

# Tabellenfunktionen in Excel 2003

aus : **Helmut Vonhoegen Excel 2003 professionell anwenden – Galileo Press**

Excel 2003 wartet mit einer überwältigenden Fülle von Funktionen auf, die aus einem oder mehreren Werten neue Werte ermitteln, und das teilweise auf eine recht komplexe Weise. Dabei werden auch Funktionen sein, die, wenn sie nicht im Dialog *Funktion einfügen* aufgelistet sind, erst über *Extras/ Add-Ins* geladen werden müssen, damit sie Ihnen zur Verfügung stehen. Diese Funktionen werden dort als Analyse-Funktionen angeboten.

Eine Funktion ist eine Art von „Black box“, bei der auf der einen Seite etwas eingegeben wird und auf der anderen Seite etwas herauskommt. Ohne dass der Anwender das oft sehr komplizierte Formelwerk für den Betrieb der Funktion kennen muß, wird ihm nach Eingabe der notwendigen Werte das Ergebnis der Operationen ausgegeben.

Jede Funktion liefert ein Ergebnis. Je nach Funktion kann dieses Ergebnis numerisch sein, also ein bestimmtes Rechenergebnis, eine Zeichenfolge, ein Text oder ein Wahrheitswert (WAHR oder FALSCH). Zuweilen ist das Ergebnis nicht nur ein Wert, sondern eine Matrix (Array) von Werten. Der Typ des Ergebnisses einer Funktion muß berücksichtigt werden, wenn in einer weiteren Formel auf diese Funktion Bezug genommen wird. Ansonsten kommt es eventuell zu einem Fehler, wenn die Datentypen nicht zusammenpassen.

## Finanzmathematische Funktionen

Neben einer Anzahl von Funktionen, die sich mit allgemeinen finanzmathematischen Themen beschäftigen, fallen vor allem drei große Gruppen von Funktionen ins Gewicht: Rentenberechnungen, Abschreibung und Wertpapiere.

Bei Renten geht es natürlich nicht um die sozialrechtlichen Versorgungsrenten, sondern um privatwirtschaftlich vereinbarte regelmäßige Zahlungen. Die einfachste und für das Verständnis dieser Funktionen durchschaubarste Form hierfür ist folgendes Modell:

Sie zahlen auf der Bank eine bestimmte Summe an Geld ein, die dort nach einem vereinbarten Zinssatz verzinst wird. Die Bank zahlt Ihnen aus diesem Guthaben regelmäßig einen bestimmten Betrag („Rente“), so lange bis das Guthaben verbraucht ist.

Eine Anzahl von Argumenten taucht bei diesen Funktionen immer wieder auf, so dass es vernünftig scheint, diese zunächst im Zusammenhang zu besprechen.

<b>Funktionsargument</b>	<b>Erklärung</b>
Bw [Barwert]:	Der gegenwärtige rechnerische Wert einer Reihe regelmäßiger, gleichbleibender Zahlungen, z. B. einer Rente.
Rmz [Ratenzahlung]:	Regelmäßig zu leistende oder zu erhaltende Zahlungen.
Zins [Zinssatz]:	Das ist der Zinssatz für einen Zahlungszeitraum. Er wird dezimal (z. B. 0,08) oder als % (z. B. 8 %) angegeben.
Zr [Zeitraum]:	Die einzelne Zahlungsperiode im Gesamtzeitraum.
Zw [Zukunftswert]:	Der zukünftige Wert einer Investition. Wird Zw als optionales Argument verwandt, dann wird es von Excel bei Nichteintrag mit 0 gesetzt.
Zzr [Zahlungszeiträume]:	Anzahl der Zahlungsperioden. Es ist darauf zu achten, dass die Zzr mit den anderen Argumenten in den Einheiten übereinstimmen, d. h., dass einheitlich Jahre, Monate oder Tage in einer Funktion verwandt werden. Werden z. B. Monate verwendet, dann muß auch der

	<p>Monatszins (Jahreszins/12) eingesetzt werden.</p> <p>Beachten Sie hierbei bitte, dass in Deutschland das Zinsjahr in 12 Zinsmonate mit jeweils 30 Tagen eingeteilt wird. In anderen Ländern gelten teilweise andere Regelungen.</p>
F [Fälligkeit]:	<p>Das Argument F für Fälligkeit legt fest, ob die Berechnung für vorschüssige oder nachschüssige Rentenzahlungen etc. erfolgen soll. Wird F nicht angegeben oder mit 0, dann wird die Funktion als nachschüssig berechnet, die Zahlungen erfolgen also am Ende der Periode, wird F mit 1 angegeben, so erfolgt die Berechnung für eine vorschüssige Rente.</p>

## Referenz der finanzmathematischen Funktionen

### AMORDEGRK()

**Syntax:** AMORDEGRK(Kosten; Datum; Erste\_Periode; Restwert; Periode; Rate; Basis)

Liefert den Abschreibungsbetrag nach dem französischen Buchhaltungssystem. Die Argumente sind: Kosten der Anschaffung, Datum der Anschaffung, Ende der ersten Periode, Restwert, die Periode, für die der Betrag errechnet werden soll, Abschreibungsrate und Zeitbasis. Die Funktion ist der Funktion AMORLINEARK ähnlich, arbeitet aber zusätzlich mit einem Abschreibungskoeffizienten, der von der Nutzungsdauer abhängig ist.

### AMORLINEARK()

**Syntax:** AMORLINEARK(Kosten; Datum; Erste\_Periode; Restwert; Periode; Rate; Basis)

Liefert den Abschreibungsbetrag nach dem französischen Buchhaltungssystem. Die Argumente sind: Kosten der Anschaffung, Datum der Anschaffung, Ende der ersten Periode, Restwert, die Periode, für die der Betrag errechnet werden soll, Abschreibungsrate und Zeitbasis.

### AUFGELZINS()

**Syntax:**

AUFGELZINS(Emission; Erster\_Zinstermin; Abrechnung; Nominalzins; Nennwert; Häufigkeit; Basis)

Liefert für Wertpapiere die Summe der aufgelaufenen Zinsen für einen bestimmten Zeitraum bei periodischen Zahlungen. Dient dazu, den Tageswert von festverzinslichen Papieren zu kalkulieren. Das Verfahren ist in Europa nicht üblich.

Emission ist das Datum der Ausgabe des Wertpapiers (ab diesem Termin wird das Papier verzinst). Erster\_Zinstermin ist der Termin, bis zu dem die Verzinsung gültig ist. Mit Abrechnung wird der Stichtag angegeben, für den die aufgelaufenen Zinsen berechnet werden sollen.

Nominalzins ist der Zinssatz des Wertpapiers und Nennwert der Nennwert (ohne Angabe wird 1000 angesetzt). Mit Häufigkeit wird angegeben, wie oft die Zinsen im Jahr fällig werden (1, 2 oder 4). Basis gibt an, nach welchem Zeitsystem die Zinstage berechnet werden. Vgl. hierzu den Vorspann zu den Finanzfunktionen.

### AUFGELZINSF()

**Syntax:** AUFGELZINSF(Emission; Abrechnung; Nominalzins; Nennwert; Basis)

Liefert für Wertpapiere die Summe der aufgelaufenen Zinsen für einen bestimmten Zeitraum bei einmaliger Zahlung zum Abrechnungstermin. Dient dazu, den Tageswert von festverzinslichen Papieren zu kalkulieren. Das Verfahren ist in Europa nicht üblich.

Emission ist das Datum der Ausgabe des Wertpapiers (ab diesem Termin wird das Papier verzinst). Mit Abrechnung wird der Stichtag angegeben, für den die aufgelaufenen Zinsen berechnet werden sollen.

Nominalzins ist der Zinssatz des Wertpapiers und Nennwert der Nennwert (ohne Angabe wird 1000 angesetzt). Basis gibt an, nach welchem Zeitsystem die Zinstage berechnet werden. Vgl. hierzu den Vorspann zu den Finanzfunktionen.

## AUSZAHLUNG()

**Syntax:** AUSZAHLUNG(Abrechnung; Fälligkeit; Anlage; Disagio; Basis)

Liefert den Auszahlungsbetrag, den Sie für eine Anlage in festverzinslichen Wertpapieren zu einem bestimmten Zeitpunkt erhalten.

Mit Abrechnung wird der Zeitpunkt des Kaufs angegeben, mit Fälligkeit der Zeitpunkt, zu dem der Auszahlungsbetrag berechnet werden soll. Anlage ist die Summe, die beim Kauf investiert wird. (Die Funktion setzt voraus, dass der Gesamtbetrag in die Wertpapiere investiert wird.) Disagio ist der prozentuale Abschlag, den Sie bei der Ausgabe erhalten (bei 5 % Disagio z. B. wird ein Wertpapier mit einem Nominalwert von 100 EURO für 95 EURO verkauft). Basis gibt an, nach welchem Zeitsystem die Zinstage berechnet werden. Vgl. hierzu den Vorspann zu den Finanzfunktionen.

## BW()

**Syntax:** BW(Zins; Zzr; Rmz; Zw; F)

Die Funktion berechnet den Barwert einer Reihe von regelmäßigen Zahlungen.

