, x $ 8)
history()	gibt Inhalt der Historytabelle aus	Parameterfrei	history()

Der Befehl „help()" bzw. „?..." ist einer der elementarsten. Wenn man mal vor einem Problem steht, wofür nichteinmal hier :) eine Lösung steht, so hilft ein Blick in die englischsprachige Hilfe oftmals weiter.

3.2 Einfache Beispielrechnungen zum Verständnis

3.2.1 Absolutbetrag

Berechne: $|2 \pm 3|$

Eingabe in MuPAD:

```
abs(2+3); abs(2-3);
```

Lösung:

```
5
1
```

Berechne: $|a + a|$

Eingabe in MuPAD:

```
abs(a+a);
```

Lösung:

```
abs(a*2)
```

[6] oder „substituiert" für all diejenigen, die protzen wollen :^)

[7] voreingestellt sind 20 Schritte, bei Programmen 3 Schritte

3.2.2 Erweitern, Ausmultiplizieren und Co.[8]

Multipliziere aus: $(x+1)*(y+z)$

Eingabe in MuPAD:

```
expand((x+1)*(y+z));
```

Lösung:

```
y+z+x*y+x*z
```

Erweitere mit $(x+z)$: $\frac{x+2z-1}{x+3z}$

Eingabe in MuPAD:

```
expand((x+2*z-1)/(x+3*z), (x+z));
```

Lösung:

```
    1          x         2 z
--------- + ------- + -------
- x - r z   x + 3 z   x + 3 z
```

Gib die binomische Form an: $(2+x+y)^3$

Eingabe in MuPAD:

```
expand((2+x+y)^3);
```

Lösung:

```
               2   3    2   3     2    2
12x+12y+12xy+6x +x +6y +y +3xy +3x  y+8
```

3.2.3 Faktorisierung ganzer Zahlen

Zerlege die Zahl 51 in seine Primfaktoren!

Eingabe in MuPAD:

```
ifactor(51);
```

Lösung:

```
[1, 3, 1, 17, 1]
```

Dabei bedeutet die erste Zahl das Vorzeichen[9], die zweite der erste Primfaktor, die dritte, der dazugehörige Exponent, usw. Ich fasse nochmal zusammen:

ifactor() liefert die Faktorisierung einer ganzen Zahl in der Form [s, p1, e1, ..., pn, en] zurück, wobei s das Vorzeichen angibt, p1 der 1-te Primfaktor, e1 der dazugehörende Exponent.

[8] Der Befehl „expand" ist nicht nur zum Ausmultiplizieren nützlich, sondern erweitert einen Ausdruck, sh. auch die nächsten Beispiele

[9] Wir erinnern uns an den Befehl sign(), also „1" für postives, „-1" für negatives Vorzeichen

3.2.4 Primzahlen

Mit dem Kommando „isprime(x)“ kann man testen, ob x ein Primzahl ist, wenn ja, wird ein „TRUE“, wenn nein ein „FALSE“ zurückgeliefert.

Mit dem Kommando „ithprime(x)“ kann man sich die von MuPAD in einer internen Tabelle gespeicherten x-te Primzahl ausgeben lassen. Wenn die gesuchte Primzahl nicht in dieser tabelle ist, wird ein Fehler zurückgeliefert.

Das Kommando „nextprime(x)“ gibt die kleinste Primzahl aus, die größer oder gleich $x$ist.

3.2.5 Größter gemeinsamer Teiler und kleinstes gemeinsames Vielfaches

Gib den ggT der Zahlen 184 und 26220 an!

Eingabe in MuPAD:

```
igcd(184, 26220);
```

Lösung:

```
92
```

Sei d der ggT der Zahlen $a = 184$ und $b = 26220$. Berechne d für das gilt: $d = ax + by$

Eingabe in MuPAD:

```
igcdex(184, 26220);
```

Lösung:

```
92, -142, 1
```

igcdex(a, b) liefert den Ausdruck d, x, y zurück, hier also $d = 92$, $x = -142$, $y = 1$

Berechne das kgV der Zahlen 5231, 2537, 3648, 10903!

Eingabe in MuPAD:

```
ilcm(5231,2537,3648,10903);
```

Lösung:

```
527844534408768
```

Bestimme die Anzahl der positiven ganzzahligen Teiler der Zahl 12!

Eingabe in MuPAD:

```
numdivisors(12);
```

Lösung:

```
6
```

Bestimme die Anzahl der Primteiler der Zahl 12!

Eingabe in MuPAD:

```
primedivisors(12);
```

Lösung:

```
2
```

Bestimme die Anzahl der teilerfremden Zahlen von 61!

Eingabe in MuPAD:

```
phi(62);
```

Lösung:

```
30
```

Die eulersche Funktion ϕ wird in MuPAD also mit dem Befehl „phi(x)“ aufgerufen

Lösen von Gleichungen

Ja, ja! Es klingt einfach, ist es auch, mit MuPAD! Auf geht´s...

Berechne die Nullstellen der Funktion $f(x) = y = x^2 - 3x + 3$!

