



# Stack (Maxima)

## Mathematische Aufgaben mit Moodle/Stack

STACK (System for Teaching and Assessment using a Computer algebra Kernel) ermöglicht es Ihnen, mathematische Fragestellungen in Moodle-Tests zu realisieren. Das Plugin nutzt das Computeralgebrasystem „Maxima“ um mathematische Eigenschaften der eingegebenen Antworten bzw. Ausdrücke zu ermitteln und diese zu bewerten.

Um Ihnen den Einstieg zu diesem Fragetyp zu erleichtern, können Sie sich in dem folgenden Video einen Überblick verschaffen:

[stackfragen\\_einfuehrung\\_video.mp4](#)

Im Folgenden wird zur Einführung eine Teilmenge der Möglichkeiten dieses Aufgabentyps beschrieben. Ergänzende Informationen können Sie den entsprechenden Dokumentationen entnehmen:

<http://maxima.sourceforge.net/docs/manual/de/maxima.html>

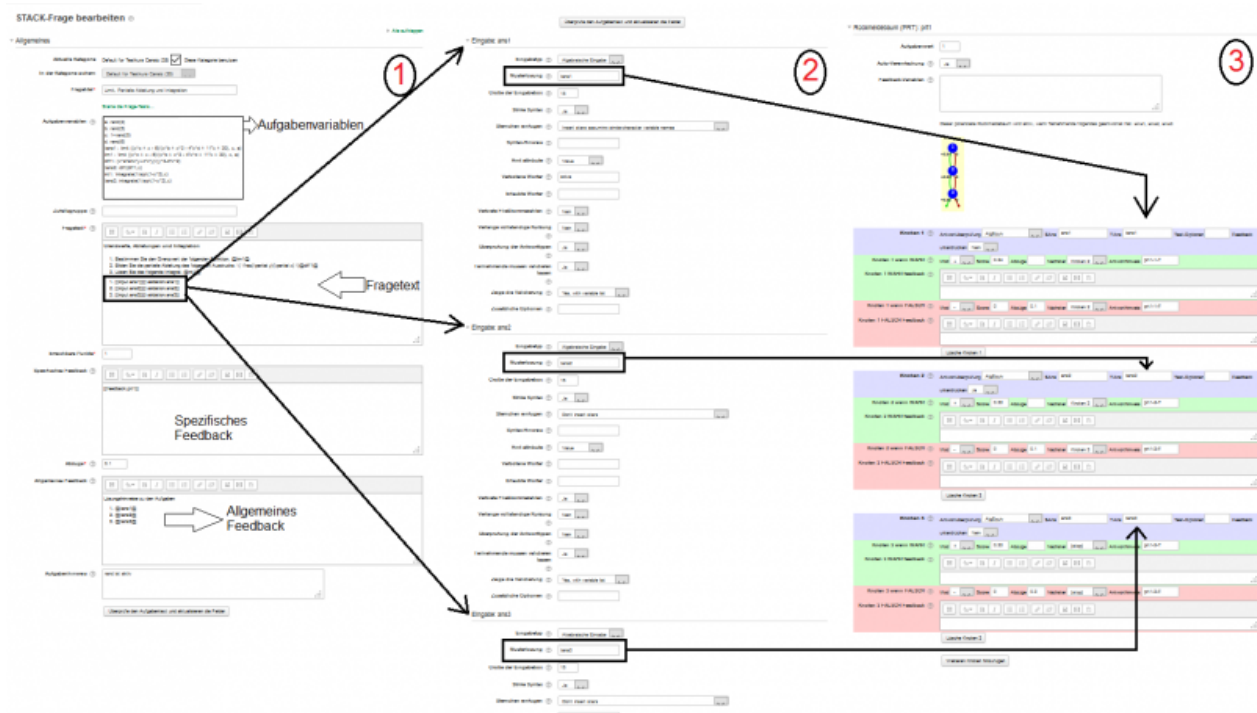
[https://moodle.org/plugins/qtype\\_stack](https://moodle.org/plugins/qtype_stack)