Mit Zins wird der Zins und mit Zzr die Zahlungszeiträume (oder die Anzahl der Zahlungen) angegeben.

Außer dem Betrag, der periodisch zu zahlen ist (Rmz oder auch die Ratenzahlung), können noch zwei optionale Argumente angegeben werden. Zw bedeutet den Zukunftswert, der nach der letzten Zahlung erreicht werden soll (im obigen Beispiel, wieviel Geld nach 20 Jahren auf der Bank übrig sein soll). Wird Zw nicht angegeben, so wird er als 0 angenommen.

Mit F wird angegeben, ob die Zahlungen jeweils am Ende einer Periode erfolgen ( $F = 0$  oder weggelassen) oder am Anfang ( $F = 1$ ).

Grundlage der Funktion BW() ist die Formel

$$\text{Barwert} = \text{Rate} \cdot (q^{\wedge} \text{Zzr} - 1) / ((q^{\wedge} \text{Zzr}) \cdot (q - 1))$$

wobei mit  $q$  der Zinsfaktor gemeint ist ( $q = 1 + (\text{Zins}/100)$ ).

## DIA()

**Syntax:** DIA(Ansch\_Wert; Restwert; Nutzungsdauer; Zr)

Die Funktion berechnet die Abschreibungsrate für einen bestimmten Abschreibungszeitraum nach der digitalen Methode.

Das Argument Ansch\_Wert ist der Betrag, der für die Investition aufgewandt wurde.

Restwert ist der Wert, den die Investition nach der Abschreibung noch hat.

Nutzungsdauer entspricht der Anzahl von Zeiträumen bis zum Ende der Abschreibung. Zr ist der Zeitraum, für den die Abschreibung errechnet werden soll.

## DISAGIO()

**Syntax:** DISAGIO(Abrechnung; Fälligkeit; Kurs; Rückzahlung; Basis)

Berechnet das Disagio (Abschlag beim Handel mit einem Wertpapier).

Abrechnung ist der Kauf- bzw. Emissionstermin, Fälligkeit der Termin des Verkaufs. Kurs ist der Wert zum Kauftermin und Rückzahlung der Wert zum Verkaufstermin. Basis gibt an, nach welchem Zeitsystem die Zinstage berechnet werden. Vgl. hierzu den Vorspann zu den Finanzfunktionen.

## DURATION()

**Syntax:** DURATION(Abrechnung; Fälligkeit; Nominalzins; Rendite; Häufigkeit; Basis)

Liefert die hypothetische Anlagedauer eines festverzinslichen Wertpapiers in Jahren.

Je niedriger die hypothetische Anlagedauer im Vergleich zur tatsächlichen ist, um so profitabler ist das Wertpapier.

Abrechnung ist der Kauf- oder Emissionstermin, Fälligkeit der Verkaufstermin. Mit Nominalzins wird der Zinssatz angegeben, mit Rendite der tatsächlich erzielte Zinssatz (setzt sich zusammen aus dem Nominalzins und dem Gewinn aus der Differenz von Kauf- und Verkaufspreis). Häufigkeit ist die Zahl der Zinszahlungen im Jahr (1, 2, oder 4).

Basis gibt an, nach welchem Zeitsystem die Zinstage berechnet werden. Vgl. hierzu den Vorspann zu den Finanzfunktionen.

## **EFFEKTIV()**

**Syntax:** EFFEKTIV(Nominalzins; Perioden)

Liefert den effektiven Jahreszins für eine Anlage oder ein Darlehen.

Nominalzins ist der jährliche Nominalzins, Perioden die Zahl der jährlichen Zinszahlungen.

Da bei mehreren jährlichen anteiligen Zinszahlen ein Teil der Jahreszinsen praktisch vorweg gezahlt wird, erhöht sich der Effektivzins mit der Zahl der jährlichen Zinszahlungen.

## **GDA()**

**Syntax:** GDA(Ansch\_Wert; Restwert; Nutzungsdauer; Periode; Faktor)

Die Funktion berechnet die Abschreibungsrate für einen bestimmten Abschreibungszeitraum nach der geometrisch-degressiven Methode.

Das Argument Ansch\_Wert ist der Betrag, der für die Investition aufgewandt wurde.

Restwert ist der Wert, den die Investition nach der Abschreibung noch hat.

Nutzungsdauer entspricht der Anzahl von Zeiträumen bis zum Ende der Abschreibung.

Periode ist der Zeitraum, für den die Abschreibung ermittelt werden soll. Mit Faktor wird die Stärke der Degressivität angegeben (ohne Angabe wird 2 angesetzt).

Der Buchwert kann mit dieser Methode nie auf 0 gehen, deshalb wird in der Praxis für spätere Perioden auf die lineare Abschreibung übergegangen.

## **GDA2()**

**Syntax:** GDA2(Ansch\_Wert; Restwert; Nutzungsdauer; Periode; Monate)

Die Funktion berechnet die Abschreibungsrate für einen bestimmten Abschreibungszeitraum nach der geometrisch-degressiven Methode (vgl. GDA).

Das Argument Ansch\_Wert ist der Betrag, der für die Investition aufgewandt wurde.

Restwert ist der Wert, den die Investition nach der Abschreibung noch hat.

Nutzungsdauer entspricht der Anzahl von Zeiträumen bis zum Ende der Abschreibung.

Periode ist der Zeitraum, für den der Abschreibungsbetrag ermittelt werden soll. Wird das optionale Argument Monate weggelassen, so wird es von Excel als 12 angenommen. Die Funktion liefert dann denselben Wert wie GDA(). Tragen Sie für Monate einen anderen Wert als 12 ein, so wird die Abschreibung für die Anzahl der Monate im entsprechenden Jahr berechnet.

## **IKV()**

**Syntax:** IKV(Werte; Schätzwert)

Die Funktion berechnet den internen Ertragszins einer Investition. Dadurch kann die Rentabilität einer Investition geprüft werden.

Das Argument Werte ist eine Matrix oder ein Bereich, der mindestens einen positiven und einen negativen Wert enthält. Der erste Wert stellt die getätigte Investition als Ausgabe dar, erscheint also negativ, die weiteren Zellen stellen die periodischen Rückflüsse dar.

Das optionale Argument Schätzwert sollte mit einem Wert belegt werden, der in einer Schätzung dem erwarteten Ergebnis schon möglichst nahe kommt. Wird er nicht eingetragen, so wird sein Wert von Excel mit 10 % angenommen.

## **ISPMT()**

**Syntax:** ISPMT(Zins; Zr; Zzr; Bw)

Berechnet die Zinsen, die während einer bestimmten Periode gezahlt werden.

## **KAPZ()**

**Syntax:** KAPZ(Zins; Zr; Zzr; Bw; Zw; F)

Die Funktion berechnet den Tilgungsanteil für die Abzahlung eines Darlehens.

Das Darlehen wird in periodischen, gleichbleibenden Raten abbezahlt, deren Höhe mit RMZ() berechnet wird. Die Raten bestehen jeweils aus einem Zinsanteil (berechnet mit ZINSZ) und einem Tilgungsanteil, um den sich die Gesamtschuld jeweils verringert.

Argumente sind Zins (der konstante Zinssatz), Zr (der Zeitraum, für den die Zahlung berechnet werden soll), Zzr (die Anzahl der Abzahlungszeiträume) und Bw (der Barwert, bei einem Darlehen der Auszahlungsbetrag).

Als optionalen Wert außer F (Fälligkeit; 0 = nachschüssig, 1 = vorschüssig) können Sie noch Zw angeben (den Zukunftswert oder den Betrag, der am Ende der letzten Zahlung vorhanden sein soll).

## **KUMKAPITAL()**

**Syntax:** KUMKAPITAL(Zins; Zzr; Bw; Zeitraum\_Anfang; Zeitraum\_Ende; F)

Berechnet den Betrag, der für die Tilgung eines Darlehens in einem bestimmten Zeitraum aufgebracht wird (nicht zu verwechseln mit den Zahlungsbeträgen).

Mit Zins wird der Zinssatz pro Zinszeitraum angegeben; mit Zzr die Anzahl der Zahlungsperioden, mit Bw der Barwert (ausgegebene Betrag) des Darlehens. Zeitraum\_Anfang und Zeitraum\_Ende sind die Nummern der beiden Zahlungsperioden, zwischen denen die Gesamttilgung berechnet werden soll. Mit F läßt sich angeben, ob die Zahlungen vorschüssig (1) oder nachschüssig (0 oder weggelassen) erfolgen sollen.

Während mit KUMKAPITAL der Tilgungsanteil berechnet wird, liefert die folgende Funktion KUMZINSZ den Zinsanteil für den betreffenden Zeitraum. Beides zusammen ist der Betrag, der für den betreffenden Zeitraum insgesamt aufgebracht werden muß.

Analog verhalten sich die Funktionen KAPZ() und ZINSZ(), beide zusammen ergeben die RMZ().