Eingabe in MuPAD:

```
solve(x^2-3x+3, x);
```

Lösung:

```
           1/2                    1/2
[1/2 I 3       + 3/2, (-1/2 I) 3      + 3/2]
```

Hmmh. Wie soll man das wohl interpretieren? Auch nicht schwer, ich schreibe das Ergebnis nochmal der Übersichtlichkeit wegen in mathematischer Symbolschreibweise:

$L = [\frac{1}{2}i * 3^{\frac{1}{2}} + \frac{3}{2}, (-\frac{1}{2}i) * 3^{\frac{1}{2}} + \frac{3}{2}]$ bzw. $L = [\frac{1}{2}i * \sqrt{3} + \frac{3}{2}, (-\frac{1}{2}i) * \sqrt{3} + \frac{3}{2}]$

Wir sehen, daß die Lösung nur mit komplexen Zahlen möglich ist, und sind total erstaunt, was MuPAD alles kann, gelle? Ok. hier gibt es nochmal eine Zusammenfassung zum „solve()“-Befehl:

Funktion	Beschreibung	Bemerkung	Beispiel
solve($f(x), x$)	löst Gleichungen nach x auf		sh. oben
linsolve($\{f(x), g(y), h(z)\}$,{x, y,z})	löst lineare Gleichungen auf	sonst solve()!	

Ok, noch eine Aufgabe:

Bestimme x, y, z!

$2 = 3x - 2$

$4 = 2 - 4y - 3x + z$

$3 = z + x$

Eingabe in MuPAD:

```
linsolve({2=3*x-2, 4=2-4*y-3*x+z, 3=z+x},{x,y,z});
```

Lösung:

```
{y=-13/12, x=4/3, z=5/3}
```

3.3 Kurvendiskussion

Nullstellenbestimmung haben wir ja jetzt schon kennengelernt. Jetzt geht es langsam auf´s Abiturniveau zu. Am Besten, wir versuchen mal so eine typische Prüfungsaufgabe zu lösen, weil man daran am ehesten sieht, worauf es ankommt:[10]Untersuche die Funktion $ft(x) = \frac{4}{x} - \frac{4t}{x^2}$, $t > 0$

Nullstellen

zuerst müssen wir die Gleichung $0 = \frac{4}{x} - \frac{4t}{x^2}$ nach t umstellen, da MuPAD kein Ergebnis bei folgender Eingabe liefert:

solve((4*x-4*t)/(x*x),x);

das macht aber nichts, wie wir gleich sehen werden, also rechnen wir $\mid +(\frac{4t}{x^2}) \mid * x^2 \mid \div 4$, so daß wir $t = x$ erhalten

Polstellen

ist auch einfach zu berechnen, da $t > 0$ ergibt sich die Polstelle für x=0

3.3.1 Grenzwerte

Untersuchen wir das Verhalten der Funktion im Unendlichen:

```
limit((4*x-4*t)/(x*x),x=infinity);
```

Lösung:

```
0
```

Untersuchen wir das Verhalten der Funktion wenn x gegen 0 läuft,

von links:

```
limit((4*x-4*t)/(x*x),x=0, Left);
```

von rechts:

```
limit((4*x-4*t)/(x*x),x=0, Right);
```

Herauskommen müßte $-\infty$. Da MuPAD den Befehl „limit()“ über den „series()“ realisiert, kommt es in diesem Beispiel zu einem Fehler, da „sign()“ den Wert von t nicht ermitteln kann, hier hilft einfach die Zuweisung von $t = 1$[11], jetzt spuckt MuPAD die Lösung aus:

```
-infinity
```

[10] bei einer Prüfung kommt es auf was ganz anderes an, zB. ausreichend Schlaf, Sex, Frühstück bei McDoof... Ja, in der Reihenfolge ;-)

[11] da t eine Konstante ist, ist dies kein Problem... :)

Extrema

3.3.2 Differenzieren

Wir benötigen die 1. und 2. Ableitung, also MuPAD bemüht:

```
diff(4/x-4*t/(x*x),x);
```

Lösung:

```
    4     8t
- --- + ----
     2     3
    x     x
```

so daß gilt: $ft'(x) = -\frac{4}{x^2} + \frac{8t}{x^3}$

Eingabe zur Berechnung der 2. Ableitung:

diff(-4/(x*x)+8*t/(x^3), x);

Lösung:

```
8     24 t
-- - ----
 3     4
x     x
```

so daß gilt: $ft''(x) = \frac{8}{x^3} - \frac{24t}{x^4}$

Jetzt setzen wir $ft'(x) = 0$, lösen dies mit „solve()", setzen den Wert in $ft''(x)$ für x ein, lösen mit „solve()" auf, und wissen ob wir ein Minimum (ft''>0) oder Maximum (ft''<0) haben.

Für die Berechnung der Wendestelle gehen wir genauso vor. Mit „diff()" berechnen wir dann die 3. Ableitung.