<http://michel.gosse.free.fr/documentation/fichiers/maxima.pdf>

[https://stack2.maths.ed.ac.uk/demo/question/type/stack/doc/doc.php/Authoring/Authoring\\_quick\\_start.md](https://stack2.maths.ed.ac.uk/demo/question/type/stack/doc/doc.php/Authoring/Authoring_quick_start.md)

## Grundlagen zum Stack Fragetyp

Im Folgenden wird die grundlegende Struktur einer Stack-Frage beschrieben. Im **ersten Bereich** werden die Aufgabenvariablen ( $ans_1, ans_2, \dots, ans_n$ ) definiert, die konkrete Frage/Aufgabe formuliert, die Art und die Anzahl der Eingabe/Input-Felder (Textfelder, Matrizen, ...) festgelegt und das allgemeine Feedback (z.B. Lösungshinweise) vorbereitet. Im **zweiten Bereich** werden die spezifischen Eigenschaften der Eingabefelder (Platzhalter, Typ des Feldes, Musterantwort etc.) festgelegt. Dabei können bzw. sollten Sie in dem jeweiligen Feld für die Musterlösung, die im ersten Bereich definierten Aufgabenvariablen ( $tans_1, tans_2, \dots, tans_n$ ) verwenden. Im **dritten Bereich** wird zum Schluss die Auswertungslogik erstellt. Dazu werden sogenannte „Potential-Response-Trees“ (PRT, Rückmeldebäume) erstellt (siehe [Potential-Response-Trees](#)).



## Aufbau der Stack Fragen

### Frage Titel

Geben Sie hier einen aussagekräftigen Titel z.B. „Nullstellenbestimmung - 1“ ein.

### Aufgabenvariablen

Im Bereich „Aufgabenvariablen“ können Variablen und entsprechende Wertzuweisungen und mathematische Ausdrücke vorbereitet werden.

variablenname : Ausdruck

#### Aufgabenvariablen ?

```
t : f(x)
a : f(x)=0
f(x)= x^2 - 5*x - 36
tans1 : solve([ x^2 - 5*x - 36=0,[x]])
```

```
t : 2           // der Variable t wird der Wert 2 zugewiesen
a : f(x)=x^2    // Abbildung einer Funktion
b : diff(x^2,x) // differenziere x^2 nach x
```

```
c : diff(a,x)      // alternativ: differenziere a nach x
d : rand(15)       // Zufallszahlen von 0 bis 14
```

Siehe dazu auch: [Arbeiten mit Zufallszahlen](#), [Maxima Dokumentation](#), [Arbeiten mit Funktionen](#)

## Fragetext

In diesen Abschnitt erfolgt die konkrete Aufbereitung bzw. Formulierung der Frage/Aufgabe. Bitte beachten Sie, dass die Syntax der Stackausdrücke sich verändert hat! @...@ wird zu {@...@}

**Fragetext\*** ?

$f(x) = \sqrt{x^2 - 5x - 36}$

Bestimmen Sie die Werte von Funktion {@a@}

[[input:ans1]] [[validation:ans1]]

Mit Hilfe von Medieninhalten (Bilder, Videos) und LaTeX stehen Ihnen einige Möglichkeiten zur Formulierung von komplexen Aufgaben zur Verfügung. Sie können die erweiterten Features in dem Menü des WYSIWYG-Editors nutzen, um Ihre Aufgaben z.B. mit LaTeX-Ausdrücken zu formulieren:

1. Klicken Sie dazu auf das Icon „mehr Symbole anzeigen“
2. Öffnen Sie den Gleichungseditor

## Eingabefelder verwenden / erweitern

Im Aufgabentyp Moodle Stack wird das erste „Eingabefeld“ automatisch beim Anlegen einer Aufgabe erstellt. Wenn zusätzliche Eingabefelder benötigt werden, können Sie beliebig viele Eingabefelder nach dem folgenden Muster

```
[[input:ans2]] [[validation: ans2]]
[[input:ans3]] [[validation: ans3]]
...
[[input:ansx]] [[validation: ansx]]
```

in einer Aufgabe hinzufügen (nach jedem Hinzufügen eines neuen Eingabefelds muss die Frage aktualisiert werden). Für jedes Eingabefeld muss eine Musterlösung (z.B. Aufgabenvariable oder Ausdruck) festgelegt werden.