## **KUMZINSZ()**

**Syntax:** KUMZINSZ(Zins; Zzr; Bw; Zeitraum\_Anfang; Zeitraum\_Ende; F)

Berechnet den Betrag, der für die Zinsen eines Darlehens in einem bestimmten Zeitraum aufgebracht wird (nicht zu verwechseln mit den Zahlungsbeträgen).

Die Argumente der Funktion sind identisch mit denen von KUMKAPITAL (siehe dort). Dort wird auch das Verhältnis der beiden Funktionen zueinander dargelegt.

## **KURS()**

**Syntax:** KURS(Abrechnung; Fälligkeit; Zins; Rendite; Rückzahlung; Häufigkeit; Basis)

Liefert den Kurswert eines festverzinslichen Wertpapiers mit dem Nennwert 100 EURO abhängig von der anvisierten Rendite.

Abrechnung ist das Verkaufsdatum; Fälligkeit das Fälligkeitsdatum des Wertpapiers. Mit Zins wird der Nominalzins angegeben, mit Rendite die Rendite, die das Papier bringen soll. Rückzahlung ist der Betrag, zu dem das Wertpapier zum Fälligkeitstermin ausbezahlt wird (bei Anleihen im allgemeinen 100). Häufigkeit gibt an, wie oft die Zinsen im Jahr ausgeschüttet werden. Basis gibt an, nach welchem Zeitsystem die Zinstage berechnet werden. Vgl. hierzu den Vorspann zu den Finanzfunktionen.

## **KURSDISAGIO()**

**Syntax:** KURSDISAGIO(Abrechnung; Fälligkeit; Disagio; Rückzahlung; Basis)

Berechnet den Ausgabekurs eines nichtverzinslichen Wertpapiers mit einem Nennwert von 100 EURO.

Abrechnung ist der Ausgabetermin, Fälligkeit der Rücknahmetermin. Rückzahlung ist der Wert des Papiers zum Fälligkeitstermin. Disagio legt fest, mit welchem Abschlag (aufs Jahr gerechnet) das Papier verkauft werden soll. Basis gibt an, nach welchem Zeitsystem die Zinstage berechnet werden. Vgl. hierzu den Vorspann zu den Finanzfunktionen.

## **KURSFÄLLIG()**

**Syntax:** KURSFÄLLIG(Abrechnung; Fälligkeit; Emission; Zins; Rendite; Basis)

Liefert den Kurswert eines festverzinslichen Wertpapiers (Nennwert 100 EURO), bei dem die Zinsen zum Fälligkeitsdatum (mit der Rückzahlung) ausgezahlt werden.

Abrechnung ist das Datum, zu dem der Kurswert errechnet werden soll, Fälligkeit das Datum, zu dem das Wertpapier zurückgenommen wird. Emission ist das Ausgabedatum, ab dem die Zinsen laufen; Zins der jährliche Zins.

Mit Rendite wird angegeben, wieviel das Wertpapier jährlich einbringen soll (in Prozent vom Kaufpreis). Basis gibt an, nach welchem Zeitsystem die Zinstage berechnet werden. Vgl. hierzu den Vorspann zu den Finanzfunktionen.

## **LIA()**

**Syntax:** LIA(Ansch\_Wert; Restwert; Nutzungsdauer)

Die Funktion berechnet die Abschreibungsrate für einen bestimmten Abschreibungszeitraum nach der linearen Methode (AFA).

Bei dieser Methode wird über den gesamten Abschreibungszeitraum mit periodisch gleichbleibenden Beträgen abgeschrieben.

Das Argument Ansch\_Wert ist der Betrag, der für die Investition aufgewandt wurde. Restwert ist der Wert, den die Investition nach der Abschreibung noch hat. Nutzungsdauer entspricht der Anzahl von Zeiträumen bis zum Ende der Abschreibung.

## **MDURATION()**

**Syntax:** MDURATION(Abrechnung; Fälligkeit; Nominalzins; Rendite; Häufigkeit; Basis)

Die Funktion berechnet den Nettokapitalwert periodischer Cashflow (Ein- oder Auszahlungen).

Mit Zins wird ein konstanter Zinssatz für den gesamten betrachteten Zeitraum angegeben. Wert1, Wert2, ... sind Zahlungen (Einzahlung positiv, Auszahlung negativ), die jeweils am Ende einer Periode erfolgen.

Ist das Ergebnis negativ, dann wäre es besser gewesen, das Geld zum angegebenen Zinssatz auf die Bank zu tragen.

## **NOMINAL()**

**Syntax:** NOMINAL(Effektiver\_Zins; Perioden)

Berechnet den Nominalzins (Jahreszins auf den Nennwert). Die Funktion ist die Umkehrung der Funktion EFFEKTIV().

Mit Effektiver\_Zins wird der tatsächlich erzielte Jahreszins angegeben; Perioden ist die Zahl der jährlichen Zinszahlungen. Bei einer jährlichen Zahlung sind Nominalzins und Effektivzins identisch.

## **NOTIERUNGBRU()**

**Syntax:** NOTIERUNGBRU(Zahl; Teiler)

Liefert einen als Bruch interpretierbaren Ausdruck einer Dezimalzahl.

Zahl ist die umzuwandelnde Zahl; Teiler der Nenner des Bruchs. Ist Teiler wie im Beispiel 5, dann ist die Nachkommastelle des Ergebnisses zu lesen als 1/5.

## **NOTIERUNGDEZ()**

**Syntax:** NOTIERUNGDEZ(Zahl; Teiler)

Konvertiert einen als Bruch interpretierten Ausdruck (ZAHL) in eine Dezimalzahl. Die Funktion ist die Umkehrung zu NOTIERUNGBRU() (siehe dort).

## **QIKV()**

**Syntax:** QIKV(Werte; Investition; Reinvestition)

Mit dieser Funktion bestimmen Sie die interne Ertragsrate einer Reihe von Cashflow (Ein- und Auszahlungen).

Drei Argumente müssen für die Funktion eingetragen werden:

Werte ist ein Bezug oder eine Reihe von Zellen, die die Werte von Abgängen und Eingängen enthalten. Es muß mindestens ein negativer und ein positiver Wert vorhanden sein (positiv für Eingänge); der erste Betrag wird zumeist der Investition entsprechen und somit ein negativer Wert sein.

Investition ist der Zinssatz für die Einzahlungen; Reinvestition ist der Zinssatz für reinvestierte Beträge.

## **RENDITE()**

**Syntax:** RENDITE(Abrechnung; Fälligkeit; Zins; Kurs; Rückzahlung; Häufigkeit; Basis)

Berechnet die jährliche Rendite eines festverzinslichen Wertpapiers.

Abrechnung ist der Kauftermin, Fälligkeit der Rücknahme- und damit Auszahlungstermin. Zins ist der Nominalzins. Das Papier wurde zum Preis von Kurs erworben und wird zum Fälligkeitstermin zum Preis von Rückzahlung eingelöst. Häufigkeit ist die Anzahl der jährlichen Zinsausschüttungen. Basis gibt an, nach welchem Zeitsystem die Zinstage berechnet werden. Vgl. hierzu den Vorspann zu den Finanzfunktionen.

Liefert die Rendite eines Wertpapiers, das periodisch Zinsen auszahlt. Mit RENDITE können Sie die Rendite von Anleihen und Obligationen berechnen.

## **RENDITEDIS()**

**Syntax:** RENDITEDIS(Abrechnung; Fälligkeit; Kurs; Rückzahlung; Basis)

Berechnet die jährliche Rendite eines unverzinslichen Wertpapiers.  
Zur Interpretation der Argumente vgl. RENDITE().

## **RENDITEFÄLL()**

**Syntax:** RENDITEFÄLL(Abrechnung; Fälligkeit; Emission; Zins; Kurs; Basis)

Berechnet die jährliche Rendite eines festverzinslichen Wertpapiers, dessen Zinsen zusammen mit der Rückzahlung zum Fälligkeitstermin ausgezahlt werden.

Abrechnung ist der Kauftermin, Fälligkeit der Rücknahme- und damit Auszahlungstermin, Emission der Ausgabetermin, ab dem die Verzinsung läuft. Zins ist der Nominalzins. Das Papier wurde zum Preis von Kurs am Abrechnungstermin erworben und wird zum Fälligkeitstermin zum Preis von Rückzahlung zusammen mit den aufgelaufenen Zinsen eingelöst. Basis gibt an, nach welchem Zeitsystem die Zinstage berechnet werden. Vgl. hierzu den Vorspann zu den Finanzfunktionen.

## **RMZ()**

**Syntax:** RMZ(Zins; Zzr; Bw; Zw; F)

Die Funktion liefert die kontinuierlichen Raten für eine Investition. Argumente sind Zins, Zzr (die Zahlungszeiträume) und Bw (der Barwert, die Investition). Optional kann außer F (Fälligkeit; 0 für Ende der Perioden, 1 für Anfang) noch Zw (Zukunftswert) eingegeben werden. Vgl. KAPZ().