Die Funktion „diff($f(x), x$)" differenziert die Funktion $f(x)$ nach x. Will man beispielsweise die achte Ableitung von $f(x)$berechnen, so gibt man dies mit dem Sequenzierungsoperator ein:

diff((1/x),x $ 8);[12]

Flächenberechnung

3.3.3 Integrieren

Berechnen sie die Fläche der Funktion $f(x) = \frac{x-2}{\sqrt{(x^2+4x+4)}}$ in den Grenzen $\{0; 0,5\}$:

Das entspricht also dem Integral der Funktion: $\int f(x)dx = 0; 0,5 \int \frac{x-2}{\sqrt{(x^2+4x+4)}}dx$

Eingabe unter MuPAD:

int((x-2)/sqrt(x*x+4*x+4),x)[13];

ergibt keine Lösung, da es MuPAD nicht schafft den Ausdruck $(x^2 + 4x + 4)$ als $(x + 2)^2$ aufzulösen, und nicht sieht, daß sich die Wurzel dann mit dem Quadrat aufhebt. Wir kommen deswegen nicht auf ein Ergebnis, weil MuPAD dann den Wald vor lauter Bäumen nicht bemerkt, und seine Regeln für symbolische Integration dann versagen. Wollten wir also die Stammfunktion zu obiger Formel haben, so müssen wir per Hand nachhelfen. Und wenn man jetzt diese Funktion eingibt:

int((x-2)/(x+2), x)

---

[12] wen es interessiert: $f(x) = \frac{1}{x} \rightarrow f^8(x) = \frac{40320}{x^9}$

[13] unbestimmtes Integral

so erhält man: $\int \frac{x-2}{x+2}dx = x - 4\ln(x+2) + C$

Man sieht, das MuPAD zwar viel kann, eines aber nicht, die Denkarbeit abnehmen. Um aber trotzdem zu einer Lösung zu kommen (ohne das man nachdenken muß) zwingen wir MuPAD numerisch zu integrieren, und zwar so:

```
float(int((x-2)/sqrt(x*x+4*x+4),x=0..0.5));
```

und siehe da, als Ergebnis bekommen wir:

```
-0,3925...
```

Da Flächeninhalt immer positiv: 0,3925...

Ok, überprüfen wir das, denn wir sind jetzt mißtrauisch geworden. Setzen wir in obiges Integral einmal 0 ein und einmal 0,5 und wir ziehen beides voneinander ab.

in MuPAD:

```
x:=0;x-4*ln(x+2);x:=0.5;x-4*ln(x+2);
last(3)-last(1);
```
[14]

Ergebnis:

```
0,3925...
```

3.4 Vektorrechnung, lineare Algebra

MuPAD kann man auch für dieses Teilgebiet der Mathematik einsetzen, obwohl stark eingeschränkt. Eingeschränkt deshalb, weil MuPAD bezüglich der grafischen Darstellung nicht nutzbar ist, und gerade die elementar für das Verständnis dieses Teilbereiches der Mathematik ist. Gut, man könnte mit dem „plot3d“-Kommando versuchen, Ebenen, Kugeln darzustellen[15]. Befriedigend ist das nicht.

Jetzt wird es kompliziert, denn es sind ein paar Vorbereitungen nötig, da MuPAD im Grundzustand, also nach Programmstart oder dem „reset()“-Befehl, nichts über Matrizen weiß. Zuerst muß MuPAD also mitgeteilt werden, daß es uns bei der Vektorrechnung helfen soll:

```
loadlib(„linalg“);
export(linalg);
```

Es wird also die Bibliothek „linalg“ geladen und MuPAD bekanntgemacht. Die Ausgaben auf dem Bildschirm[16]können ignoriert werden. Wichtig dagegen ist aber die unbedingte Angabe der Gänsefüßchen, da sonst MuPAD abstürzt[17]und die History hin ist.

3.4.1 Matrizen

Wenn wir also eine Matrix auf unseren Schirm bringen wollen, müssen wir zuerst einen Variablentyp Matrix deklarieren:

```
MI:=Matrix(Integer);
```

definiert den Ganzzahlen-Matrixtyp MI,

[14]last(x), wir erinnern uns, holt den x-letzten Wert aus der History
[15]indem man die Ebenengleichungen als Parameter einsetzt, bzw. Graphische Primitive nutzt
[16]hier nicht dargestellt
[17]Harhar :)

```
MR:=Matrix(Rational);
```

definiert den rationalen Matrixtyp MR.

Nun können wir eine Matrix[18]herstellen:

```
A:=MR([[1,2,3],[4,5,6],[7,8,9]]);
```

Diese Matrix sieht nun so aus: $A = \begin{bmatrix} 1 & 2 & 3 \\ 4 & 5 & 6 \\ 7 & 8 & 9 \end{bmatrix}$

Wir können aber die Typdefinition MuPAD überlassen, für Vektoren ist dies sogar notwendig. Das Ganze wird dann einfacher. Für obiges Beispiel kann der Aufruf nun auch so aussehen:

```
A:=Matrix()([[1,2,3],[4,5,6],[7,8,9]]);
```

für einen Vektor $\overrightarrow{b} = \begin{bmatrix} 1 \\ 2 \\ 3 \end{bmatrix}$ schreibt man:

```
b:=Matrix()([[1,2,3]]);
```

Ok. Wir wissen nun, wie man Matrizen und Vektoren definiert. Mit dem Kommando „type(A);“ können wir jederzeit den Matrixtyp der Matrix A erfahren.