Frage**text**

Berechnen Sie den Grenzwert  
 $\lim_{x \rightarrow 0} \frac{x^a \cdot c \cdot x - 6}{x^b \cdot d \cdot x^3 - 19x^d \cdot 11x + 30}$

Eingabetyp Algebraische Eingabe

Eingabe: ans2 Musterlösung tans2

[[input.ans1]][[validation.ans1]]  
 [[input.ans2]][[validation.ans2]]  
 [[input.ans3]][[validation.ans3]]

für jedes Eingabefeld wird eine Musterlösung festgelegt

Eingabe: ans3 Musterlösung tans3

Eingabetyp Algebraische Eingabe

Eingabe: ans1 Musterlösung tans1

## Allgemeines Feedback

Beim allgemeinen Feedback handelt es sich um eine Ergänzung der Musterlösung oder einen Hinweis zur Lösung einer Aufgabe. Dieses Feedback wird ihren Studenten nach oder (optional) während der Bearbeitung der Aufgabe angezeigt.

## Optionen / Einstellungen zu den Eingabefeldern

- Art der Eingabe - Auswirkung, Beschreibung, Screenshots (Matrix, Wahr/Falsch, Textfeld...)

Erklären Sie die Regel von L'Hospital. Frage nachbessern | Starte die

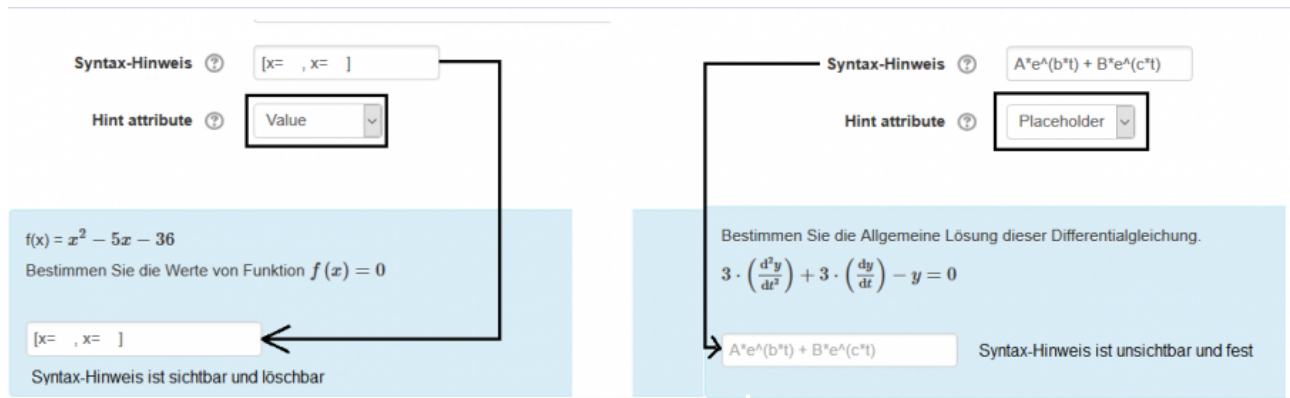
die Regel von L'Hospital besagt dass

$\lim_{x \rightarrow 2} \frac{x^3 - 8}{x - 2} = 0$

Nicht beantwortet   
 Falsch  
 Wahr  
 Nicht beantwortet

Bestimmen Sie die Inverse dieser Matrix:  $\begin{bmatrix} 2 & 5 \\ 1 & 3 \end{bmatrix}$

- Syntax Hint oder Syntax-Hinweise sind Platzhalter, innerhalb der Eingabefelder, die Ihren Studenten eine Vorlage liefert, wie ein Term eingegeben werden soll. (z.B. sqrt, [x=...,x=..]).



## Rückmeldebaum (PRT)

Die **P**otential **R**esponse **T**rees (PRT) sind binäre Baumstrukturen und werden zum Aufbau einer Auswertungslogik der Aufgabe verwendet. Dabei werden Antworten bzw. die Eingaben eines Clienten mit einem math. Ausdruck (z.B. der „Musterlösung“) verglichen.