## **TBILLÄQUIV()**

**Syntax:** TBILLÄQUIV(Abrechnung; Fälligkeit; Disagio)

Berechnet die vergleichsweise jährliche Verzinsung eines Schatzwechsels (Treasury Bill). Ein Schatzwechsel ist eine kurzfristige Schuldverschreibung, die zum Termin Abrechnung mit einem Abschlag von Disagio (in Prozent auf ein Jahr umgerechnet) erworben wird und zum Terminfälligkeit zum vollen Wert eingelöst wird. Abrechnung und Fälligkeit müssen im gleichen Jahr liegen.

Da der Disagio praktisch ein vorschüssiger Zins ist, rechnet die Funktion um, wie der Disagio sich in einem nachschüssigen Zins widerspiegeln würde.

## **TBILLKURS()**

**Syntax:** TBILLKURS(Abrechnung; Fälligkeit; Disagio)

Liefert den Ausgabekurs für einen Schatzwechsel, der einen Nominalwert (und damit Rückgabewert) von 100 EURO hat.

Zu den Argumenten vgl. TBILLÄQUIV().

## **TBILLRENDITE()**

**Syntax:** TBILLRENDITE(Abrechnung; Fälligkeit; Kurs)

Liefert die Rendite eines Schatzwechsels, der zu einem Ausgabekurs von Kurs erworben wurde.

Zu den Argumenten vgl. TBILLÄQUIV().

## **UNREGER.KURS()**

**Syntax:** UNREGER.KURS(Abrechnung; Fälligkeit; Emission; Erster\_Zinstermin; Zins; Rendite; Rückzahlung; Häufigkeit; Basis)

Die Funktion berechnet die Abschreibungsrate für einen bestimmten Abschreibungszeitraum nach der variabel-degressiven Methode.

Diese Funktion ist eine Variation der geometrisch-degressiven Abschreibung. Sie ermöglicht es, sobald die lineare Abschreibung höhere Abschreibungsbeträge liefert, in diese überzuwechseln.

Das Argument Ansch\_Wert ist der Betrag, der für die Investition aufgewandt wurde. Restwert ist der Wert, den die Investition nach der Abschreibung noch hat. Nutzungsdauer entspricht der Anzahl von Zeiträumen bis zum Ende der Abschreibung. Mit Faktor wird die Stärke der Degressivität angegeben (ohne Angabe wird 2 angesetzt). Anfang ist der Anfang der Periode, für die die Abschreibung berechnet werden soll, Ende der Endzeitpunkt. Soll z. B. die Abschreibung für das erste Jahr berechnet werden, dann

wären 0 und 1 einzusetzen. Die Abschreibung kann so auch über mehrere Perioden berechnet werden.

Nicht\_wechseln wird durch einen Wahrheitswert geschaltet. Wenn die nach der linearen Methode erzielten Abschreibungen höher liegen als bei der degressiven, wechselt die Funktion auf die lineare Abschreibung, falls Nicht\_wechseln mit FALSCH belegt wird oder nicht angegeben ist. Ist das Argument mit WAHR belegt, so wird das Abschreibungsverfahren nicht geändert.

## **XINTZINSFUSS()**

**Syntax:** XINTZINSFUSS(Werte; Zeitpkte; Schätzwert)

Die Funktion berechnet den Zinssatz einer Investition bei regelmäßigen Auszahlungen. Zsr sind die Zahlungszeiträume, über die die Auszahlungen erfolgen. Rmz ist der über die Zahlungszeiträume konstante regelmäßige Auszahlungsbetrag (muß negativ sein). Bw der Barwert, also der aktuelle Wert, z. B. die anfängliche Einzahlung.

Wird Zw (der Zukunftswert) nicht angegeben, so wird er von der Funktion als 0 angenommen, wird F (Zahlungen am Anfang oder am Ende der Zahlungszeiträume) nicht gesetzt, so wird es ebenfalls als 0 (Ende der Periode) angenommen.

Mit Schätzwert können Sie angeben, wie hoch Sie den Zins einschätzen (das kann das Rechenverfahren verkürzen). Geben Sie keinen Wert ein, so wird er von der Funktion mit 10 % veranschlagt.

## **ZINSSATZ()**

**Syntax:** ZINSSATZ(Abrechnung; Fälligkeit; Anlage; Rückzahlung; Basis)

Berechnet den (jährlichen) Zinssatz für eine Investition, bei der zwischen Abrechnung und Rückzahlung keine Zinsen ausgeschüttet werden.

Bei der Investition handelt es sich nach der Terminologie der Argumente um den Kauf von Wertpapieren, die Funktion kann aber auf jede Situation angewendet werden, in der ein Betrag eingezahlt und ein anderer Betrag nach einer Frist zurückgezahlt wird.

Abrechnung ist der Kauftermin, Fälligkeit der Rücknahmetermin. Anlage ist der Betrag, der angelegt wurde, Rückzahlung der Betrag, der zum Fälligkeitstermin zurückgezahlt wird. Basis gibt an, nach welchem Zeitsystem die Zinstage berechnet werden. Vgl. hierzu den Vorspann zu den Finanzfunktionen.

## **ZINSTERMNZ()**

**Syntax:** ZINSTERMNZ(Abrechnung; Fälligkeit; Häufigkeit; Basis)

Berechnet das Datum der ersten Zinsausschüttung nach dem Kauftermin (Abrechnung).

Abrechnung ist der Kauftermin, Fälligkeit der Rücknahmetermin. Häufigkeit ist die Zahl der jährlichen Zinsausschüttungen (1, 2 oder 4). Basis gibt an, nach welchem Zeitsystem die Zinstage berechnet werden. Vgl. hierzu den Vorspann zu den Finanzfunktionen.

## **ZINSTERMTAGE()**

**Syntax:** ZINSTERMTAGE(Abrechnung; Fälligkeit; Häufigkeit; Basis)

Berechnet die Anzahl der Tage in derjenigen Zinsperiode, in die der Abrechnungszeitpunkt fällt.

Abrechnung ist der Kauftermin, Fälligkeit der Rücknahmetermin. Häufigkeit ist die Zahl der jährlichen Zinsausschüttungen (1, 2 oder 4). Basis gibt an, nach welchem Zeitsystem die Zinstage berechnet werden. Vgl. hierzu den Vorspann zu den Finanzfunktionen.

Die Funktion hat natürlich nur Sinn, wenn die Basis so gewählt ist, dass das Zinsjahr nicht mit 360 Tagen gerechnet wird (sonst kommt bei einer Häufigkeit von 4 immer 90 heraus) und wenn die Häufigkeit nicht mit 1 angesetzt ist (sonst kommt immer 360 bzw. 365 heraus).

## **ZINSTERMTAGNZ()**

**Syntax:** ZINSTERMTAGNZ(Abrechnung; Fälligkeit; Häufigkeit; Basis)

Berechnet die Tage vom Kauftermin bis zum ersten Zinstermin.

Abrechnung ist der Kauftermin, Fälligkeit der Rücknahmetermin. Häufigkeit ist die Zahl der jährlichen Zinsausschüttungen (1, 2 oder 4). Basis gibt an, nach welchem Zeitsystem die Zinstage berechnet werden. Vgl. hierzu den Vorspann zu den Finanzfunktionen.



## ZINSTERMTAGVA()

**Syntax:** ZINSTERMTAGVA(Abrechnung; Fälligkeit; Häufigkeit; Basis)

Berechnet die Tage vom letzten Zinstermin vor der Abrechnung bis zur Abrechnung. Abrechnung ist der Kauftermin, Fälligkeit der Rücknahmetermin. Häufigkeit ist die Zahl der jährlichen Zinsausschüttungen (1, 2 oder 4). Basis gibt an, nach welchem Zeitsystem die Zinstage berechnet werden. Vgl. hierzu den Vorspann zu den Finanzfunktionen.

## ZINSTERMVZ()

**Syntax:** ZINSTERMVZ(Abrechnung; Fälligkeit; Häufigkeit; Basis)

Berechnet das Datum des letzten Zinstermins vor der Abrechnung. Abrechnung ist der Kauftermin, Fälligkeit der Rücknahmetermin. Häufigkeit ist die Zahl der jährlichen Zinsausschüttungen (1, 2 oder 4). Basis gibt an, nach welchem Zeitsystem die Zinstage berechnet werden. Vgl. hierzu den Vorspann zu den Finanzfunktionen.

## ZINSTERMZAHL()

**Syntax:** ZINSTERMZAHL(Abrechnung; Fälligkeit; Häufigkeit; Basis)

Berechnet die Zahl der Zinstermine zwischen dem Kaufdatum und dem Fälligkeitsdatum. Abrechnung ist der Kauftermin, Fälligkeit der Rücknahmetermin. Häufigkeit ist die Zahl der jährlichen Zinsausschüttungen (1, 2 oder 4). Basis gibt an, nach welchem Zeitsystem die Zinstage berechnet werden. Vgl. hierzu den Vorspann zu den Finanzfunktionen.

## ZINSZ()

**Syntax:** ZINSZ(Zins; Zr; Zzr; Bw; Zw; F)

Die Funktion ermittelt den Zinsanteil für die Abzahlung eines Darlehens bei gleichbleibenden Zahlungen und bei gleichbleibendem Zinssatz (vgl. KAPZ und RMZ). Argumente sind Zins, Zzr (die Zahlungszeiträume) und Bw (der Barwert, die Investition). Optional kann außer F (Fälligkeit; 0 für Ende der Perioden, 1 für Anfang) noch Zw (Zukunftswert) eingegeben werden. Vgl. KAPZ().

