



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 Computeralgebra-System „Maxima“, um mathematische Eigenschaften der eingegebenen Antworten bzw. Ausdrücke zu ermitteln und diese zu bewerten.

Hilfen zur Einarbeitung

Einen Vortrag (Dauer: 36:09), in dem STACK von anderen Aufgabentypen, wie z.B. Multiple Choice, abgegrenzt und ein Überblick über die Möglichkeiten von STACK gegeben wird, finden Sie hier:
<https://www.rubel.rub.de/news/vortrag-vom-1692015-online-intelligente-rechenaufgaben-mit-stack-moodle>

Sehr hilfreich ist auch der Moodle-Kurs, der in der Ruhr-Universität Bochum von Herrn Kallweit und seinem Team erstellt wurde:

<https://moodle.ruhr-uni-bochum.de/m/course/view.php?id=13674>

Dort gibt es viele Videos und Beispiele zum Ausprobieren.

Wenn ein Benutzeraccount unter

<https://moodle.ruhr-uni-bochum.de/m/login/signup.php>

erstellt wurde, kann man in dem Kurs sogar die vorgestellten Beispiele im XML-Format herunterladen.

In diesem Video können Sie sich einen Überblick über die Erstellung von Aufgaben in STACK verschaffen:

[stackfragen_einfuehrung_video.mp4](#)

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

Maxima Manuals:

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

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

Offizielle Plugin-Seite:

https://moodle.org/plugins/qtype_stack (englisch)

Weiterer, durch Videos unterstützter Kurs „Authoring quick start“ (besteht z.Zt. aus acht Einheiten) von den Entwicklern von STACK:

https://github.com/mathsg/moodle-qtype_stack/blob/master/doc/en/Authoring/Authoring_quick_start.md (Startseite, englisch)

Die Videos sind unter z.B.

https://www.youtube.com/watch?time_continue=223&v=cpwo-D6EUgA&feature=emb_logo direkt bei YouTube zu finden

Grundlagen zum Stack Fragetyp

Im Folgenden wird die grundlegende Struktur einer Stack-Frage beschrieben. Im **ersten Bereich** werden die Aufgabenvariablen (`ans1, ans2, ..., ansn`) definiert, die konkrete Frage/Aufgabe formuliert, die Art und die Anzahl der Eingabe/Input-Felder (Textfelder, Matrizen, etc.) 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 (`tans1, tans2, ..., tansn`) 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

Fragestitel

Geben Sie hier einen aussagekräftigen Titel, wie 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]])
```

ai

```
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 sich die Syntax der Stackausdrücke 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 im 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.



Allgemeines Feedback

Beim allgemeinen Feedback handelt es sich um eine Ergänzung der Musterlösung. Dieses Feedback wird allen Studierenden nach dem Ende des Versuchs angezeigt, unabhängig davon, welche Antworten sie gegeben haben. In diesem Feld können die Aufgabenvariablen verwendet werden, allerdings nicht die Eingaben der Studierenden, wie ans1, ans2 usw.

Optionen / Einstellungen zu den Eingabefeldern

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

Erklären Sie die Regel von L'Hospital.

die Regel von L'Hospital besagt
dass

Textfeld

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

Nicht beantwortet
Falsch
Wahr
Nicht beantwortet

Wahr/Falsch

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

Matrix Eingabefeld

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

Syntax-Hinweis $[x= \quad , x= \quad]$

Hint attribute Value

$f(x) = x^2 - 5x - 36$

Bestimmen Sie die Werte von Funktion $f(x) = 0$

$[x= \quad , x= \quad]$

Syntax-Hinweis ist sichtbar und lösbar

Syntax-Hinweis $A^*e^{(b^*t)} + B^*e^{(c^*t)}$

Hint attribute Placeholder

Bestimmen Sie die Allgemeine Lösung dieser Differentialgleichung.

$3 \cdot \left(\frac{d^2y}{dt^2} \right) + 3 \cdot \left(\frac{dy}{dt} \right) - y = 0$

$A^*e^{(b^*t)} + B^*e^{(c^*t)}$

Syntax-Hinweis ist unsichtbar und fest

Rückmeldebaum (PRT)

Die **Potential Response Trees** (PRT) sind binäre Baumstrukturen und werden zum Aufbau einer Auswertungslogik der Aufgabe verwendet. Dabei werden Antworten bzw. die Eingaben eines Studenten mit einem math. Ausdruck (z.B. der „Musterlösung“) verglichen.