Jetzt können wir die Standardfunktionen von MuPAD auf diese Matrix anwenden:

$A = \begin{bmatrix} 1 & 2 \\ 0 & -3 \end{bmatrix}$; $B = A * 3$; $C = A - B$; $D = A \times C$

Eingabe in MuPAD:

```
A:=mq([[1,2],[0,-3]]);
```

Ausgabe von MUPAD:

```
+-      -+
| 1,  2 |
|        |
| 0, -3 |
+-      -+
```

Eingabe in MuPAD:

```
B:=3*A;
```

Ausgabe von MuPAD:

```
+-      -+
| 3,  6 |
|        |
| 0, -9 |
+-      -+
```

Eingabe in MuPAD:

[18] oder mehrere

```
C:=A-B;
```

Ausgabe von MuPAD:

```
+-        -+
| -2, -4 |
|          |
|  0,  6 |
+-        -+
```

Eingabe in MuPAD:

```
D:=A*C;
```

Ausgabe von MuPAD:

```
+-         -+
| -2,  8   |
|           |
|  0, -18 |
+-         -+
```

Hier eine kurze Übersicht über die Befehle zur Matrizenmanipulation:

Funktion	Beschreibung	Beispiel
det()	Ermittelt die Determinante	nur für quadrat. Matrix
dimen()	Dimension der Matrix	
eigenvalues()	Eigenwert der Matrix	
eigenvectors()	Eigenwerte u. -vektoren	
gaussElim()	Gaussch. Elimin.	
gaussJordan()	Gauss-Jordan Elim.	
grad()	Grad der Matrix	
jordanform()	Jordanform	
linearSolve()	löst Matrix A nach Boder $\overrightarrow{b}$	A, $\overrightarrow{b}$: „linearSolve(A, b);“
rank()	Rang der Matrix	
sylvester()	Sylvesterform	
transpose()	vertauscht Zeilen und Spalten	quadrat. Matrix

Da die Befehle zur Matrizenmanipulation sich in einigen MuPAD-Versionen unterscheiden, sollte man unbedingt „?linalg“ eingeben und sich mit der genauen Syntax bekannt machen.

3.4.2 Vektorrechnung

Vektoren werden prinzipiell genauso wie Matrizen definiert[19], der Vektor $\overrightarrow{b} = \begin{bmatrix} 1 \\ 2 \\ 3 \end{bmatrix}$ also so:

```
b:=Matrix()([[1,2,3]]);
```

Folgende Befehle können für die Vektorrechnung verwendet werden:

Funktion	Beschreibung	Bemerkung
angle($\overrightarrow{a}$, $\overrightarrow{b}$)	Gibt Winkel zurück	$\overrightarrow{a}$, $\overrightarrow{b}$ müssen gleiche Dimension haben
normalize()	bestimmt Normalvektor	
scalarProducts($\overrightarrow{a}$, $\overrightarrow{b}$)	Skalarprodukt	

[19]sh. ein paar Zeilen weiter oben

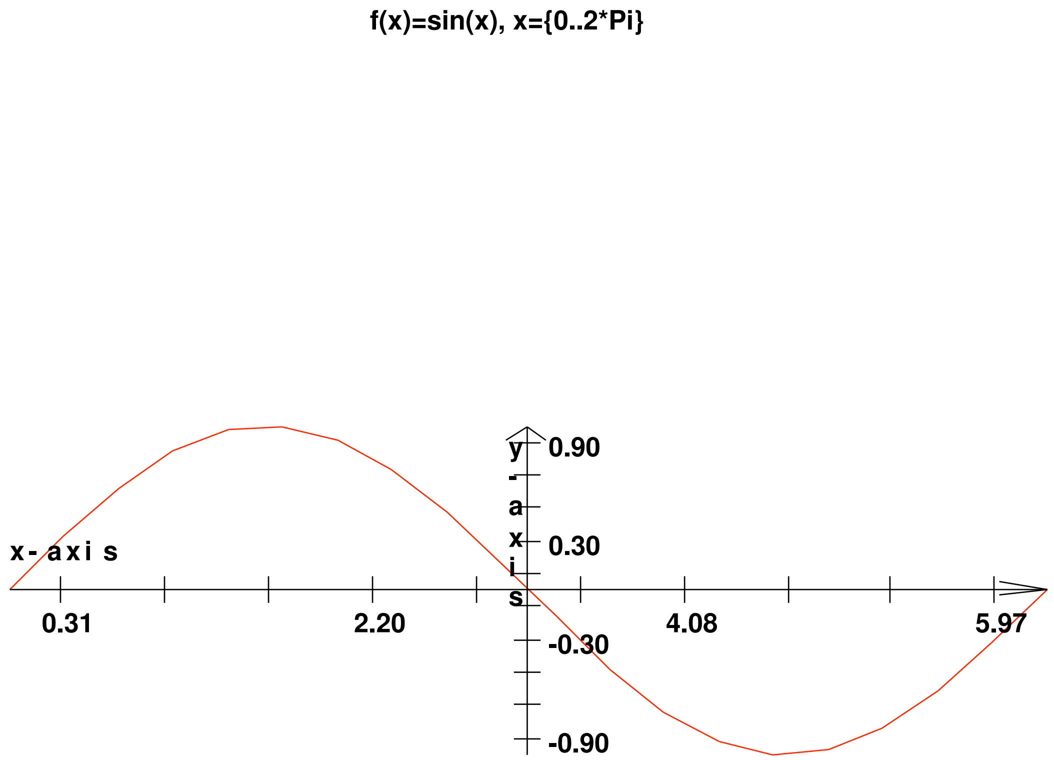
Kapitel 4

Grafische Ausgabe mit MuPAD

MuPAD kann zweidimensionale und dreidimensionale Grafiken erzeugen. MuPAD berechnet die Daten und schickt sie, wenn möglich[1] an ein Interface, wo diese Daten angezeigt, untersucht und interaktiv verändert werden können.

