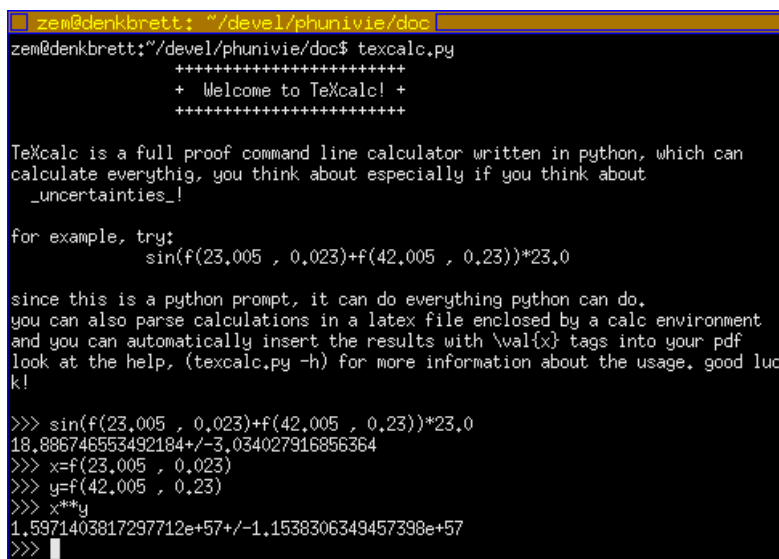


Anleitung zu den TeXcalc L^AT_EX 2_ε-Paketten

Hans Freitag (und einige andere)

21. März 2013

Dieses Dokument gibts als PDF zum Downloaden auf <http://www.nawi.at/~zem/texcalc/texcalc.de.pdf>



```
zem@denkbrett: ~/devel/phunivie/doc
zem@denkbrett:~/devel/phunivie/doc$ texcalc.py
+ Welcome to TeXcalc! +
TeXcalc is a full proof command line calculator written in python, which can
calculate everythig, you think about especially if you think about
_uncertainties_!
for example, try:
sin(f(23,005 , 0,023)+f(42,005 , 0,23))*23,0
since this is a python prompt, it can do everything python can do.
you can also parse calculations in a latex file enclosed by a calc environment
and you can automatically insert the results with \val{x} tags into your pdf
look at the help, (texcalc.py -h) for more information about the usage. good luc
k!
>>> sin(f(23,005 , 0,023)+f(42,005 , 0,23))*23,0
18,886746553492184+/-3,034027916856364
>>> x=f(23,005 , 0,023)
>>> y=f(42,005 , 0,23)
>>> x**y
1,5971403817297712e+57+/-1,1538306349457398e+57
>>>
```

Abbildung 1: Ein Screenshot des texcalc.py Taschenrechners

TeXcalcBesteht aus drei Komponenten.

1. texcalc.py ist ein in python geschriebener Taschenrechner der das Pythonmodul uncertainties benutzt.
2. texcalc.sty ist ein L^AT_EX 2_ε-Packet mit dem Berechnungen mit Messunsicherheiten direkt aus dem L^AT_EX 2_εDokument heraus möglich sind.
3. Zusatz styles die zwar nicht direkt etwas mit TeXcalczu tun haben, aber für das Schreiben von Physikprotokollen Nützlich sein können. Die beiden zusatzpakete gliedern sich in:
 - physictools.sty eine allgemeine toolsammlung die das Leben leichter machen soll
 - anfpralayout.sty Layoutparameter Deckbletter etc, die wahrscheinlich nie einmal irgendwo anders als an der Uniwien in der Physik verwendet werden.

Die Contrib Pakete sind eine Sammlung von L^AT_EX 2_ε-Makros von Studierenden der Fakultät, die sich im Laufe des Studiums als Hilfreich erwiesen haben.