ResponseTrees können flexibel aufgebaut werden. Die Knoten können (nachdem 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 Anzahl der maximal zu erreichenden Punkte (Score) beträgt 1,00. Wenn die Teilnehmenden falsche Antwort gegeben haben, werden 0,25 Punkte abgezogen.

- wie funktioniert die Antwortüberprüfung

Eine **Antwortüberprüfung(Answer test)** wird verwendet um zwei Ausdrücke zu vergleichen und um festzustellen ob sie einige Kriterien erfüllen.

Je nach dem Frage-Typ sind die Antwortüberprüfungen vorhanden z.B. Int. Diff oder RegExp

hier muss mit der Musterlösung abgeglichen werden

Knoten 1 Antwortüberprüfung AlgEquiv SAns **ans1** TAns **tans1** Test-Optionen Feedback

unterdrücken Nein

Knoten 1 wenn WAHR Mod = Score 1 Abzüge Nächster [stop] Antworthinweis prt1-1-T

Knoten 1 WAHR feedback Mod = Score 0 Abzüge 0.25 Nächster [stop] Antworthinweis prt1-1-F

Knoten 1 wenn FALSCH Mod = Score 0 Abzüge 0.25 Nächster [stop] Antworthinweis prt1-1-F

Weiteren Knoten hinzufügen

Hinweise zur Antwortüberprüfung

NumDecPlaces

NumDecPlaces prüft, ob die Antwort des Schülers die des Lehrers entspricht und auf die vorgegebene Anzahl von Dezimalstellen geschrieben wird. Die Option, die eine positive ganze Zahl sein muss, gibt die Anzahl der Stellen an, die auf das Dezimaltrennzeichen folgen. Beachten Sie, dass nachfolgende Nullen erforderlich sind, d.h., bei einer Vorgabe von zwei Dezimalstellen, müssen Sie 12.30 und nicht nur 12.3 schreiben. Der Test runden die Zahlen auf die angegebene Anzahl von Dezimalstellen, bevor versucht wird, eine Äquivalenz herzustellen.

Beispiel: Wir möchten für die Frage eine Antwort mit 3 Dezimalstellen haben. Dies bedeutet, dass die Option den Wert 3 enthalten sollte. Sei nun 15.3468 eine möglich Antwort. Dieser Test runden die Zahl 15.3468 auf 15.347, da 3 Dezimalstellen gewünscht werden.

NumRelative & Numabsolute

Die Option für diese Tests ist eine Toleranz. Die Standardtoleranz beträgt 0,05.

```
Relative: Tests whether abs(sa-ta) <= opt * abs(ta)
Absolute: Tests whether abs(sa-ta) < opt
```

NumRelative und NumAbsolute können auch Listen und Sets akzeptieren. Elemente werden automatisch in Floats umgewandelt und vereinfacht (z.B. `Ev(Float(ex),simp)`) und mit der Antwort des

Lehrers unter Verwendung des entsprechenden numerischen Tests und der entsprechenden Genauigkeit verglichen. Es muss eine einheitliche Genauigkeit verwendet werden. Bei Listen ist die Reihenfolge wichtig, bei Sets jedoch nicht. Überprüfung, ob zwei Sätze ungefähr gleich sind, ist ein interessantes mathematisches Problem...

Grundlagen

Zufallszahlen

Eine Zufallszahl von 0 - 98 oder eine Zufallszahl aus der Menge {1,2,3,4}

```
z1 : rand(99)
z2 : rand([1,2,3,4])
```

Eine Zufallszahl mit den folgenden möglichen Werten: Min: 2, Max: 25, Inkrement: 3

```
z3 : rand_with_step(2,25,3)
```

oder eine Zufallszahl ohne den Wert von z3

```
z4 : rand_with_prohib(2,8,[z3])
```

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

Konstanten

```
e          // Eulersche Zahl
pi         // π Kreiszahl
infinity   // ∞ komplex unendlich
inf        // positiv unendlich
minf       // negativ unendlich
i          // imaginäre Einheit i (in der Elektrotechnik oft bezeichnet mit
j)
phi        // Goldener Schnitt φ=(1+√5)/2
```

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

Trigonometrische Funktionen

`sin(x), cos(x), tan(x) und cot(x)`

und deren Inverse:

`asin(x), acos(x), atan(x) und acot(x)`

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 \(...\) 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
```

Musterlösung 

→ unsere Variablen sind A und B
→ die Matrizen sind randomisiert

Zufallsgruppe 

FrageText* 

Berechnen Sie $\text{ans1} = \text{A} \cdot \text{B}$

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

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

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

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

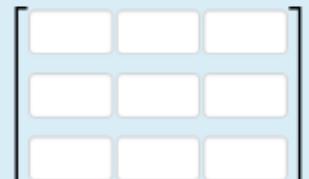
Vorschau Frage Matrix 3

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**

  **Eingabefeld**

Prüfen

Funktionen und Funktionstypen

Ausdrücke automatisch auswerten

Die Informationen zu den möglichen Funktionen

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

ev(Evaluate)

```
f1: x+3*y+z;
tans1: ev(f1,x=2,y=1,z=1);
tans1=5
```

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 nicht auswertet. 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
```

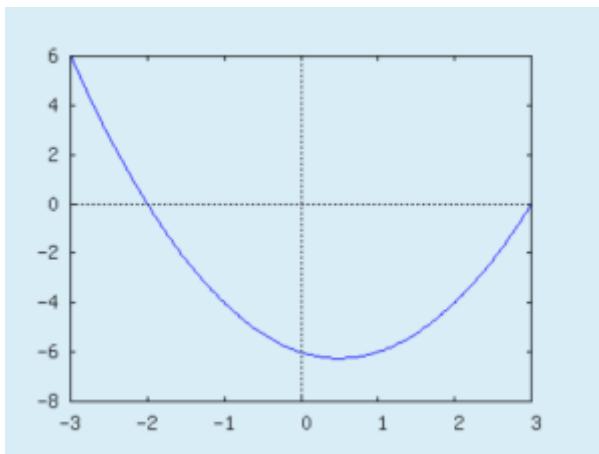
Ohne das Symbol '
verursacht einen
Fehler

Befehle zur Lösung dieser Differentialgleichung:

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

Siehe auch: [Differential Equations](#)

plot



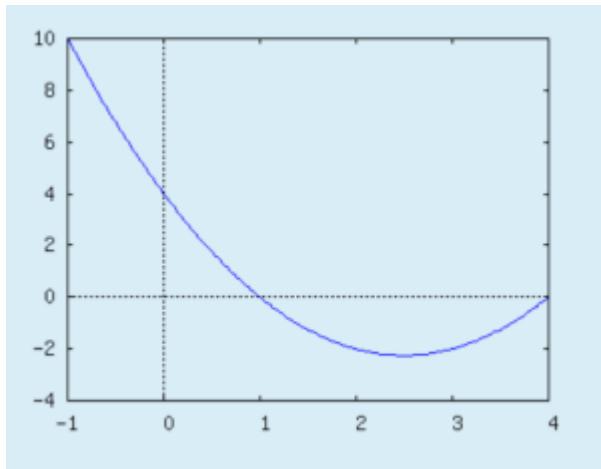
Beispiel:

Mit Randomisierung der Werte für den Plot:

```
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])
```



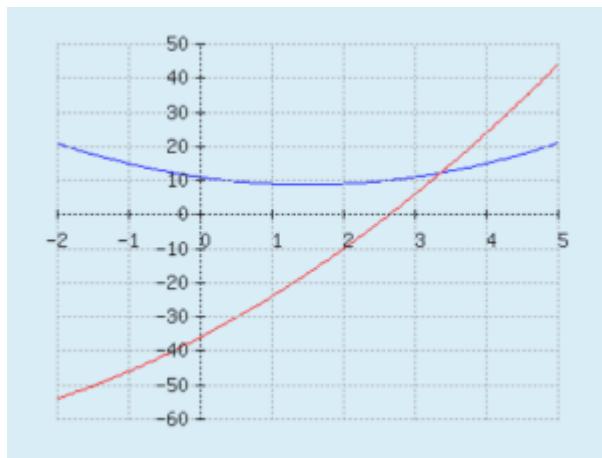
Man sollte dabei auf die Grenzen achten! Hier z.B. muss „d“ kleiner „e“ sein.



Ein Diagramm mit mehreren 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

Beachte: Die Maxima Funktionen plot2d() und plot3d() wurden für STACK aus Sicherheitsgründen in plot() gekapselt und funktionieren somit auch nur ausschließlich als 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](#)

Multiple Choice Stack Aufgaben (MCQ)

Im folgenden Teil wird erzählt wie eine MCQ erstellt wird.