## ZW()

**Syntax:** ZW(Zins; Zzr; Rmz; Bw; F)

Die Funktion liefert als Ergebnis den zukünftigen Wert auf der Grundlage gleicher Zahlungen zu gleichem Zinssatz über die Laufzeit.

Mit Zins wird der Zins und mit Zzr die Zahlungszeiträume (oder die Anzahl der Zahlungen) angegeben.

Außer dem Betrag, der periodisch zu zahlen ist (Rmz oder auch die Ratenzahlung), können noch zwei optionale Argumente angegeben werden. Bw ist der gegenwärtige Barwert (bei Einzahlungen auf der Bank etwa der gegenwärtige Kontostand). Wird Bw nicht angegeben, so wird er als 0 angenommen.

Mit F wird angegeben, ob die Zahlungen jeweils am Ende einer Periode erfolgen (F = 0) oder am Anfang (F = 1).

## ZW2()

**Syntax:** ZW2(Kapital; Zinsen)

Die Funktion berechnet die Zahl der Zahlungsperioden, z. B. zur Tilgung eines Darlehens, wenn die Zahlungen zu einem gleichbleibendem Zinssatz erfolgen.

Mit Zins und Rmz wird der Zinssatz und der Betrag der jeweiligen Zahlung angegeben. Bw ist der Betrag, der durch die Zahlungen erreicht werden soll (z. B. die Höhe des Darlehens). Die optionalen Argumente Zw und F stehen für den Zukunftswert (welcher Betrag soll am Ende vorhanden sein) und die Fälligkeit der Zahlungen (F = 0 für Ende der Perioden). Werden diese Argumente nicht angegeben, dann werden sie mit 0 angenommen.

## Referenz der Datums- und Zeitfunktionen

### ARBEITSTAG()

**Syntax:** ARBEITSTAG(Ausgangsdatum; Tage; Freie\_Tage)

Liefert von einem Ausgangsdatum aus gerechnet ein neues Datum, das sich um die gegebene Zahl an Arbeitstagen und freien Tagen vom Ausgangsdatum unterscheidet.

Das Ausgangsdatum kann im Datumsformat oder als serielle Zahl in der Bezugszelle stehen. Wird es direkt als Argument eingetragen, muß es als serielle Zahl oder als DATWERT(Datum) (siehe dort) eingegeben werden. Von diesem Datum (einschließlich) aus rechnet die Funktion die mit Tage angegebenen Arbeitstage sowie die als Freie\_Tage gegebenen Tage hinzu, ergänzt die in den Zeitraum fallenden Samstage und Sonntage und gibt das neue Datum aus.

Werden negative Tage angegeben, rechnet die Funktion rückwärts; negative Freie\_Tage führen zu einer Fehlermeldung.

## **BRTEILJAHRE()**

**Syntax:** BRTEILJAHRE(Ausgangsdatum; Enddatum; Basis)

Liefert für Zinsberechnungen die Zeitspanne zwischen Ausgangsdatum und Enddatum als Bruchteil von Jahren.

Mit Basis wird angegeben, wie das Jahr berechnet werden soll: 0 oder nicht angegeben (12 Monate mit 30 Tagen, USA); 1 (genau); 2 (Tage/360); 3 (Tage/365); 4 (Europa, 30/360). Vgl. den Vorspann zu den Finanzfunktionen.

## **DATUM()**

**Syntax:** DATUM(Jahr; Monat; Tag)

Die Funktion berechnet eine serielle Zahl für das eingegebene Datum, das im Zeitraum vom 1.1.1900 bis zum 31.12.2078 liegen darf. Die wiedergegebenen Werte liegen dann zwischen 1 und 65380.

Die Werte für Monat und Tag dürfen auch größer als 12 bzw. 31 liegen, die „überschüssigen“ Monate oder Tage werden intern verrechnet.

Wie das Ergebnis der Funktion in der Zelle angezeigt wird, hängt vom Zahlenformat der Zelle ab. Ohne weitere Formatierung wird das Datumsformat verwendet.

## **DATWERT()**

**Syntax:** DATWERT(Datumstext)

Ein Datum läßt sich als Zeichenfolge eingeben, wenn diese in Anführungszeichen gesetzt wird. Die Funktion DATWERT wandelt diese Zeichenfolge in eine serielle Zahl um. Die Zeichenfolge muß allerdings in einer Schreibweise eingegeben werden, die einem der gültigen Excel-Datumsformate entspricht.

Die zeitlichen Begrenzungen sind dieselben wie in der Funktion DATUM(). Diese Funktion eignet sich speziell für Importe, z. B. aus Textverarbeitungsprogrammen oder aus Programmen, die ein anderes Datumsformat verwenden.

## **EDATE()**

**Syntax:** EDATE(Ausgangsdatum; Monate)

Liefert ein um Monate verschobenes Datum.

## **HEUTE()**

**Syntax:** HEUTE()

Beim Aufruf der Funktion wird das Systemdatum aus der Computeruhr als Ergebnis geliefert.

Die Funktion aktualisiert den ausgegebenen Wert jeweils bei Neuberechnung. Das kann ein durchaus unerwünschter Effekt sein. Vermeiden läßt sich die Neuberechnung, wenn nach Eingabe der Funktion [F9] (Umwandeln in den Wert) gedrückt wird oder indem mit *Bearbeiten/Kopieren* und anschließend *Bearbeiten/Inhalte einfügen* mit der Option *Werte* die Funktion durch ihr Ergebnis ersetzt wird.

## **JAHR()**

**Syntax:** JAHR(Zahl)

Wird als Argument eine Zahl, die einer Datumsseriennummer entspricht, eingefügt, so wird als Ergebnis das Jahr ausgegeben, das dieser Zahl entspricht. Ebenso kann als Argument auch Text verwendet werden, wenn er einem Excel-Datumsformat entspricht und in Anführungszeichen gesetzt ist.

## **JETZT()**

**Syntax:** JETZT()

Die Funktion liefert Datum und Uhrzeit aus der Systemuhr Ihres Computers.

Die Funktion aktualisiert den ausgegebenen Wert jeweils bei Neuberechnung. Das kann ein durchaus unerwünschter Effekt sein. Vermeiden läßt sich die Neuberechnung, wenn nach Eingabe der Funktion [F9] (Umwandeln in den Wert) gedrückt wird oder indem mit *Bearbeiten/Kopieren* und anschließend *Bearbeiten/Inhalte einfügen* mit der Option *Werte* die Funktion durch ihr Ergebnis ersetzt wird.

Verwenden können sie JETZT für zeitliche Dokumentationen oder als Stoppuhr für Zeitabläufe. Obwohl die Funktion kein Argument verlangt, müssen die leeren Klammern mit angegeben werden.

## **KALENDERWOCHE()**

**Syntax:** KALENDERWOCHE(Datum;Rückgabe)

Liefert für das angegebene Datum die entsprechende Kalenderwochenzahl. Rückgabewert 1 gibt an, dass die Woche am Sonntag beginnt, Rückgabewert 2, dass die Woche am Montag beginnt.

## **MINUTE()**

**Syntax:** MINUTE(Zahl)

Der Minutenanteil einer seriellen Zahl wird ausgegeben. Der Wert kann auch als Text eingegeben werden, dann muß er aber in Anführungszeichen und in einem Excel-Datumsformat erscheinen.

## **MONAT()**

**Syntax:** MONAT(Zahl)

Die Funktion gibt einen Wert von 1 bis 12 aus, der aus der seriellen Zahl für das Datum den entsprechenden Monat ermittelt. Ebenso kann als Argument auch Text verwendet werden, wenn er einem Excel-Datumsformat entspricht und in Anführungszeichen gesetzt ist.

## **MONATSENDE()**

**Syntax:** MONATSENDE(Ausgangsdatum;Monate)

Liefert den letzten Tag des Monats, in den das um Monate verschobene Datum fällt.

## **NETTOARBEITSTAGE()**

**Syntax:** NETTOARBEITSTAGE(Ausgangsdatum;Enddatum;Freie\_Tage)

Liefert die Zahl der Arbeitstage zwischen den beiden angegebenen Daten.

Als Arbeitstage werden alle Werktage (nicht Samstag und Sonntag) gezählt; mit Freie\_Tage lassen sich zusätzlich Feiertage und Urlaubstage angeben, die abgezogen werden.

## **SEKUNDE()**

**Syntax:** SEKUNDE(Zahl)

Der Sekundenanteil einer Zeitseriennummer wird als eine Zahl von 0 bis 59 ausgegeben. Der Wert kann auch als Text eingegeben werden, dann muß er aber in Anführungszeichen und in einem Excel-Datumsformat erscheinen. Im Beispiel wird von der Funktion nur der Zeitanteil rechts vom Komma für die Sekundendarstellung verwendet.