Inhaltsverzeichnis

1	Features/Benutzung	3
1.1	Taschenrechner	3
1.2	Berechnungen direkt im \LaTeX File	3
2	Download	4
2.1	Abhängigkeiten	4
2.2	Dieses Packet	4
2.3	Software für Windows	4
2.3.1	\LaTeX	4
2.4	Python3	5
2.4.1	\TeX calc	5
2.5	Software für Linux(Debian)	5
2.6	Entwicklung und Fehlerbehebung	6
3	Installation	6
3.1	Linux	6
3.2	Windows	6
3.3	Bekannte \TeX calcProbleme und deren Lösungen	6
3.3.1	Integer vs float	6
4	Contrib	7
4.1	physictools Packet	7
4.1.1	Ergebnis Einleitung	7
4.1.2	image commands	8
4.1.3	23×10^{10}	8
4.1.4	$\frac{\partial}{\partial t}$	8
4.1.5	$\frac{\partial}{\partial x}$	8
4.1.6	$\frac{\partial x}{\partial y}$	8
4.1.7	$\frac{\partial^2 x}{\partial y \partial z}$	8
4.1.8	$\frac{\partial^2 x}{\partial y^2}$	8
4.1.9	$\vec{\nabla}$	9
4.1.10	$\hat{\mathbf{x}}$	9
4.1.11	$\langle x $	9
4.1.12	$ x\rangle$	9
4.1.13	$\langle x y\rangle$	9
4.1.14	\uparrow	9
4.1.15	\downarrow	9
4.1.16	$\uparrow\uparrow$	9
4.1.17	$\downarrow\downarrow$	10
4.1.18	$\langle x y x\rangle$	10
4.1.19	$\sqrt{\frac{x}{y}}$	10
4.2	anfpralayout	10
4.2.1	Titelseite	10

1 Features/Benutzung

1.1 Taschenrechner

Der Taschenrechner wird mit dem Kommando `texcalc.py` gestartet¹.

Abbildung 1 zeigt einen Screenshot des Taschenrechners an. Zahlen mit Unsicherheiten können durch $f(\textit{number}, \textit{uncertainty})$ notiert werden. In dem Prompt kann alles berechnet werden was in Python berechnet werden kann.

1.2 Berechnungen direkt im \LaTeX File

Die Rechnungen können auch direkt im LaTeX File durchgeführt werden, indem sie innerhalb des `calc` environments notiert werden²

```
\begin{calc}
x=f(23.55,0.02)
#r 23.55+/-0.02
def foo(x):
y=x**2
return y

z=33
#r 33
foo(z)
#r 1089
y=f(222.0,22)
#r 222.0+/-22.0

a=f(24423.0,220)
#r 222.0+/-22.0
\end{calc}
```

Die `#r` kommentarzeilen werden von `texcalc.py` automatisch eingefügt um die Fehlersuche und die Übersicht zu erleichtern. Sie enthalten das Ergebnis der letzten Variablenzuweisung bzw des letzten Statements.

Die errechneten Werte können jetzt mit dem `val` komando im Dokument verwendet werden. Der Parameter `[2]` gibt dabei bei einer normalen Zahl die zu rundenden Nachkommastellen an. Bei einer Zahl mit Unsicherheit gibt er die Anzahl der Signifikanten Stellen an, die dann automatisch gerundet und eingefügt werden. Default ist 2. Beispiel:

```
$$x_{5sigDigits}=\val[5]{x}$$
$$x_{2sigDigits}=\val{x}$$
$$x^2=\val{foo(x)}$$
$$z=\val{z}$$
$$y=\val{y}$$
$$a=\val{a}$$
```

¹Doppelklick geht auf Windows auch

²`\usepackage{texcalc}` nicht vergessen.