Testkurs Cansiz

1

[Dashboard](#) / [Meine Kurse](#) / [Testkurs Cansiz](#) / Allgemeines / [Moodle \(Stack\) Fragen](#) / Testinhalt bearbeiten

Test bearbeiten: Moodle (Stack) Fragen [?](#)

Fragen: 0 | Aktuell läuft dieser Test

Beste Bewertung 10,00

Speichern

Seitenumbrüche durchführen

Mehrere Elemente auswählen

Summe der Bewertungen: 0,00



Fragen mischen [?](#)

[Hinzufügen](#) [▼](#)

[+ Frage hinzufügen...](#)

[+ aus der Fragensammlung](#)

[+ Zufallsfrage hinzufügen...](#)

Klicken Sie auf „Frage hinzufügen“

2

Zuordnung	STACK ermöglicht es mathematische Fragestellungen in Moodle-Tests zu verwenden. Es bedient sich dabei eines Computeralgebrasystems um mathematische Eigenschaften der eingegebenen Antworten zu ermitteln und diese dann zu bewerten.
<input type="radio"/>	Zuordnung
<input type="radio"/>	Einfach berechnet
<input type="radio"/>	Erweiterter Lückentext
<input type="radio"/>	Kombiniert
<input type="radio"/>	Kprim(ETH)
<input type="radio"/>	Lückentext-Frage
<input type="radio"/>	Lückentextauswahl
<input type="radio"/>	Multiple-Choice (OU)
<input type="radio"/>	Musterabgleich
<input type="radio"/>	PoodLL-Aufnahme
<input checked="" type="radio"/>	STACK
<input type="radio"/>	Zufällige Kurzantwortzuordnung

Wählen Sie „Stack“ aus der Liste aus

Fragestitel

MCQ Test

3

Starte die Frage-Tests...

Aufgabenvariablen

p:sin(2*x);

```

ta1: [[diff(p,x),true],[p,false],[int(p,x),false],[2*(1-2*(sin(x))^2),true]];
ta2:maplist(first,ta1)
ta3:maplist(second,ta1)

```

Frage

?

4

If \((f(x) = \{ @p @ \})\)

\(\left(\frac{d}{dx} f(x) = ? \right)\)

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

Erreichbare Punkte

?

2

▼ Eingabe: ans1

Eingabetyp

?

Musterlösung

ta1

Größe der Eingabebox

?

Strike Syntax

?

Sternchen einfügen

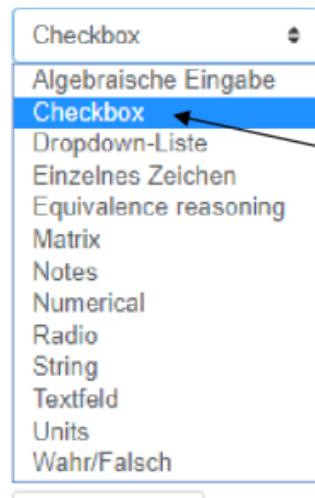
?

Syntax-Hinweis

?

Hint attribute

?



Für diesen Aufgabentyp
wählen wir „Checkbox“
aus

5

▼ Rückmeldebaum (PRT): prt1

6

Aufgabenwert

1

Auto-Vereinfachung

?

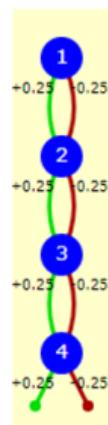
Ja

Feedback-Variablen

?

setans:stify(ans1)

Dieser potenzielle Rückmeldebaum wird aktiv, wenn Teilnehmende folgendes geantwortet hat:
ans1



7

Knoten 1

?

Antwortüberprüfung: AlgEquiv SAns: `elementp(ta2[1],setans)` TAns: `ta3[1]`

Feedback unterdrücken: Nein SAns: `elementp(ta2[1],setans); TAns: ta3[1]`

Knoten 1 wenn WAHR

?

Mod: + Score: 0.25 Abzüge: Nächster: Knoten 2 Antworthinweis: prt1-1-T

Knoten 1 WAHR feedback

?

Mod: + Score: 0.25 Abzüge: Nächster: Knoten 2 Antworthinweis: prt1-1-T

Knoten 1 wenn FALSCH

?

Mod: - Score: 0.25 Abzüge: Nächster: Knoten 2 Antworthinweis: prt1-1-F

Knoten 1 FALSCH Feedback

?