## **STUNDE()**

**Syntax:** STUNDE(Zahl)

Die Funktion gibt einen Wert von 0 bis 23 aus, indem aus der seriellen Zahl für Datum und Zeit die entsprechende Stunde ermittelt wird. Ebenso kann als Argument auch Text verwendet werden, wenn er einem Excel-Datumsformat entspricht und in Anführungszeichen gesetzt ist.

## **TAG()**

**Syntax:** TAG(Zahl)

Die Funktion gibt einen Wert von 1 bis 31 aus, indem aus der seriellen Zahl für Datum und Zeit der entsprechende Tag ermittelt wird. Ebenso kann als Argument auch Text verwendet werden, wenn er einem Excel-Datumsformat entspricht und in Anführungszeichen gesetzt ist.

## TAGE360()

**Syntax:** TAGE360(Ausgangsdatum;Enddatum;Methode)

Die Funktion gibt die Anzahl der Tage an, die zwischen zwei verschiedenen Daten liegen. Grundlage dafür ist das Zinsjahr, die Einteilung des Jahres in 12 Monate zu 30 Tagen. Liegt das Enddatum kalendarisch vor dem Anfangsdatum, so wird das Ergebnis in negativer Form ausgegeben.

Das optionale Argument Methode schaltet einen Wahrheitswert. Mit WAHR oder nicht angegeben wird die amerikanische Methode für die Berechnung verwendet, mit FALSCH die europäische.

## WOCHENTAG()

**Syntax:** WOCHENTAG(Zahl;Typ)

Das Ergebnis dieser Funktion ist eine Zahl von 1 bis 7. Jede Zahl stellt dabei die Nummer eines Wochentags dar. Ist für das optionale Argument Typ 1 angegeben oder fehlt das Argument, beginnt die Zählung am Sonntag mit 1. Mit Typ=2 beginnt die Zählung am Montag, mit Typ=3 beginnt die Zählung am Dienstag.

## ZEIT()

**Syntax:** ZEIT(Stunde;Minute;Sekunde)

Die Funktion liefert die Zeit, die durch die Werte für Stunde, Minute und Sekunde festgelegt wird.

## ZEITWERT()

**Syntax:** ZEITWERT(Zeit)

Eine Zeitangabe läßt sich als Zeichenfolge eingeben, wenn diese in Anführungszeichen gesetzt sind. Die Funktion ZEITWERT wandelt diese Zeichenfolge in eine serielle Zahl um. Die Zeichenfolge muß allerdings in einer Schreibweise eingegeben werden, die einem der gültigen Excel-Datumsformate entspricht.

Diese Funktion eignet sich speziell für Importe, z. B. aus Textverarbeitungsprogrammen oder aus Programmen, die ein anderes Zeitformat verwenden.

# Mathematische Funktionen

Angesichts der Fülle von Funktionen aus den Bereichen von Mathematik und Trigonometrie sind einige Vorbemerkungen nötig.

## Trigonometrische Funktionen

Zu den zahlreichen trigonometrischen Funktionen, die Excel bereitstellt, vorweg einige Erläuterungen. Die Winkelfunktionen, die nicht nur in der Geometrie, sondern auch in all jenen wissenschaftlichen Bereichen benötigt werden, die sich mit Schwingungen im weitesten Sinne (Schall, Licht, Elektrizität, Mechanik) befassen, verdanken sich ursprünglich Berechnungen am rechtwinkligen Dreieck.

Gearbeitet wird mit Winkelgrößen, die nicht in Grad, sondern in Bogenmaß angegeben sind. Als Bogenmaß eines Winkels wird die Länge des Kreisbogens bezeichnet, den der Winkel aus dem Einheitskreis (Kreis mit dem Radius 1) ausschneidet. Da der Umfang eines Kreises  $2 \cdot r \cdot \pi$  ist, beträgt der Umfang des Einheitskreises  $2 \cdot \pi$ , das Bogenmaß des Winkels 360 Grad ist also  $2 \cdot \pi$ .

Die Umrechnung von Grad in Bogenmaß ist demnach:

$$\begin{aligned}\text{Grad} &= \text{Bogenmaß} \cdot 180/\pi \\ \text{Bogenmaß} &= \text{Grad} \cdot \pi/180\end{aligned}$$

Die Winkelfunktionen geben die Verhältnisse bestimmter Seiten eines rechtwinkligen Dreiecks in Abhängigkeit von einem Winkel an:

$$\begin{aligned}\text{SIN}(x) &= \text{Gegenkathete}/\text{Hypothense} \\ \text{COS}(x) &= \text{Ankathete}/\text{Hypothense} \\ \text{TAN}(x) &= \text{Gegenkathete}/\text{Ankathete} \\ \text{COT}(x) &= \text{Ankathete}/\text{Gegenkathete}\end{aligned}$$

Der Cotangens ist als Funktion in Excel nicht enthalten, da er der Kehrwert des Tangens ist:

$$\text{COT}(x) = 1/\text{TAN}(x)$$

Mit den jeweiligen Arcus-Funktionen wird ausgehend von einer Winkelfunktion der zugehörige Winkel im Bogenmaß (= Arcus) ermittelt. Es gilt also für alle Winkelfunktionen eine Beziehung nach dem Muster:

$$\text{Wenn } y = \text{SIN}(x) \text{ dann gilt } \text{ARCSIN}(y) = x$$

## Hyperbolische Funktionen

Die hyperbolischen Funktionen sind, trotz ihrer an Winkelfunktionen erinnernden Namen, keine eigentlichen Winkelfunktionen. Das wird schon daran deutlich, dass sie im Gegensatz zu den Winkelfunktionen nicht periodisch verlaufen. Die Namensgebung rechtfertigt sich aber aus einer großen formalen Übereinstimmung bei den Beziehungen zwischen den einzelnen Funktionen sowie aus mathematischen Zusammenhängen zwischen Winkelfunktionen und hyperbolischen Funktionen. Anwendung finden derartige Funktionen z. B. in einigen statistischen Näherungsverfahren, bei statischen Berechnungen und in der Analysis.

Die Bezeichnung der inversen Funktionen (Area-Funktionen) mit ARC ... (analog zu den Arcus-Funktionen der Winkelfunktionen) ist allerdings trotz der formalen Ähnlichkeit zu den Arcusfunktionen sehr unglücklich: die Area-Funktionen liefern keinen Winkel im Bogenmaß (Arcus), sondern in der geometrischen Deutung eine Fläche. Der Cotangens hyperbolicus ist in Excel nicht eigens als Funktion enthalten, da er der Kehrwert des Tangens hyperbolicus ist:

$$\text{coth}(x) = 1/\text{tanh}(x)$$

Für die Area-Funktionen gilt (analog zu den Winkelfunktionen) für alle hyperbolischen Funktionen eine Beziehung nach dem Muster:

$$\text{Wenn } y = \text{SINHYP}(x) \text{ dann gilt } \text{ARCSINHYP}(y) = x$$

## Referenz der mathematische Funktionen

### ABRUNDEN()

**Syntax:** ABRUNDEN(Zahl;Anzahl\_Stellen)

Die Funktion rundet eine Zahl in Richtung 0 auf dem Zahlenstrahl. Positive Zahlen werden also abgerundet, negative Zahlen aufgerundet.

Mit dem Argument Anzahl\_Stellen wird die Anzahl der Stellen angegeben, die beim Runden berücksichtigt wird. Ist Anzahl\_Stellen gleich 1 oder größer, dann wird auf die entsprechende Anzahl Dezimalstellen abgeschnitten, ist es Null, dann werden die Dezimalstellen abgeschnitten, so dass die Ganzzahl übrigbleibt. Ist das Argument negativ, wird um die angegebenen Stellen nach links gerundet und zwar so, dass sich die gerundete Zahl in Richtung Null bewegt.

### ABS()

**Syntax:** ABS(Zahl)

Diese Funktion eliminiert die Vorzeichen von Zahlenwerten. Ob ABS auf Zahlen mit negativem oder positivem Vorzeichen angewendet wird, als Ergebnis wird der nicht negativen Betrag dieser Zahlen geliefert (in mathematischer Schreibweise  $|x|$ ).

### ARCCOS()

**Syntax:** ARCCOS(Zahl)

Die Funktion ARCCOS ergibt zu dem gegebenen Cosinus eines Winkels den zugehörigen Winkel im Bogenmaß. Für Cosinuswerte zwischen -1 und 1 liegt der Winkel zwischen 0 und  $\pi$ .

Soll der Winkel in Grad angegeben werden, müßte die Formel im Beispiel lauten:

$$\text{ARCCOS}(0,9)*180/\text{PI}()$$

Das Ergebnis ist dann 25,84 Grad.

## **ARCCOSHYP()**

**Syntax:** ARCCOSHYP(Zahl)

Umkehrfunktion zu COSHYP().

## **ARCSIN()**

**Syntax:** ARCSIN(Zahl)

ARCSIN berechnet zu einem gegebenen Sinuswert den zugehörigen Winkel. Für Sinuswerte zwischen -1 und 1 liegt der Winkel zwischen  $-\pi/2$  und  $\pi/2$ , d. h. zwischen -90 und 90 Grd.

## **ARCSINHYP()**

**Syntax:** ARCSINHYP(Zahl)

Umkehrfunktion zu SINHYP().