ResponseTrees können flexibel aufgebaut werden. Die Knoten können (nach dem Sie angelegt wurden) miteinander verknüpft werden (z.B. Folgefehler-Szenarien). Um Knoten hinzuzufügen, müssen Sie lediglich auf den Button „weitere Knoten hinzufügen“ klicken.

- wie werden Knoten und Punkte verarbeitet (Score, Abzüge):

Die maximal erreichbare Punkte (Score) ist 1,00. Wenn die Teilnehmenden falsche Antwort haben, wird 0.25 Punkte abgezogen.

- wie funktioniert die Antwortüberprüfung

Eine **Antwortüberprüfung (Answer test)** wird verwendet zwei Ausdrücke zu vergleichen, festzustellen ob sie einige kriterien erfüllen.



## Hinweise zur Antwortüberprüfung

### NumDecPlaces

Tests (i) whether the student's answer is equivalent to the teacher's and is written to opt decimal places. The option, which must be a positive integer, dictates the number of digits following the decimal separator .. Note that trailing zeros are required, i.e. to two decimal placed you must write 12.30 not just 12.3. The test rounds the numbers to the specified number of decimal places before trying to establish equivalence.

z.B: we want an answer of the question to have 3 decimal places, it means that the option is 3 and the answer is 15.3468. This test rounds the number 15.3468 to 15.347 because 3 decimal places are wanted.

### NumRelative & Numabsolute

The option to these tests is a tolerance. The default tolerance is 0.05.

Relative: Tests whether  $\text{abs}(sa-ta) \leq \text{opt} * \text{abs}(ta)$   
Absolute: Tests whether  $\text{abs}(sa-ta) < \text{opt}$

NumRelative and NumAbsolute can also accept lists and sets. Elements are automatically converted to floats and simplified (i.e.  $\text{ev}(\text{float}(ex), \text{simp})$ ) and are compared to the teacher's answer using the appropriate numerical test and accuracy. A uniform accuracy must be used. With lists the order is important, but with sets it is not. Checking whether two sets are approximately equal is an interesting mathematical problem....



noch im Aufbau



Fix Me!

## Grundlagen

### Zufallszahlen

Zufallszahlen von 0 - 98 oder Zufallszahlen aus einer Menge von Zahlen {1,2,3,4}

```
f1 : rand(99)
f2 : rand([1,2,3,4])
```

Eine Liste kann wie folgt erstellt werden : [a,b,c]

## Konstanten

```
e          // Eulerische Zahl
pi         // π Kreiszahl
infinity   // ∞ Komplex unendlich
inf        // positive Unendlich
minf       // negative Unendlich
i          // imaginäre Einheit i (in der Elektrotechnik bezeichnet es mit j)
phi        // Goldener Schnitt  $\phi = (1+\sqrt{5})/2$ 
```

Für mehrere Informationen bitte klicken Sie auf den folgenden Link: [Mathematische Konstanten](#)

## Matrizen

Beachten Sie bei der Realisierung von Aufgaben mit Matrizen unbedingt die entsprechenden Rechenregeln!

1. Definieren Sie Ihre Aufgabenvariablen

```
A : rand(matrix([1,2,4,3],[4,5,3,5],[4,5,3,5])) // Matrix A
B : rand(matrix([1,3,5],[1,2,1],[7,1,2],[1,5,9])) // Matrix B
tans : A.B //
Matrixmultiplikation
```

2. Erstellen Sie nun Ihre Fragestellung

```
Berechnen Sie @A@ \(\cdot\) @B@
// Zwischen den @-Symbolen werden Aufgabenvariablen verarbeitet,
aufbereitet und ausgegeben
// Zwischen \(\dots\) kann LaTeX-Code untergebracht werden
```

Aufgabenvariablen ?

```
A: rand(matrix([1,2,5,4],[4,5,3,5],[4,5,3,5]));
B: rand(matrix([1,3,5],[1,2,1],[7,1,2],[1,5,9]));
tans: A.B
```

-> unsere Variablen sind  
A und B  
-> die Matrizen sind  
randomisiert

Musterlösung

Zufallsgruppe ?

Fragetext\* ?