Mod: - Score: 0.25 Abzüge: Nächster: Knoten 2 Antworthinweis: prt1-1-F

8

Knoten 2

Antwortüberprüfung: AlgEquiv SAns: elementp(ta2[2],setans) TAns: ta3[2]
Feedback unterdrücken: Nein SAns: elementp(ta2[2],setans); TAns: ta3[2]

Knoten 2 wenn WAHR

Mod: + Score: 0.25 Abzüge: Nächster: Knoten 3 Antworthinweis: prt1-2-T

Knoten 2 WAHR feedback

Mod: - Score: 0.25 Abzüge: Nächster: Knoten 3 Antworthinweis: prt1-2-F

Knoten 2 wenn FALSCH

Knoten 2 FALSCH Feedback

Lösche Knoten 2

9

Knoten 3

Antwortüberprüfung: AlgEquiv SAns: elementp(ta2[3],setans) TAns: ta3[3]
Feedback unterdrücken: Nein SAns: elementp(ta2[3],setans); TAns: ta3[3]

Knoten 3 wenn WAHR

Mod: + Score: 0.25 Abzüge: Nächster: Knoten 4 Antworthinweis: prt1-3-T

Knoten 3 WAHR feedback

Mod: - Score: 0.25 Abzüge: Nächster: Knoten 4 Antworthinweis: prt1-3-F

Knoten 3 wenn FALSCH

Knoten 3 FALSCH Feedback

Lösche Knoten 3

10

Knoten 4

Antwortüberprüfung: AlgEquiv SAns: elementp(ta2[4],setans) TAns: ta3[4]
Feedback unterdrücken: Nein SAns: elementp(ta2[4],setans); TAns: ta3[4]

Knoten 4 wenn WAHR

Mod: + Score: 0.25 Abzüge: Nächster: [stop] Antworthinweis: prt1-4-T

Knoten 4 WAHR feedback

Mod: - Score: 0.25 Abzüge: Nächster: [stop] Antworthinweis: prt1-4-F

Knoten 4 wenn FALSCH

Knoten 4 FALSCH Feedback

Lösche Knoten 4

Weiteren Knoten hinzufügen

Frage 1
Teilweise richtig
Erreichte Punkte
1,00 von 2,00

If $f(x) = \sin(2x)$

$\frac{d}{dx} f(x) = ?$

11

- $2 \cos(2x)$
- $\sin(2x)$
- $-\frac{\cos(2x)}{2}$
- $2(1 - 2 \sin^2 x)$

Ihre letzte Antwort wurde folgendermaßen interpretiert: $[2 \cos(2x)]$

In Ihrer Antwort wurden die folgenden Variablen gefunden: $[x]$

[[feedback:prt1]]

Eine richtige Antwort ist $[2 \cos(2x), 2(1 - 2 \sin^2 x)]$.

Frage 1
Richtig
Erreichte Punkte
2,00 von 2,00

If $f(x) = \sin(2x)$

$\frac{d}{dx} f(x) = ?$

12

- $2 \cos(2x)$
- $\sin(2x)$
- $-\frac{\cos(2x)}{2}$
- $2(1 - 2 \sin^2 x)$

Ihre letzte Antwort wurde folgendermaßen interpretiert: $[2 \cos(2x), 2(1 - 2 \sin^2 x)]$

In Ihrer Antwort wurden die folgenden Variablen gefunden: $[x]$

[[feedback:prt1]]

Eine richtige Antwort ist $[2 \cos(2x), 2(1 - 2 \sin^2 x)]$.

› Erstellt / Verändert

13

Fix dollars

 Replace $\$...$$ with $\backslash(...\backslash)$, $\$\$...\$$ with $\backslash[...\\]$ and $@...@$ with $\{\@...@\}$ on save. **Änderungen speichern**

Abbrechen

Pflichtfelder Test bearbeiten: Multiple Choice (Presentation) 

Fragen: 1 | Aktuell läuft dieser Test

Seitenumbrüche durchführen

Mehrere Elemente auswählen

Letztendlich Klicken Sie auf
„Änderungen speichern“

Beste Bewertung 10,00

Speichern

Summe der Bewertungen: 1,00



Unsere Frage ist da!

Seite 1

1

MCQ Test If [$f(x)=(@p@)$] [$\frac{d}{dx}f(x)=?$] [[input:ans1]] [[validation:ans1]]1,00 

Hinzufügen

 Fragen mischen 

Hinzufügen

[mitarbeitende], [stuhlbein], [moodle]

Direkt-Link:

https://doku.tu-clausthal.de/doku.php?id=multimedia:moodle:stack_maxima&rev=1576053380

Letzte Aktualisierung: 08:36 11. December 2019