## **ARCTAN()**

**Syntax:** ARCTAN(Zahl)

Mit ARCTAN wird zu dem gegebenen Tangens eines Winkels der Winkel errechnet. Für Tangenswerte zwischen 0 und Unendlich liegt der Winkel zwischen 0 und  $\pi/2$ .

## **ARCTAN2()**

**Syntax:** ARCTAN2(x\_Koordinate;y\_Koordinate)

Diese spezielle Form des Arcustangens gestattet es, bei einer Geraden, die durch den 0-Punkt eines Koordinatenkreuzes geht, direkt durch Angabe von x-Koordinate und y-Koordinate eines Punktes den Steigungswinkel der Geraden zu ermitteln. Das Ergebnis liegt zwischen  $\pi$  und  $-\pi$ .

## **ARCTANHYP()**

**Syntax:** ARCTANHYP(Zahl)

Umkehrfunktion zu TANHYP().

## **AUFRUNDEN()**

**Syntax:** AUFRUNDEN(Zahl;Anzahl\_Stellen)

Die Funktion rundet eine Zahl von der Null weg. Mit dem Argument Anzahl\_Stellen wird die Anzahl der Stellen angegeben, die beim Runden berücksichtigt wird. Ist Anzahl\_Stellen gleich 1 oder größer, dann wird auf die entsprechende Anzahl Dezimalstellen aufgerundet, ist es Null, dann wird auf die nächste Ganzzahl aufgerundet, ist es negativ, dann wird die entsprechende Ziffer links vom Dezimalzeichen aufgerundet.

## **COS()**

**Syntax:** COS(Zahl)

Die Funktion berechnet den Cosinus eines Winkels (im Beispiel 30 Grd). Der Winkel wird im Bogenmaß angegeben. Das Ergebnis liegt zwischen -1 und 1.

## **COSHYP()**

**Syntax:** COSHYP(Zahl)

$$\text{COSHYP}(x) = (e^x + e^{-x})/2$$

x liegt zwischen -Unendlich und Unendlich, COSHYP(x) zwischen 1 und Unendlich.

## **EXP()**

**Syntax:** EXP(Zahl)

Die Eulersche Zahl e (2,71828...) wird mit Zahl potenziert. Da e die Basis der natürlichen Logarithmen ist, ergibt EXP(1) den Wert von e (2,71828...). EXP(LN(x)) liefert den Wert x. EXP ist die Umkehrfunktion von LN. Die Funktion EXP wird in erster Linie zur Beschreibung von Wachstumsfunktionen verwendet.

## FAKULTÄT()

**Syntax:** FAKULTÄT(Zahl)

Die Fakultät einer Zahl (in der Mathematik geschrieben  $3!$  für Fakultät von 3) ist das Produkt  $1*2*3*4*...*$ Zahl. Anwendungsbereich ist die Kombinatorik, in der im weitesten Sinne die Frage gestellt wird, wie viele Möglichkeiten es gibt, Elemente aus einer Menge zu kombinieren. Hier nur ein 13983816, d. h., es gibt über 13 Mio. Möglichkeiten, 6 Zahlen auf einem Lottoschein anzukreuzen, und soviel Kombinationen müßten durchgespielt werden, um ganz sicher „6 Richtige“ im Lotto zu haben.

## GANZZAHL()

**Syntax:** GANZZAHL(Zahl)

Ein numerischer Ausdruck wird auf die nächstkleinere ganze Zahl reduziert, die Nachkommastellen werden gleichsam abgeschnitten. Negative Zahlen werden von 0 weg, also zur nächsten kleineren Zahl gerundet. GANZZAHL(-5,7) ergibt also -6.

## GERADE()

**Syntax:** GERADE(Zahl)

Die Funktion rundet Zahlen zur nächsten geraden Ganzzahl weg von 0 und ist damit komplementär zur Funktion UNGERADE. Wenn Zahl schon eine gerade ganze Zahl ist, dann wird dieser Wert wiedergegeben.

## GGT()

**Syntax:** GGT(Zahl1; Zahl2; ...)

Liefert den größten gemeinsamen Teiler von zwei oder mehreren ganzen Zahlen.

Ein gemeinsamer Teiler ist eine Zahl, durch die alle angegebenen Zahlen geteilt werden können, ohne dass ein Rest bleibt. Der größte gemeinsame Teiler ist die größte derartige Zahl.

## GRAD()

**Syntax:** GRAD(Winkel)

Winkelangaben werden aus dem Bogenmaß (Radiant oder als Einheit rad) in Grad übertragen. Für die Konversion gilt die Formel:

$$\text{Grad} = \text{rad} * 360 / (2 * \text{PI}).$$

## KGV()

**Syntax:** KGV(Zahl1; Zahl2; ...)

Liefert das kleinste gemeinsame Vielfache der angegebenen Zahlen.

Das kleinste gemeinsame Vielfache ist die kleinste Zahl, die durch alle angegebenen Zahlen ohne Rest teilbar ist.

## KOMBINATIONEN()

**Syntax:** KOMBINATIONEN(n; k)

Die Funktion gibt den Wert des Binomialkoeffizienten aus. KOMBINATIONEN() rechnet nach der Formel

$$n! / (k! * (n-k)!)$$

Die Funktion beantwortet die Frage, wie viele Gruppen der Größe k aus n Elementen gebildet werden können, wenn die Reihenfolge keine Rolle spielt.

Werden keine ganzen Zahlen eingegeben, so werden sie zu GANZZAHL abgeschnitten, 5,456 ergibt dann 5. Wenn die Argumente kleiner Null eingegeben werden und n kleiner als k sein sollte, wird eine Fehlermeldung ausgegeben.

Wollen Sie etwa berechnen, wie viele Möglichkeiten es gibt, 6 Zahlen von 49 anzukreuzen, dann liefert KOMBINATIONEN(49; 6) das Ergebnis 13983816 (vgl. FAKULTÄT).

## KÜRZEN()

**Syntax:** KÜRZEN(Zahl; Anzahl\_Stellen)

Ein numerischer Ausdruck wird auf die mit Anzahl\_Stellen angegebenen Stellen verkürzt, d. h., die Stellen hinter dem Komma werden abgeschnitten. Wird Anzahl\_Stellen nicht

angegeben, so wird das Argument als 0 angenommen und der ganzzahlige Rest wiedergegeben. Dies gilt auch (im Gegensatz zu GANZZAHL) bei negativen Zahlen: KÜRZEN(-5,7) ergibt -5.

## LN()

**Syntax:** LN(Zahl)

LN() ist der natürliche Logarithmus einer Zahl, d. h. derjenige Wert, mit dem die Eulersche Zahl potenziert werden müßte, damit die Zahl als Ergebnis herauskommt. Die Anwendung liegt ähnlich wie bei der Funktion EXP() im Bereich von Wachstumsprozessen in Naturwissenschaft und Technik.

## LOG()

**Syntax:** LOG(Zahl;Basis)

LOG ist von allen Logarithmusfunktionen, die Excel bereitstellt, die allgemeinste. Sie gestattet es, den Logarithmus einer Zahl zu jeder beliebigen Basis zu ermitteln, wobei das Ergebnis immer derjenige Wert ist, mit dem die Basis potenziert werden müßte, um die Zahl zu erhalten.

LOG(Zahl;2) ergibt den binären Logarithmus  
LOG(Zahl;EXP(1)) ergibt den natürlichen Logarithmus;  
LOG(Zahl;10) ergibt den dekadischen Logarithmus.

Wird auf die Angabe einer Basis verzichtet, dann nimmt Excel als Basis 10 an (das entspricht der folgenden Funktion LOG10).

Zur Erinnerung hier einige der Rechenregeln, die für alle Logarithmen gelten:

LOG(a;a)=1 (der Logarithmus der Basis ist 1)  
LOG(1)=0 (Unabhängig von der Basis ist der Logarithmus von 1 gleich 0)  
LOG(a\*b)=LOG(a)+LOG(b)  
LOG(a/b)=LOG(a)-LOG(b)

d. h., Multiplikation und Division werden auf logarithmischer Ebene auf Addition und Subtraktion zurückgeführt.

LOG(a^n)=n\*LOG(a) (Potenzen werden auf logarithmischer Ebene auf Multiplikationen zurückgeführt).

## LOG10()

**Syntax:** LOG10(Zahl)

Der dekadische Logarithmus einer Zahl, d. h., der Wert, mit dem 10 potenziert werden müßte, um die Zahl zu erhalten. LOG10(100)=2, d. h., 10 muß zum Quadrat genommen werden, um 100 zu erhalten.

## MDET()

**Syntax:** MDET(Matrix)

Die Ermittlung der Determinante einer Matrix wird benötigt bei der Lösung von linearen Gleichungen mit mehreren Unbekannten. Gleichungen dieser Art kommen unter anderem im naturwissenschaftlich-technischen Bereich (z. B. Berechnung von Widerstandsnetzwerken) und in der Ökonomie (z. B. Optimierungsaufgaben, die von mehreren Faktoren abhängen) vor.