4.1 Zweidimensionale Darstellung

Ein einfaches Beispiel:



wird übrigens so erzeugt:

```
plot2d([Mode=Curve, [x, sin(x)], x=[0,2*PI]]);
```

[1] Wenn MuPAD im DOS-Modus, oder auf Shell-Ebene gestartet ist, wird eine Datei „save.mp“ erzeugt, leider bin ich noch nicht dahintergekommen, wie man die Art der Datei beeinflußt, MuPAD kann ja in VCAM den Plot in GIF , PS ua. Grafikformate abspeichern. Wenn das jemand, weiß, ich freue mich auf jede Antwort...

Der Befehl „plot2d()“ hat folgende Syntax:

plot2d(Optionen, [Objekt, Optionen]...)

Nein, keine Angst, das ganze wird schnell klar, wenn uns obiges Beispiel anschauen. Ich muß zugeben, ich habe ein wenig geflunkert, der korrekte Aufruf zum Erzeugen obiger Grafik lautet wie folgt:

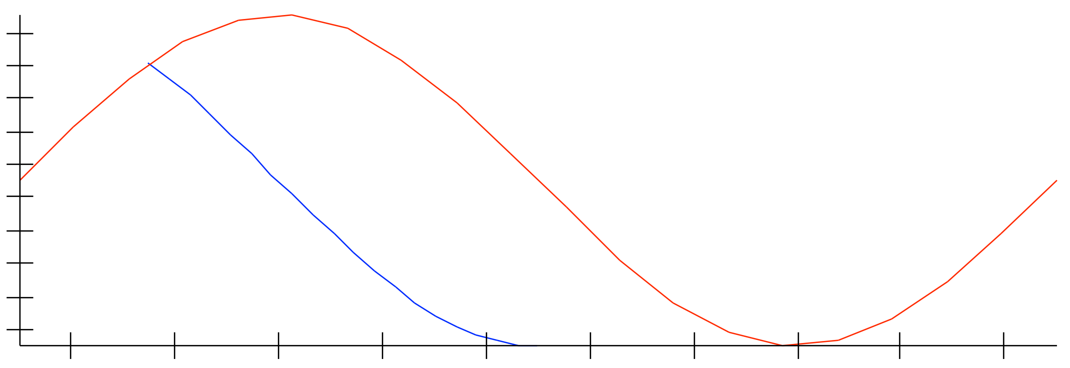
```
plot2d(Title=“f(x)=sin(x), x={0..2*Pi}“, Axes=NormalAxes, Labels=[„x-axis“, „y-axis“], [Mode=Curve, [x, sin(x)], x=[0,2*PI]]);
```

Mit den Optionen ist im Prinzip egal, da man alle Optionen im VCAM-Modul[2] verändern interaktiv kann, und deshalb so gut wie nichts darüber wissen muß. Wen es doch interessiert, der sollte

```
?plot2d;
```

eingeben. Was man vielleicht noch wissen sollte, ist, daß man auch mehrere Plots gleichzeitig in „plot2d()“ eingeben kann:

```
plot2d([Mode=Curve,[u, sin(u)],u=[0,2*PI]], [Mode=Curve, [v,cos(v)],v=[PI/4,PI]]);
```



Soviel zum zweidimensionalen Plotten.