Berechnen Sie  $A \cdot B$

mit @...@ Zeichen werden die Variablen validiert, folglich wird dann Rand Funktion aktiviert

[[input:ans1]] [[validation:ans1]]

3. Im nächsten Schritt wird das Matrix-Eingabefeld einer Aufgaben ermöglicht indem als Eingabetyp „Matrix“ verwendet wird. Wenn Sie diesen Schritt nicht durchführen, sehen Ihre Studenten, nur ein einfaches Eingabefeld!

▼ Eingabe: ans1

Eingabetyp ? Matrix

Musterlösung ?

Größe der Eingabebox ?

Strike Syntax ?

Sternchen einfügen ?

- Algebraische Eingabe
- Checkbox
- Dropdown-Liste
- Einzelnes Zeichen
- Matrix**
- Radio
- Textfeld
- Units
- Wahr/Falsch



## Vorschau Frage Matrix 3

**Frage 1**  
 Unvollständig  
 Erreichbare Punkte:  
 1,00

Berechnen Sie  $\begin{bmatrix} 0 & 0 & 0 & 3 \\ 2 & 4 & 1 & 2 \\ 0 & 3 & 2 & 1 \end{bmatrix} \cdot \begin{bmatrix} 0 & 1 & 2 \\ 0 & 0 & 0 \\ 2 & 0 & 0 \\ 0 & 1 & 3 \end{bmatrix}$  ➔ **Randomisierte Werte**

$\left[ \begin{array}{|c|c|c|} \hline \text{ } & \text{ } & \text{ } \\ \hline \text{ } & \text{ } & \text{ } \\ \hline \text{ } & \text{ } & \text{ } \\ \hline \end{array} \right]$ 
➔

Eingabefeld

## Funktionen und Funktionstypen

### Ausdrücke automatisch auswerten

Die Informationen zu den möglichen Funktionen

→ <http://maxima.sourceforge.net/docs/manual/maxima.html>

### solve / linsolve

Beispiel 1

```
f: x^2-2*x+1=0
tans1:solve([f],[x])
```

Beispiel 2 (Lineargleichungen)

```
f1: x+3*y+z=9
f2: x-8*y+6*z=17
f3: 2*x+7*y-9*z=22
tans1:linsolve([f1,f2,f3],[x,y,z])
```

Für mehrere Informationen sehen Sie auch [Maxima-Equations](#)

### limit

```
s: (1+1/x)^x
tans1:limit(s,x,infinity)
```

## ode2 (Differentialgleichungen)

Beispiel:

```
dg1 : 'diff(y,x,2) - 3*'diff(y,x) - 4*y=0
```

Dabei bedeutet das Symbol `< ' >` lediglich, dass Maxima den Ausdruck „berechnet“. Das ist z.B. dann praktisch, wenn man in diesem Beispiel die Funktion nicht tatsächlich differenzieren, aber eben entsprechend in diesem Kontext  $\frac{dy}{dx}$  ausgeben möchte.

### Aufgabenvariablen ?

Das CAS lieferte folgende Fehler zurück: tans1 : ode2(dg1,y,x) verursacht durch den folgenden Fehler: (%t11) 8\*y = 0"not a proper differential equation"

```
m:2+rand(5)
k:3+rand(18)
dg1 : diff(y,x,2)+ m*diff(y,x)+k*y=0
tans1 : ode2(dg1,y,x)
```

Ohne das Symbol '  
verursacht einen Fehler

Befehle zur Lösung dieser Differentialgleichung:

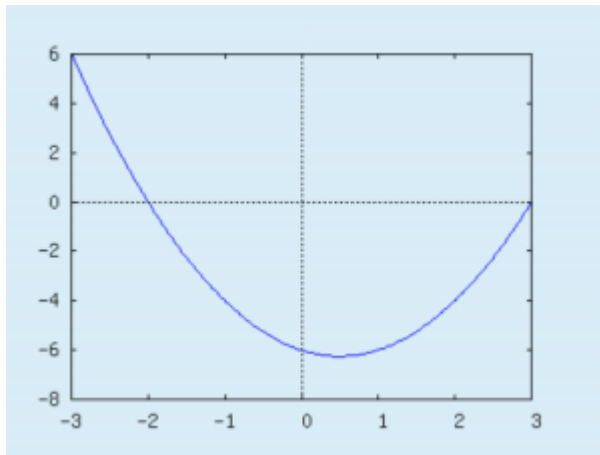
```
tans1:ode2(dg1,y,x)
```

Siehe auch: [Differential Equations](#)

