

# Mathematik

Die Blöcke der Kategorie Mathematik werden genutzt, um Berechnungen anzustellen. Die Ergebnisse der Berechnungen können zum Beispiel als Werte für Variablen verwendet werden. Die meisten Mathematik-Blöcke beziehen sich auf allgemeine mathematische Berechnungen und sollten selbsterklärend sein.

## Blöcke

### Zahlen

Nutze den "Zahl"-Block, um eine beliebige Zahl in dein Programm hinzuzufügen oder einer Variable diese Zahl als Wert zuzuweisen. Dieses Programm weist der Variablen Alter die Zahl 12 zu:



### Einfache Rechnungen

Dieser Block hat die Struktur Wert-Operator-Wert. Als Operatoren stehen die Rechenarten +, -, ÷, × und ^ zur Verfügung. Der Operator kann über das Dropdown-Menü ausgewählt werden. Er kann unmittelbar auf Zahlen oder auch auf Werte von Variablen angewendet werden. Beispiel:



Dieser Block gibt Ergebnis 144 ( $12^2$ ) aus.

### Bitoperationen

Dieser Block ermöglicht es, Bitoperationen zwischen zwei Werten durchzuführen. Der Operator für die Operation kann über ein Dropdown-Menü ausgewählt werden, das eine Reihe von Bitoperationen wie Links-Shift (<<), Rechts-Shift (>>), AND, OR, XOR und NOT anbietet. Diese Operatoren können auf Zahlen sowie auf Variablenwerte angewendet werden. Im gezeigten Beispiel wird die Links-Shift-Operation << verwendet, um die Zahl 10 um 2 Stellen nach links zu verschieben. Das Ergebnis ist 40, da die Bits der Zahl 10 (binär 1010) um zwei Stellen verschoben werden, was zu 101000 binär oder 40 dezimal führt.



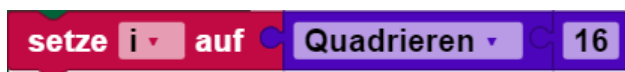
### Spezielle Rechnungen

Dieser Block wendet die, über das Dropdown-Menü ausgewählte Rechenart auf die dahinter platzierte Zahl oder auf den Wert der dahinter platzierten Variable an. Die zur Verfügung stehenden Rechenoperationen sind:

- Quadratwurzel,
- Betrag,
- natürlicher Logarithmus,

- dekadischer Logarithmus,
- Exponentialfunktion mit der Basis  $e$  ( $e^1$ ,  $e^2$ , ...),
- Exponentialfunktion mit der Basis 10 ( $10^1$ ,  $10^2$ , ...),
- Vorzeichenwechsel (Multiplikation mit -1).

$e$  ist hierbei die Euler'sche Zahl. Dieser Block zieht die Quadratwurzel aus 16 und setzt die Variable  $i$  auf das Ergebnis.



## Trigonometrische Funktionen

Dieser Block funktioniert ähnlich wie der zuvor beschriebene Block, mit dem Unterschied, dass als Rechenoperationen die trigonometrischen Funktionen Sinus, Cosinus, Tangens und ihre Umkehrfunktionen genutzt werden. Die angegebene Zahl oder der Wert der angegebenen Variable wird also in die im Dropdown-Menü gewählte Funktion eingesetzt und das Ergebnis kann dann im Programm weiterverarbeitet werden. Zusätzlich gibt es noch den Block "arctan2 of X: ... Y: ...", der es erlaubt, sich mithilfe von zwei reellen Zahlen (einzusetzen als X und Y) einen Funktionswert des arctan2 im Bereich von  $360^\circ$  ausgeben zu lassen.

## Häufig verwendete Konstanten

Dieser Block funktioniert genauso wie der "Zahl"-Block, jedoch gibt man hier den Zahlenwert nicht selbst an. Stattdessen sind häufig verwendete Konstanten (z.B.  $\pi$ ) vorgespeichert. Die Konstante kann über das Dropdown-Menü ausgewählt werden.

## Prüfen auf Eigenschaften

Der Block "... ist ..." wird verwendet, um mathematische Eigenschaften einer gegebenen Zahl zu überprüfen. Im gezeigten Beispiel wird beispielsweise geprüft, ob die Zahl  $x$  eine Primzahl ist.

Eigenschaften, die der Block prüfen kann:

- Gerade: Überprüft, ob die Zahl eine gerade Zahl ist. Eine Zahl ist gerade, wenn sie ohne Rest durch 2 teilbar ist.
- Ungerade: Überprüft, ob die Zahl ungerade ist. Eine Zahl ist ungerade, wenn sie bei der Division durch 2 einen Rest von 1 lässt.
- Eine Primzahl: Wie bereits erwähnt, prüft diese Option, ob die Zahl eine Primzahl ist, das heißt, sie ist nur durch 1 und sich selbst teilbar und größer als 1.
- Eine ganze Zahl: Überprüft, ob die Zahl eine ganze Zahl ist.
- Positiv: Überprüft, ob die Zahl größer als 0 ist.
- Negativ: Überprüft, ob die Zahl kleiner als 0 ist.
- Teilbar durch: Diese Option erlaubt es zu überprüfen, ob die Zahl durch eine andere, spezifizierte Zahl teilbar ist.
- Keine Nummer: Überprüft, ob die Eingabe keine numerische Wertigkeit hat, was nützlich sein kann, um Datentypfehler zu identifizieren.