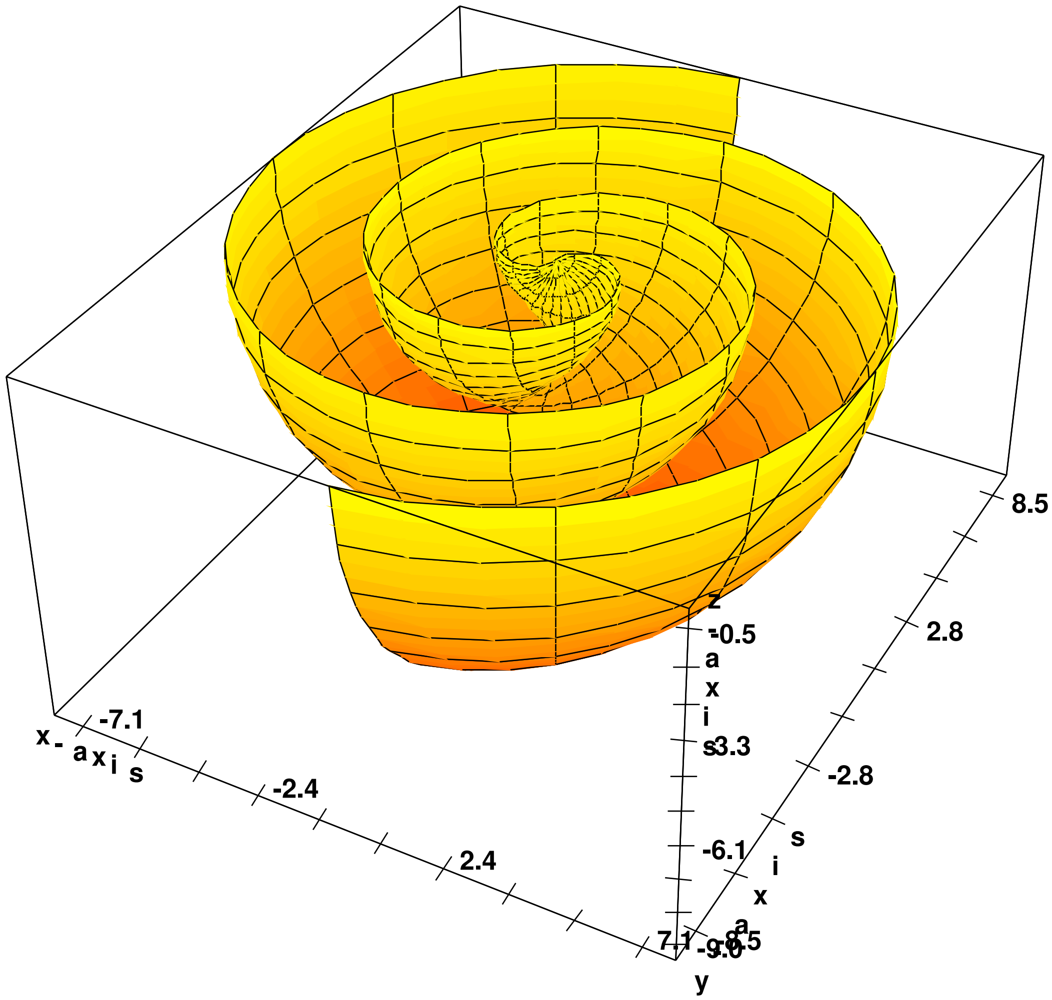
4.2 Dreidimensionale Darstellung

Der Befehl hierzu heißt analog zum „plot2d“ wie? Ja, „plot3d()“ und wird fast genauso aufgerufen.

plot3d([Mode=Surface,[u*cos(v)*sin(u), u*cos(u)*cos(v),-u*sin(v)], u=[0,3*PI], v=[0,PI]]);

erzeugt[3] dieses Bild:

[2] Das Plotprogramm von MuPAD, wird aus „xmupad“, „winmupad“ und „macmupad“ heraus aufgerufen.
[3] und einige kleine Änderungen im VCAM-Modul :)



-7.1
x-axis
-2.4
2.4
7.1
8.5
2.8
-2.8
-8.5
y-axis
z-axis
-0.5
-3.3
-6.1
-9.0

Kapitel 5

Wenn es mal nicht weitergeht...

Nicht verzagen, MuPAD fragen. Nein, im Ernst. Wenn dieses Dokument Dir keine Hilfe mehr ist, dann hilft es Dir vielleicht doch mit dieser kleinen TopTen der Informationsbeschaffung.

Es gibt verschiedene Ursachen, wenn man mal nicht weiter weiß. Entweder fällt einem ein bestimmter Befehl nicht ein, oder man weiß nicht, wie man dies und jenes berechnet. Oder es kommt zu einer Fehlermeldung. Es kann auch sein, daß man überhaupt nicht weiß worin ein Problem besteht, weil man es nicht versteht. Gut, genug gelabert, fangen wir an:

5.1 Fehler in der Berechnung, keine Fehlermeldung

Dies tritt dann auf, wenn man Sachen berechnen will, aber entweder das mathematische Problem nicht verstanden oder Besonderheiten und Macken von MuPAD nicht berücksichtigt hat. Zum Beispiel hatte MuPAD in älteren Versionen den Cosinus-Pi-Fehler, daß heißt, wenn man $cos\pi$ durch Eingabe von

```
cos(PI);
```

berechnen wollte, kam statt

```
-1
```

die Meldung

```
cos(PI)
```

zurück. Bei solchen Fehlern heißt es Befehle checken, bestimmte Dinge per Hand vereinfachen[1] und eines immer im Hinterkopf behalten: MuPAD ist ein Programm, und genauso fehlbar, wie sein Benutzer[2]

5.2 Fehlermeldung bei einem Befehl

Dies kann zwei Ursachen haben, entweder man hat die Syntax nicht eingehalten, oder ein Befehl innerhalb des Befehles ist defekt, funktioniert nicht, oder weiß der Geier was. Hier hilft nur eines, Hilfe zum Befehl aufrufen und Verschachtelung auflösen, also Befehl für Befehl überprüfen. Wenn ein Syntaxfehler vorliegt, so gibt MuPAD die Zeile aus und versucht die Stelle des Fehlers mit dem Cursor zu markieren. Diese Markierung ist ein guter Anhaltspunkt.