Wird zu:

$$x_{5sigDigits} = (23,550000 \pm 0,020000)$$

$$x_{2sigDigits} = (23,550 \pm 0,020)$$

$$x^2 = (554,60 \pm 0,95)$$

$$z = 33$$

$$y = (222 \pm 22)$$

$$a = (24423 \pm 220)$$

Übersetzt werden kann ein solches \LaTeX File mit folgenden Komandos:

<code>texcalc.py Datei.tex</code>	--> Fügt die <code>#r</code> kommentarzeilen ein
<code>texcalc.py -c Datei.tex</code>	--> erzeugt ein pdf <code>texcalc</code> ruft <code>pdflatex</code> auf
<code>pdflatex --shell-escape Datei.tex</code>	--> Erzeugt ein pdf <code>pdflatex</code> ruft <code>texcalc</code> auf
<code>lualatex --shell-escape Datei.tex</code>	--> Das gleiche wie <code>pdflatex</code>
<code>texcalc.py -C -y Datei.tex</code>	--> Wie <code>-c</code> nur werden alle <code>\val{}</code> fest durch die werte ersetzt, der prozess ist nicht reversibel, aber nützlich um schnell noch manuelle aenderungen vorzunehmen.

2 Download

2.1 Abhängigkeiten

- Eine LaTeX Distribution wie MikTeX oder TeXlive
- Python
- Python-numpy
- Python-uncertainties

2.2 Dieses Packet

- Download der aktuellen Version von GitHub <https://github.com/zem/texcalc/archive/master.zip>

2.3 Software für Windows

2.3.1 \LaTeX

Da `TeXcalcauf` \LaTeX aufsetzt, ist eine \LaTeX Umgebung erforderlich, außer du beschränkst dich auf die Funktion als Taschenrechner. MikTeX ist eine gute wahl, dazu optional Texnicenter als Arbeitsumgebung, in der Version 2.0 Beta.

- <http://mirrors.ctan.org/systems/win32/miktex/setup/basic-miktex-2.9.4813.exe>
- http://sourceforge.net/projects/texniccenter/files/TeXnicCenter/2%20Beta%201/TXCSetup_2Beta1_Win32.exe/download

An dieser Stelle wird auch empfohlen ein Versionskontrollsystem wie GIT zu installieren.

2.4 Python3

Da TexCalc auf python basiert, brauchst du auch Python, sowie die Module Numpy und Uncertainties.

- <http://www.python.org/ftp/python/3.3.0/python-3.3.0.msi>
- <http://sourceforge.net/projects/numpy/files/NumPy/1.7.0/numpy-1.7.0-win32-superpack-python3.3.exe/download>
- <http://pypi.python.org/packages/source/u/uncertainties/uncertainties-1.9.tar.gz>

Uncertainties kommt als tar archiv ohne grafischen installer daher, damit du das installiert bekommst entpackst du es erstmal in ein verzeichnis, dann führst du cmd.exe aus, und wechselst in das eben angelegte Verzeichnis³. Dann führst du `setup.py` installäus.

2.4.1 T_EXcalc

- <https://github.com/zem/texcalc/archive/master.zip>

Die Dateien

- `contrib/anfpralayout.sty`
- `contrib/physictools.sty`
- `texcalc.sty`

kopierst du nach: `C:\delimiter"026E30FProgramFiles\delimiter"026E30FMiKTeX2.9\delimiter"026E30Ftex\delimiter"026E30Flatex\delimiter"026E30Ftexcalc\delimiter"026E30F`. Den Ordner musst du anlegen. Die Datei `texcalc.py` kopierst du nach `C:\delimiter"026E30FProgramFiles\delimiter"026E30FMiKTeX2.9\delimiter"026E30Fmiktex\delimiter"026E30Fbin\delimiter"026E30F`.

Damit die `.sty` Dateien vom miktex auch gefunden werden, musst du Miktex Optionen als Administrator ausführen und Rebuild FNTDB machen.

Im Texnicenter kannst du unter `ausgabe/ausgabeprofile` definieren das Profil `LatexPDF` kopieren nach `Latex PDF TeXcalc` als pfad des LaTeX compilers verwendest du:

```
C:\Python33\python.exe "C:\Program Files (x86)\MiKTeX 2.9\miktex\bin\texcalc.py" -c
```

Zum Schluss legst du noch eine Verknüpfung von `C:\delimiter"026E30FProgramFiles\delimiter"026E30FMiKTeX2.9\delimiter"026E30Fmiktex\delimiter"026E30Fbin\delimiter"026E30Ftexcalc.py` auf den Desktop oder wo du es sonst hinhaben willst.