## Konvertierung

Der Block "konvertiere zu ..." ermöglicht die Umwandlung einer Zahl aus dem Dezimalsystem in eine andere numerische Basis, wie Hexadezimal (hex), Binär (bin), Integer (int), Grad und Radiant. Dieser Vorgang ist besonders nützlich, wenn Daten in unterschiedlichen Darstellungsformen benötigt werden.



## Runden

Mit dem "runde ..." -Block lässt sich eine angegebene Dezimalzahl oder der Wert einer angegebenen Variablen auf eine ganze Zahl runden. Dabei kann man im Dropdown-Menü drei Optionen wählen:

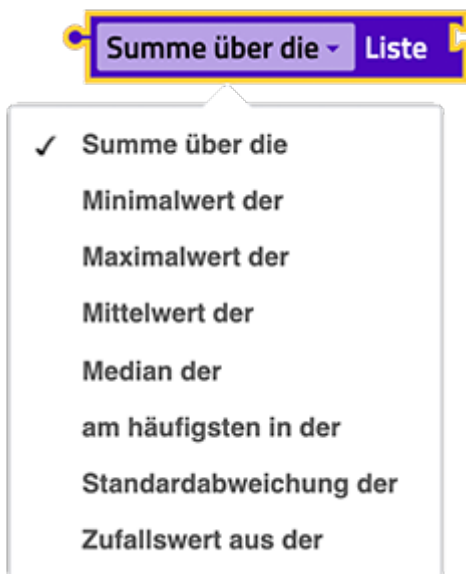
- mit "runde" kaufmännisch gerundet (z.B. wird 4,5 zu 5)
- mit "runde auf" wird aufgerundet (z.B. wird 5,1 zu 6)
- mit "runde ab" wird abgerundet (z.B. wird 5,9 zu 5).

## Auswertung von Listen

Mit dem "der Liste"-Block kann man sich

- mit "Summe" die Summe aller Werte einer Liste,
- mit "min" den kleinsten Wert einer Liste,
- mit "max" den größten Wert einer Liste,
- mit "Mittelwert" den Mittelwert aller Werte einer Liste,
- mit "Median" den Median einer Liste,
- mit "Modalwert" den/die am häufigsten vorkommenden Wert einer Liste,
- mit "Standardabweichung" die Standardabweichung aller Werte einer Liste,
- mit "Zufallswert" einen zufälligen Wert aus einer Liste

ausgeben lassen. Alle diese Optionen können über das Dropdown-Menü des Blocks ausgewählt werden:



## Rest einer Division

Der "Rest von ..." -Block wird genutzt, um den Rest einer Division auszugeben. Dieses Programm weist der Variable "Rest" den Rest der Division von 3:2, also 1, zu:



## Eingabewerte einschränken

Der "beschränke ... von ... bis ..." Block erlaubt es, Eingabewerte auf ein bestimmtes Intervall zu beschränken. Bevor ein Eingabewert weiterverarbeitet wird, wird getestet, ob er im festgelegten Intervall liegt. Es gibt drei Optionen, wie mit einem eingegebenen Wert verfahren wird:

- Der Wert liegt im Intervall, also wird er unverändert weitergegeben.
- Der Wert liegt unter der unteren Grenze des Intervalls, also wird diese untere Grenze weitergegeben.
- Der Wert liegt über der oberen Grenze des Intervalls, also wird diese obere Grenze weitergegeben.

In diesem Beispiel wird der Block genutzt, um den Wert der Variable "Geschwindigkeit" auf die vom Motor unterstützten Drehzahlen einzuschränken:



## Zufällige Werte generieren

Die beiden Blöcke "zufällige Zahl von ... bis ..." und "zufälliger Bruch" geben einen zufälligen Wert aus. Dabei gibt der "zufällige Zahl von ... bis ..." -Block eine Zahl aus dem definierten Intervall aus. Der Block "zufälliger Bruch" gibt hingegen einen Wert zwischen 0,0 (eingeschlossen) und 1,0 (ausgeschlossen) aus.

## Wertebereiche mappen

Der Block "verteile ... von niedrig ... von hoch ... zu niedrig ... zu hoch" nimmt den ersten Wert und transformiert ihn von einem durch die Parameter 2 und 3 definierten Ursprungsbereich auf einen durch die Parameter 4 und 5 festgelegten Zielbereich ab. Dies ermöglicht es, den Eingabewert entsprechend einer linearen Skalierung in den neuen Wertebereich zu überführen.

In dem gezeigten Beispiel des Programmblocks wird der Wert 5 verwendet. Dieser Wert 5 wird von einem definierten Ursprungsbereich von 0 bis 8 in einen Zielbereich von 0 bis 512 transformiert. Die Transformation erfolgt nach einer linearen Skalierungsregel, sodass der ursprüngliche Wert proportional in den neuen Wertebereich übertragen wird. Konkret bedeutet dies, dass der Wert 5 – der etwas über der Mitte des Ursprungsbereichs liegt – in einen entsprechenden Wert im Zielbereich umgerechnet wird. Da 5 genau 62.5 % des Weges von 0 bis 8 darstellt, wird dieser Prozentsatz auf den neuen Bereich von 0 bis 512 angewendet, was zu einem umgerechneten Wert von 320 führt (62.5 % von 512).



Revision #16

Created 21 February 2022 15:50:35 by Admin

Updated 15 July 2024 12:32:50 by phuesing