[1] sh. 3.3.2 Auflösen des Binoms vor Integration
[2] und seine Programmierer, und seines Prozessors (-:|>

5.3 Ich komme gleich drauf

Ja, da war doch was? In so einem Fall kann man zwei Dinge machen, zum einen den Befehl „help“[3] eingeben, dieser gibt eine englischsprachige nach Themen geordnete Funktionenliste zurück. Besser ist aber der Befehl ?demo, der eine nach mathematischen Themen geordnete Hilfe mit Beispielen (!) zurückgibt. Den gleichen Effekt erreicht man, wenn man auf das Verzeichnis „../MuPAD/WWW/demo“ zugreift[4] und sich das ganze mit einem Browser anschaut.[5]Vielleicht hilft aber auch schon ein neues Überfliegen dieses Dokumentes, bzw. dessen Indexes :)

5.4 Ich verstehe gar nicht, was diese sch... Aufgabe hier von mir will?

Hmmmh. Dies hat zwar nichts mit MuPAD zu tun, trotzdem einige Tips. Alle mathematischen Aufgaben sind so formuliert, das sie zwar eindeutig sind, aber dies nur für einen Mathematiker. Außerdem sind sie in einer Sprache formuliert, die ein normaler Mensch die Haare zu Berge stehen läßt, für einen Mathematiker aber purstes Zuckerwerk[6] sind. Um es mal anders auszudrücken, hier treffen zwei Welten aufeinander, und das bringt nunmal Probleme. Man kommt aber dahinter, wenn man strikt jedes Wort der Aufgabenstellung nimmt, mit einem Mathematiklexikon[7] vergleicht und so in mühevoller Handarbeit den Sinn auseinanderklamüsert. Wenn man Mathe an sich nicht versteht, so hilft nur eines, besseren Lehrer suchen! Es gibt zwar gute Mathematikbücher, aber im Selbststudium Mathe zu lernen ist ungefähr so, wie wenn ein taubstummer Blinder Chinesisch lernen soll. Das mit der Lehrersuche ist vielleicht nicht immer möglich, man sollte sich dann aber einen Studenten oä. als Zusatzlehrer suchen, kommt allemal billiger als ein Nachhilfekurs. Für die Oberstufe Mathematik kann ich im Prinzip nur zwei Bücher empfehlen:

Ratgeber Mathematik, Verlag Bassermann[8]

und

Taschenbuch der Schulmathematik[9]; Stöcker, Münzel; Verlag Harry Deutsch; Thun und Frankfurt am Main; ISBN 3-8171-1439-7.

Im übrigen kann ich jedem, der mit Mathematik zu tun hat den Taschenrechner „Texas Instruments TI-83“ empfehlen, da dieser alle Befehle hat, die nützlich sind, grafikfähig und programmierbar ist, einen ordentlichen Gleichungs- und Matrizenmodus hat, Differenzieren und Integrieren kann[10], sowie ein Linkkabel zu anderen TIs[11] inklusive hat. Außerdem findet man unter „http://www.ticalc.org“ ein Haufen Programme, Bastel- und Tuninganleitungen.

So, genug der Werbung...

[3]unter Windows/Macintosh/XWindows auf Button Help clicken

[4]sollte bei einer vollständigen Installation vorhanden sein, ansonsten Internet

[5]ist besonders nützlich, wenn das HyTex-Hlfesystem sich nicht um Bildschirmauflösungen schert und dir nur den oberen Teil des Fensters zeigt. Passiert ua. bei Usern des „windowmaker“ unter X11.

[6]dafür ist es aber auch so, das die normale Sprache für einen Mathematiker aus vielleicht 10 Begriffen besteht, für uns aber aus über x*10000 (sei x eine Zahl größer 4)

[7]Schülerduden Mathe reicht...

[8]Mehr Daten habe ich nicht mehr, Buch ist auch nicht mehr im Handel

[9]Gibt es auch als Gesamtwerk für Mathe, Physik, Chemie

[10]numerisch, symbolische, wie TI-9x sind zumindest in Sachsen nicht zur Prüfung zugelassen

[11]Es gibt auch ein PC-Link, mit und ohne Software (Software ist auch per Internet unter http://www.ti.com/calc erhältlich)

Index

?, 8
2D, 17
3D, 17, 18

Ableitung, 13
abs, 7, 8
Absolutbetrag, 7, 8
acos, 7
Addition, 7
Algebra, lineare, 14
angle, 16
Arcuscosinus, 7
Arcussinus, 7
Arcustangens, 7
Areasinus, 7
asin, 7
asinh, 7
atan, 7
Ausgabe, Grafische, 17
Ausmultiplizieren, 9

