



# Ameisenbefehlsliste



## DREHEN

Alle Angaben werden als ganze Winkel im Uhrzeigersinn in der Einheit Grad gemacht. 0 entspricht also rechts, 90 entspricht unten, 180 entspricht links, 270 entspricht oben und 360 entspricht wieder rechts. Die übergebenen Werte müssen sich nicht zwischen 0 und 359 befinden. Größere oder kleinere Werte werden automatisch umgerechnet. Das Drehen einer Ameise hat immer Vorrang vor dem Gehen. D.h. wenn eine Ameise einen Dreh-Befehl bekommt, während sie einen Geh-Befehl noch nicht vollständig abgearbeitet hat, wird der Geh-Befehl unterbrochen und nach der Drehung fortgesetzt. Sich gleichzeitig drehen und gehen kann die Ameise nicht.

<b>DreheInRichtung(...)</b>	Die Ameise dreht sich in die angegebene Richtung. Die Drehrichtung wird dabei automatisch bestimmt.
<b>DreheUmWinkel(...)</b>	Die Ameise dreht sich um den angegebenen Winkel. Positive Werte drehen die Ameise nach rechts, negative nach links.
<b>DreheUm()</b>	Die Ameise dreht sich um 180 Grad in die entgegengesetzte Richtung. Hat die selbe Wirkung wie DreheUmWinkel(180).
<b>DreheZuZiel(...)</b>	Die Ameise dreht sich in die Richtung des angegebenen Ziels. Hat die selbe Wirkung wie DreheInRichtung(Koordinate.BestimmeRichtung(...)).

## GEHEN

Die Einheit von Längenangaben bei den folgenden Befehlen ist Schritte. Zum Vergleich: eine Ameise ist vier Schritte lang, das Spielfeld misst im Einzspieler-Modus 1200 x 900 Schritte.

<b>BleibStehen()</b>	Die Ameise bleibt stehen und vergisst ihr aktuelles Ziel. In der nächsten Runde wird das Ereignis Wartet() aufgerufen werden.
<b>GeheGeradeaus()</b> <b>GeheGeradeaus(...)</b>	Die Ameise geht geradeaus. Das Ziel der Ameise bleibt unangetastet. Wenn ein Wert angegeben wird, wird die Ameise wieder ihr Ziel anvisieren, nachdem sie die angegebene Entfernung zurückgelegt hat.
<b>GeheWegVonZiel(...)</b>	Die Ameise dreht sich in die Richtung die vom angegebenen Ziel weg zeigt und geht dann geradeaus. Das Ziel der Ameise bleibt unangetastet und es kann eine Entfernung angegeben werden.
<b>GeheZuZiel(...)</b>	Die Ameise speichert das angegebene Ziel und geht dort hin.
<b>GeheZuBau()</b>	Die Ameise speichert den nächstgelegenen Bau als Ziel und geht dort hin.

## KAMPF

<b>GreifeAn(...)</b>	Die Ameise speichert den angegebenen Käfer als Ziel und geht dort hin. Wenn die Ameise bei dem Käfer angekommen ist, beginnt der Kampf.
----------------------	---

## NAHRUNG

<b>Nimm(...)</b>	Die Ameise nimmt die angegebene Nahrung auf. Bei einem Zuckerhaufen nimmt sie so viel wie möglich weg, bis sie ihre maximale Last erreicht hat (siehe AktuelleLast und MaximaleLast). Im Falle eines Obststückes beginnt die Ameise das Obst zu tragen (siehe GetragenesObst).
<b>LasseNahrungFallen()</b>	Die Ameise lässt die gerade getragene Nahrung fallen. Zucker geht dabei verloren, Äpfel bleiben liegen und können wieder aufgenommen werden. Der Befehl muss nicht ausgeführt werden um Nahrung im Bau abzuliefern. Das passiert dort automatisch.

## MARKIERUNGEN

<b>SprüheMarkierung(...)</b>	Die Ameise sprüht an der aktuellen Stelle eine Duftmarkierung. Mögliche Parameter sind eine in der Markierung hinterlegte Information (diese kann im Ereignis Sieht(markierung) über markierung.Information ausgelesen werden) und die Ausbreitung der Markierung. Je größer die Ausbreitung, desto schneller verschwindet die Markierung wieder.
------------------------------	---

## VOLK-EIGENSCHAFTEN

<b>MaximaleEnergie</b>	Gibt die maximale Energie der Ameise an. Die Einheit ist Lebenspunkte.
<b>MaximaleGeschwindigkeit</b>	Gibt die maximale Geschwindigkeit der Ameise an. Die Einheit ist Schritte pro Runde.
<b>MaximaleLast</b>	Gibt die maximal tragbare Last der Ameise an. Die Einheit ist Nahrungspunkte. Dieser Wert bestimmt, wie viel Zucker die Ameise auf einmal tragen kann und wie schnell sie ohne die Hilfe anderer Ameisen einen Apfel tragen kann.

<b>Reichweite</b>	Gibt die Reichweite in Schritten an die die Ameise zurücklegen kann, bevor sie vor Hunger stirbt. Nachdem die Ameise ein Drittel dieser Strecke zurückgelegt hat, wird das Ereignis WirdMüde() aufgerufen und der Wert von IstMüde auf wahr gesetzt. Siehe ZurückgelegteStrecke.
<b>Angriff</b>	Gibt den Angriffswert der Ameise an. Der Angriffswert bestimmt wie viele Lebenspunkte die Ameise einem Gegner in jeder Runde abzieht. Die Einheit ist Lebenspunkte.
<b>Sichtweite</b>	Gibt den Wahrnehmungsradius der Ameise in Schritten an. Dieser Radius bestimmt wie weit die Ameise von Spielelementen wie z.B. Zucker entfernt sein muss damit die Ameise sie sieht. Die Blickrichtung der Ameise spielt dabei keine Rolle.
<b>Drehgeschwindigkeit</b>	Gibt die Geschwindigkeit an mit der sich eine Ameise drehen kann. Die Einheit ist Grad pro Runde.