2.5 Software für Linux(Debian)

```
apt-get install texlive latexila python python-numpy python-uncertainties
```

(Wer will kann und soll natürlich auch vim verwenden, oder meinetwegen auch Emacs.)

³cd ist das Komando dafür

2.6 Entwicklung und Fehlerbehebung

Mitentwickeln kann jeder. Dafür brauchst du das Versionskontrollsystem GIT ⁴ einfach das Repository klonen, entweder von nawi.at:

```
git clone http://www.nawi.at/git/texcalc
```

Oder über github.com Was sich so tut kannst du im gitweb nachlesen: <http://www.nawi.at/gitweb/?p=texcalc;a=summary> oder halt auf github.com

Wenn du einen Patch machen willst kannst du

```
git format-patch origin/master
```

verwenden, und den Patch an zem@nawi.at mailen oder auf github einen fork anlegen und patchen.

3 Installation

3.1 Linux

Installation auf Linux geht mit:

```
make install install-contrib
```

3.2 Windows

Hoffentlich gibts bald ein install.bat script oder sowas.

Grundsätzlich müssen alle .sty in miktex eingebunden werden, wie das geht ist in der MikTeX Doku beschrieben:

<http://docs.miktex.org/manual/localadditions.html>

<http://docs.miktex.org/manual/texfeatures.html#includedirectory>

Dann muss texcalc.py noch irgendwo hingelegt werden wo es im Pfad liegt, bzw der Pfad muss angepasst werden. miktex/bin kann funktionieren. ;-)

3.3 Bekannte TeXcalcProbleme und deren Lösungen

3.3.1 Integer vs float

Da TeXcalc in python rechnet muss auch den dortigen Datentypen rechnung getragen werden. Eine Zahl die ohne Kommastelle angegeben wird, wird als Integer (ganze Zahl) interpretiert, und auch so dividiert. Beispiel:

⁴GIT ist übrigens auch Prima dafür geeignet um LaTeX dokumente zu verwalten, sowohl für einen Persönlich als auch wenn mehrere Personen ein Dokument bearbeiten.

```
>>> 23/4
5
>>> 23.0/4.0
5.75
>>>
```

Also am besten immer zahl.0 schreiben.

4 Contrib

4.1 physictools Packet

In der Preamble:

```
\usepackage{physictools}
```

4.1.1 Ergebnis Einleitung

In der Ergebnisse Section kommen einige sätze immer wieder vor. Hier die Templates:

```
\skalaerr
\mverr
\fiterr\usedqti
\usedgnuplot
\stopwatcherr
\multimetererr
\earthGerr
\syserr\usedtexcalc
\\
```

Und das wird zu:

Als Fehler für die Ablesegenauigkeit von Messskalen wurde die kleinste Skaleneinteilung verwendet. Als Fehler für Mittelwerte wurde die Standardabweichung verwendet. Der Fehler der Fitfunktionen wurde von der Plot Software ermittelt. Als Plot Software wurde QTIplot verwendet. Als Plot Software wurde GNUplot verwendet. Als Fehler für Zeitmessungen mit der Stoppuhr wurde die letzte Ziffer der Digitalanzeige ± 1 verwendet. Als Fehler für Messungen mit dem Multimeter wurde die letzte Ziffer der Digitalanzeige ± 1 verwendet. Als Wert für die Gravitationsbeschleunigung auf der Erde wurde $g = (9,81 \pm 0,03)$ verwendet. Der Messfehler ergibt sich aus der größeren Differenz der Literaturwerte für die Gravitation an Pol und Äquator zum Mittelwert. Systematische Fehler wurden mit Gaussscher Fehlerfortpflanzung ermittelt. Die Gausssche Fehlerfortpflanzung wurde mit der Software T_EXcalc durchgeführt.