Bücher, 21, 25
Bestimmung, Extrema-, 13
Bestimmung, Maxima-, 13
Bestimmung, Minima-, 13
Bibliothek, 14
binomisch, 9
Bruchumwandlung, 7

ceil, 7
cos, 7, 20
Cosinus, 7, 20

default, 8
det, 16
Determinante, 16
Dezimalzahl, 7
diff, 13
Differentialrechnung, 12, 13
digits, 8
dimen, 16
Dimension, 16
div, 7
Division, 7
Dokumentation, 21

e, 7
eigenvalues, 16
eigenvectors, 16
Eigenvektoren, 16
Eigenwerte, 16
Empfehlung, 21
Erweitern, 9
Euklidischer Algorithmus, 10
Euklidischer Algorithmus, erweiterter, 10
eulersche Funktion, 11
eulersche Zahl, 7
exp, 7
expand, 9
Exponentialfunktion, 7
export, 14

fact, 7
Faktorisierung, 9
Fakultät, 7
Fehler, 20
Fehlermeldung, 20
Flächenberechnung, 13
float, 7, 14
floor, 7
frac, 7

ganze Zahl, 7
Ganzzahlanteil, 7
Gaussches Eliminierungsverfahren, 16
gaussElim, 16
gaussJordan, 16
Genauigkeit, 8
ggT, 10
GIF, 17
Gleichungen, 11
Grad, 16
grad, 16
Grafik, 17
Grafik, dreidimensionale, 17, 18
Grafik, zweidimensionale, 17
Grenzwert, 12
Grenzwert, gegen links, 12
Grenzwert, gegen rechts, 12
Grenzwert, Verhalten im Unendlichen, 12

help, 8, 21
Hilfe, 8, 21
Hilfe, Beispiele, 21
history, 8, 14
HTML, 21

i, 7
ifactor, 9
igcd, 10
igcdex, 10

ilcm, 10
imaginäre Einheit, 7
infinity, 12
int, 13
Integer, 14
Integral, unbestimmtes, 13
Integralrechnung, 12, 13
Integralrechnung, numerische, 14
Integralrechnung, symbolische, 13
isprime, 10
ithprime, 10

jordanform, 16

Kamera, virtuelle, 17, 18
kgV, 10
Kurvendiskussion, 12

Lösen, 11
last, 8, 14
letzter Wert, 8
limit, 12
linalg, 14, 16
lineare Algebra, 14
linearSolve, 16
linsolve, 11
Literatur, 21, 25
ln, 7
loadlib, 14
Logarithmus, natürlicher, 7

Matrix, 14
Matrix, Auf\u00f6sung von, 16
Matrix, Definition, 15
Matrix, Determinante einer, 16
Matrix, Eigenvektoren einer, 16
Matrix, Eigenwert einer, 16
Matrix, Gauss-Jordan Elimination, 16
Matrix, Gaussches Eliminierungsverfahren, 16
Matrix, Grad einer, 16
Matrix, Jordanform, 16
Matrix, Rang einer, 16
Matrix, Sylvesterform einer, 16
Matrix, Transposition einer, 16
Matrizenrechnung, 14
max, 7
Maximum, 7
min, 7
Minimum, 7
mod, 7
Modulo, 7
Modus, Darstellungs-, 6
Multiplikation, 7
Multiplikation, skalare, 16

nextprime, 10
normalize, 16
Normalvektor, 16
Nullstellen, 11
numdivisors, 10

Pascal, 7
phi, 11
Pi, 7
Plot, 17
plot2d, 17, 18
plot3d, 14, 18
PostScript, 17
Potenzierung, 7
prettyprint, 6
primedivisors, 11
Primzahlen, 10, 11
Primzahlen, Zerlegung in, 9
Probleme, 20
PS, 17

Quadratwurzel, 7

random, 7
Rang, 16
rank, 16
reset, 8, 14
Rest, 7
round, 7
Rundung, 7

scalarProducts, 16
Sequenzierungsoperator, 8, 13
series, 12
sign, 7, 12
sin, 7
sinh, 7
Sinus, 7
Sinus hyperbolicus, 7
Skalarprodukt, 16
solve, 11–13
sqrt, 7, 14
Stammfunktion, 13
Stellen, 8
Subtraktion, 7
Sylvesterform, 16
Syntaxfehler, 20

tan, 7
Tangens, 7
Taschenrechner, 21
Teiler, Anzahl, 10
Teiler, größter, gemeinsamer, 10
Teiler, Prim-, Anzahl, 10, 11
teilerfremd, 11
Test, Primzahlen, 10
transpose, 16
Transposition, 16
trunc, 7
type, 15

undo, 8, 14
unendlich, 12

Variablen, 7
VCAM, 17, 18

Vektor, Definition, 15, 16
Vektor, Normal-, 16
Vektoren, Skalarprodukt zweier, 16
Vektoren, Winkel zwischen, 16
Vektorrechnung, 14, 16
Vertauschung Zeilen und Spalten, 16
Vielfaches, kleinstes, gemeinsames, 10
Vorzeichen, 7

Wiederholung, 8
Winkel zwischen Vektoren, 16
Winkelfunktion, 7
Wurzel, 7

x-mal, 8, 13

Zahl, komplexe, 7
Zahl, zufällige, 7
Zufallszahl, 7
Zuweisung, 7

Literaturverzeichnis

[1] MuPAD

Multiprocessing Algebra Data Tool

Benutzerhandbuch

MuPAD Version 1.1

Benno Fuchssteiner, Waldemar Wiwianka...

Basel; Boston; Berlin: Birkhäuser, 1993

ISBN 2-7643-2872-X