## AMEISEN-EIGENSCHAFTEN

<b>AktuelleEnergie</b>	Gibt die aktuelle Energie der Ameise an. Die Einheit ist Lebenspunkte. Hat die Ameise 0 Lebenspunkte oder weniger, dann stirbt sie. Dieser Wert ist immer kleiner oder gleich MaximaleEnergie.
<b>AktuelleGeschwindigkeit</b>	Gibt die aktuell mögliche Geschwindigkeit der Ameise an. Die Einheit ist Schritte pro Runde. Der Wert wird von der aktuellen Last der Ameise beeinflusst. Ameisen die unter voller Last bewegt werden, können nur die Hälfte ihrer Maximalgeschwindigkeit erreichen. Diese Eigenschaft liefert immer einen Wert größer 0 zurück, auch wenn die Ameise still steht. Dieser Wert ist immer kleiner oder gleich MaximaleGeschwindigkeit.
<b>AktuelleLast</b>	Gibt die aktuelle Last an, die die Ameise gerade trägt. Die Einheit ist Nahrungspunkte. Dieser Wert ist immer kleiner oder gleich MaximaleLast.
<b>AnzahlInSichtweite</b>	Gibt die Anzahl der befreundeten Ameisen im Wahrnehmungsradius der Ameise zurück. Das Ergebnis dieser Berechnung ist von der Sichtweite der Ameise abhängig.
<b>AnzahlKäferInSichtweite</b>	Gibt die Anzahl der Käfer im Wahrnehmungsradius WR der Ameise zurück.
<b>AnzahlAmeisenInSichtweite:</b>	Gibt die Anzahl der feindlichen Ameisen im WR der Ameise zurück.
<b>AnzahlVolkInSichtweite:</b>	Gibt die Anzahl der befreundeten Ameisen im WR der Ameise zurück.
<b>AnzahlTypInSichtweite:</b>	Gibt die Anzahl der befreundeten Ameisen des selben Typs im WR der Ameise zurück.
<b>EntfernungZuBau</b>	Gibt die Entfernung in Schritten zum nächstgelegenen befreundeten Ameisenbau an.
<b>GetragenesObst</b>	Gibt das aktuell getragene Obststück zurück. Wenn die Ameise gerade kein Obst trägt, zeigt dieser Verweis ins Leere.
<b>Typ</b>	Gibt des Namen des Typs der Ameise zurück.
<b>TypIndex</b>	Gibt des Index des Typs der Ameise in der Liste aller Ameisentypen zurück. Diese Eigenschaft kann anstelle von Typ verwendet werden, wenn kein langsamer Vergleich von Zeichenketten gemacht werden soll.
<b>Ziel</b>	Gibt das aktuelle Ziel der Ameise zurück. Wenn die Ameise gerade kein Ziel hat, zeigt dieser Verweis ins Leere.
<b>IstMüde</b>	Gibt an ob die Ameise müde ist. Die Ameise wird müde, sobald sie ein Drittel ihrer maximalen Reichweite zurückgelegt hat. Nach dem Übergang des Wertes dieser Eigenschaft von falsch auf wahr wird das Ereignis WirdMüde() aufgerufen.
<b>RestStrecke</b>	Gibt an wie viele Schritte die Ameise noch geradeaus gehen wird, bevor sie wieder ihr Ziel anvisiert. Dieser Wert wird in jeder Runde um AktuelleGeschwindigkeit verringert.
<b>RestWinkel</b>	Gibt an wie viele Grad die Ameise sich noch drehen wird, bevor sie wieder geradeaus gehen wird. Dieser Wert wird in jeder Runde um DrehGeschwindigkeit verringert.
<b>Richtung</b>	Gibt die Blickrichtung der Ameise auf dem Spielfeld an.
<b>Angekommen</b>	Gibt an ob die Ameise an ihrem Ziel angekommen ist.
<b>ZurückgelegteStrecke</b>	Diese Eigenschaft gibt die Gesamtanzahl an Schritten zurück die die Ameise seit ihrem letzten Besuch in einem Ameisenbau zurückgelegt hat. Siehe Reichweite

## HILFSBEFEHLE FÜR NAHRUNGS-AUFNAHME

**BrauchtNochTräger(...)** Ermittelt ob das angegebene Obst noch mehr Ameisen zum Tragen benötigt.

## HILFSBEFEHLE ZUR KOORDINATION

Um in den folgenden Befehlen Bezug auf die aktuelle Ameise zu nehmen, wird in C# das Schlüsselwort this verwendet, in Visual Basic das Schlüsselwort Me.

<b>Koordinate. BestimmeEntfernung(...)</b>	Bestimmt die Entfernung in Schritten zwischen zwei angegebenen Spielelementen.
<b>Koordinate. BestimmeRichtung(...)</b>	Bestimmt vom ersten angegebenen Spielelement aus die Blickrichtung zum zweiten angegebenen Spielelement.

## HILFSBEFEHLE FÜR ZUFALLSZAHLN

**Zufall.Zahl(...)** Erzeugt eine zufällige Zahl zwischen den angegebenen Grenzen. Wenn nur ein Parameter angegeben wird, wird eine Zahl zwischen 0 und der angegebenen Grenze - 1 bestimmt, wenn zwei Parameter angegeben werden, wird eine Zahl zwischen der unteren Grenze und der oberen Grenze - 1 bestimmt.