Das geht natürlich auch auf Englisch ohne ngerman packet:

As uncertainty for measuring scales the smallest scale graduation was used. As uncertainty for meanvalues the standard deviation was used. The uncertaintys for fit funktions where collected from the plot software. As plot software QTIplot was used. As plot software GNUplot was used. As uncertainty for time measurements with the stopwatch the last digit of the display ± 1 was used. As uncertainty for measurements with the multimeter the last digit

of the display ± 1 was used. As value for the acceleration of gravity on earth $g = (9,81 \pm 0,03)$ was used. The uncertainty results from the bigger difference of the literature values of gravitation at pole and equator to the mean value. Systematic errors were calculated with gaussian error propagation. The gaussian error propagation was done by the software T_EXcalc.

4.1.2 image commands

```
\image{texcalc.png}{Ein Screenshot des texcalc.py Taschenrechners}
```

is actually a shortcut for:

```
\begin{figure}[h]
\centering
\includegraphics[width=0.7\textwidth]{texcalc.png}
\caption{Ein Screenshot des texcalc.py Taschenrechners}
\label{texcalc.png}
\end{figure}
```

4.1.3 23×10^{10}

```
$$23\text{E}{10}$$
```

4.1.4 $\frac{\partial}{\partial t}$

```
$$
\ddt
$$
```

4.1.5 $\frac{\partial}{\partial x}$

```
$$
\dnach{x}
$$
```

4.1.6 $\frac{\partial x}{\partial y}$

```
$$
\ddnach{x}{y}
$$
```

4.1.7 $\frac{\partial^2 x}{\partial y \partial z}$

```
$$
\dddnach{x}{y}{z}
$$
```

4.1.8 $\frac{\partial^2 x}{\partial y^2}$

```
$$
\dznach{x}{y}
$$
```


4.1.9 $\vec{\nabla}$

\$\$
\nabla
\$\$

4.1.10 \hat{x}

\$\$
\evec{x}
\$\$

4.1.11 $\langle x|$

\$\$
\bra{x}
\$\$

4.1.12 $|x\rangle$

\$\$
\ket{x}
\$\$

4.1.13 $\langle x|y\rangle$

\$\$
\bracket{x}{y}
\$\$

4.1.14 \uparrow

\$\$
\up
\$\$

4.1.15 \downarrow

\$\$
\down
\$\$

4.1.16 $\uparrow\uparrow$

\$\$
\upup
\$\$

4.1.17 \Downarrow

```
$$  
\downdown  
$$
```

4.1.18 $\langle x|y|x \rangle$

```
$$  
\sandwich{x}{y}  
$$
```

4.1.19 $\sqrt{\frac{x}{y}}$

```
$$  
\fsqrt{x}{y}  
$$
```

4.2 anfpralayout

In der Preamble:

```
\usepackage{anfpralayout}
```

oder

```
\usepackage[option]{anfpralayout}
```

Optionen können sein:

basti Sebastians Layout Parameter (margin=2.5cm)

motz Motz's Layout Parameter

4.2.1 Titelseite

Für das Anfängerpraktikum ist eine Besondere Titelseite nötig, die kann natürlich ganz einfach gesetzt werden:

```
\semester{SoSe 2011}  
\fakulty{Fakultät für Sonstwas}  
\title{Physikalisches Praktikum\\für Lehramtskandidaten}  
\experiment{1. F Nord}  
\date{abgabedatum}  
\author{Hans Freitag}  
\group{3}  
\supervisor{someone}
```

```
\makeanfpratitle
```

Die auskommentierten Parameter sind optional. Und so siehts aus⁵:

⁵\date und \author gehen zur Zeit nur wenn der Block am Anfang eines Dokumentes steht

Fakultät für Physik

SoSe 2011

PROTOKOLL

Experiment (Nr., Titel):

1. Fnord

Datum:

Namen:

Kurstag/Gruppe: 3

Betreuer:

someone