## plot

Beispiel

```
plot2d(x^2-x-6,[x,-3,3])
```

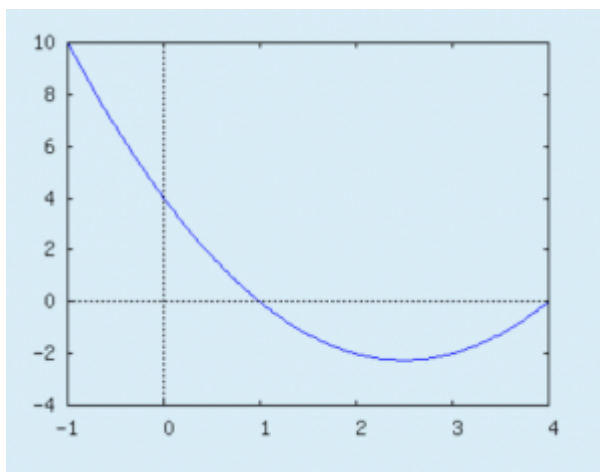


Bei der Randomisierung der Werte in diesem Diagramm:

```
a: 1+rand(5)
b: 3+rand(25)
d: -1*rand(5)
e: 1+rand(5)
t: x^2-a*x+b
tans1: plot([t],[x,d,e])
```



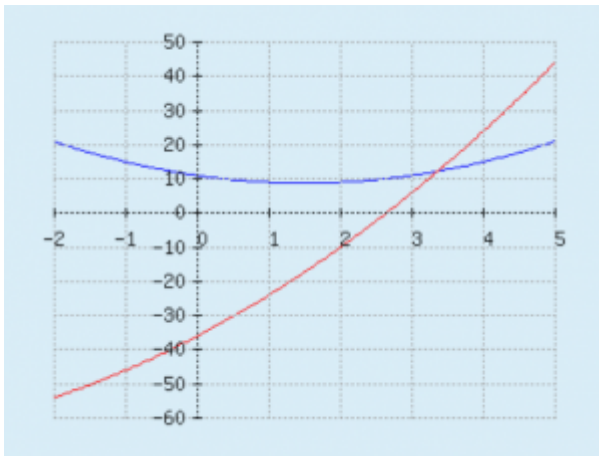
Man sollte dabei auf die Grenzen achten! Hier z.B. muss „d“ kleiner „e“ sein.



Ein Diagramm mit mehrere Plot-Optionen (grid, color, labels, usw.):

```
a: 1+rand(5)
b: 3+rand(25)
d: -1*rand(5)
e: 1+rand(5)
t1: x^2-a*x+b
t2: x^2 + b*x-12*a
```

```
tans1: plot([t1,t2],[x,d,e],[xlabel,"x-Axis"],[ylabel,"y-Axis"],[color,blue,red],[axes,true],[box,false],grid2d)
```



Stack-Plot

## Videos zur Hilfe für Erstellung der bestimmten Aufgabentypen

### Matrixaufgabe

[stack-matrix\\_video.mp4](#)

### Matrixaufgabe mit randomisierten Werten

[matrix\\_2\\_rand\\_funktion\\_.mp4](#)

### Lineare Gleichungen mit randomisierten Werten

[lineare\\_gleichungen\\_mit\\_randomisierten\\_werten.mp4](#)

### Lineare Differentialgleichung mit konstanten Koeffizienten

[differentialgleichung\\_mit\\_konstanten\\_koeffizienten.mp4](#)

[mitarbeitende], [stuhlbein], [moodle]

Direkt-Link:

[https://doku.tu-clausthal.de/doku.php?id=multimedia:moodle:stack\\_maxima&rev=1549631413](https://doku.tu-clausthal.de/doku.php?id=multimedia:moodle:stack_maxima&rev=1549631413)

Letzte Aktualisierung: **14:10 08. February 2019**