Die Lösung derartiger Gleichungssysteme ist „von Hand“ eine äußerst zeitraubende Angelegenheit, über die Bildung von Determinanten dagegen einfacher erledigt. In Excel lassen sich derartige Aufgaben allerdings wesentlich schneller mit dem Solver bearbeiten.

## MINV()

**Syntax:** MINV(Matrix)

Bildet die Inverse zu einer Matrix. Die Funktion findet Anwendung bei der Lösung von Gleichungssystemen mit mehreren Unbekannten (vgl. MDET).

## MMULT()

**Syntax:** MMULT(Matrix1;Matrix2)

Liefert das Produkt zweier Matrizen. Die Spalten von Matrix1 müssen in der Anzahl mit den Zeilen von Matrix2 übereinstimmen.



## OBERGRENZE()

**Syntax:** OBERGRENZE(Zahl,Schritt)

Die Funktion rundet den Wert Zahl auf das nächste Vielfache von Schritt auf und ist damit komplementär zu UNTERGRENZE.

Mit dem Wert 0,05 für Schritt kann z. B. bestimmt werden, dass die Hundertstelstelle beim Aufrunden immer nur eine 5 oder eine Null sein kann. Mit einem Wert 0,5 für Schritt wird dafür gesorgt, dass z. B. nicht mehr in Pfennigen, sondern nur noch für 5-Pfennig-Stücke ausgepreist wird.

Aufrunden meint im Sinne dieser Funktion, dass immer von Null weg gerundet wird, z. B. wird OBERGRENZE(-4,2546;-0,5) zu -4,5. Bei unterschiedlichen Vorzeichen für Zahl und Schritt wird eine Fehlermeldung ausgegeben.

## PI()

**Syntax:** PI()

Die Funktion PI liefert den numerischen Wert von PI. Neben der Umrechnung von Winkeln von Grad in Bogenmaß wird PI unter anderem für Kreis- und Kugelberechnungen benötigt. Obwohl für PI als Verhältniszahl kein Argument angegeben wird, benötigt Excel die beiden Klammern hinter dem Funktionsnamen.

Kreisumfang:  $U=2*r*PI()$

Kreisfläche:  $A=r^2*PI()$

Kugeloberfläche:  $O=4*r^2*PI()$

Kugelvolumen:  $V=4/3*r^3*PI()$

## POLYNOMIAL()

**Syntax:** POLYNOMIAL(Zahl1;Zahl2;...)

Liefert die Fakultät der Summe der Argumente geteilt durch das Produkt der Fakultäten, also

$(Zahl1+Zahl2...)! / Zahl1!*Zahl2!...$

Die Funktion kann in der Kombinatorik nützlich sein.

## POTENZ()

**Syntax:** POTENZ(Zahl;Potenz)

Die Funktion gibt als Ergebnis die Potenzierung einer Zahl aus. Dabei wird dasselbe Rechen-verfahren verwendet, das auch mit dem Operatorzeichen „^“ zur Anwendung kommt. Statt  $4^2$  kann also auch POTENZ(4;2) eingegeben werden.

## POTENZREIHE()

**Syntax:** POTENZREIHE(x;n;m;Koeffizienten)

Liefert eine Summe von Potenzen der Zahl x. Berechnet wird nach der Formel:

$a1*x^n + a2*x^{(n+m)} + a3*x^{(n+2m)} ...$

wobei a1, a2 ... mit Koeffizienten angegeben wird. Die Zahl der hier eingetragenen Werte liefert zugleich die Zahl der Summanden in der Formel.

Mit dieser Funktion lassen sich gut Näherungsberechnungen für Funktionen durchführen.

## PRODUKT()

**Syntax:** PRODUKT(Zahl1;Zahl2;...)

Die Argumente (Zahl1, Zahl2 bis Zahl30) werden miteinander multipliziert.

## QUADRATESUMME()

**Syntax:** QUADRATESUMME(Zahl1;Zahl2;...)

Die Funktion berechnet die Summe der Quadrate der einzelnen Werte. Außer Zahlen können auch numerische Formeln oder die Adressen von Bereichen stehen, die auf Zahlen oder numerische Formeln verweisen.

## QUOTIENT()

**Syntax:** QUOTIENT(Zähler;Nenner)

Liefert das ganzzahlige Ergebnis einer Division; der Rest wird weggelassen.

## **RADIANT()**

**Syntax:** RADIANT(Winkel)

Das Ergebnis ist der Radiant, Einheitszeichen rad oder auch das Bogenmaß eines Winkels. Im Bogenmaß wird das Verhältnis von Kreisbogen und Radius angegeben. Für die Berechnung gilt:

$$\text{rad} = 2 \cdot \pi \cdot \text{Winkel} / 360.$$

## **REST()**

**Syntax:** REST(Zahl; Divisor)

Der Restbetrag (Modulus) bei der Division Zahl/Divisor. Der Divisor muß ein anderer Wert als Null sein, das Vorzeichen des Ergebnisses ist immer das des Divisors. Ist Zahl 0, so ist das Ergebnis auch 0.

## **RÖMISCH()**

**Syntax:** RÖMISCH(Zahl; Typ)

Wandelt Zahlen in Zeichenketten um, die der römischen Zahlendarstellung entsprechen. Mit Typ kann festgelegt werden, nach welchem Schema umgewandelt wird: 0 oder nicht festgelegt ergibt den klassischen Typus, 1, 2 und 3 ergibt eine verkürzte Form, 4 die vereinfachte. Ist für Typ WAHR gesetzt, erhalten Sie den klassischen Typ, FALSCH ergibt den vereinfachten.

## **RUNDEN()**

**Syntax:** RUNDEN(Zahl; Anzahl\_Stellen)

Eine Zahl wird auf die durch Stellen angegebene Stellenzahl auf- oder abgerundet. Ist die zu rundende Dezimalstelle 5, dann wird von 0 weg gerundet. (RUNDEN(3,45; 1) ergibt 3,5). Ist Zahl = 0, wird auf eine Ganzzahl gerundet, im Unterschied zur Funktion GANZZAHL, die die Nachkommastellen abschneidet.

## **SIN()**

**Syntax:** SIN(Zahl)

Die Funktion berechnet den Sinus eines Winkels (im Beispiel 30 Grad). Der Winkel wird im Bogenmaß angegeben. Das Ergebnis liegt zwischen -1 und 1.

## **SINHYP()**

**Syntax:** SINHYP(Zahl)

$$\text{SINHYP}(x) = (e^x - e^{-x})/2$$

x liegt zwischen -Unendlich und Unendlich, SINHYP(x) zwischen -Unendlich und Unendlich.

## **SUMME()**

**Syntax:** SUMME(Zahl1; Zahl2; ...)

Die Summe der Argumente wird berechnet. Insgesamt sind 30 Argumente erlaubt. Wie sich unterschiedliche Datentypen auswirken, zeigt die folgende Tabelle. Auch hier ergeben sich wieder Unterschiede (analog zu PRÖDUKT()) zu den Ergebnissen von Formeln, die mit „+“ aufgebaut sind.

## **SUMMENPRODUKT()**

**Syntax:** SUMMENPRODUKT(Matrix1; Matrix2; Matrix3; ...)

Die einzelnen Elemente der als Argumente angegebenen Matrizen werden der Reihe nach miteinander multipliziert, und anschließend werden die Multiplikationsergebnisse summiert. Sie müssen von der Zeilenanzahl und der Spaltenanzahl her gleich sein. Es handelt sich also nicht um ein Summenprodukt, sondern um eine Produktsumme.

Diese Funktion ist immer dann nützlich, wenn die einzelnen Produkte nicht angezeigt werden müssen oder sollen, sondern nur die Endsumme. Als Argumente werden natürlich normalerweise (wie in den meisten Funktionen) Bereichsnamen oder Bereichsadressen angegeben, wie das folgende Beispiel demonstriert:

## SUMMEWENN()

**Syntax:** SUMMEWENN(Bereich; Suchkriterien; Summe\_Bereich)

Diese Funktion addiert die Summen der quadrierten x-Werte und der quadrierten y-Werte.

Die Spalten für die Argumente müssen dieselbe Größe haben, sonst wird eine Fehlermeldung ausgegeben. Diese und die anderen beiden (zusammengehörenden) Funktionen werden besonders häufig für statistische Anwendungen benötigt.

## SUMMEXMY2()

**Syntax:** SUMMEXMY2(Matrix\_x; Matrix\_y)

Die Differenz zwischen x-Werten und y-Werten wird ausquadriert und die einzelnen Werte werden dann aufsummiert.

Die Spalten für die Argumente müssen dieselbe Größe haben, sonst wird eine Fehlermeldung ausgegeben. Diese und die anderen beiden vorhergehenden Funktionen werden besonders häufig für statistische Anwendungen benötigt.

## TAN()

**Syntax:** TAN(Zahl)

Die Funktion berechnet den Tangens eines Winkels (im Beispiel 30 Grad). Der Winkel wird im Bogenmaß angegeben. Das Ergebnis liegt zwischen 0 und Unendlich.

## TANHYP()

**Syntax:** TANHYP(Zahl)

$$\text{TANHYP}(x) = (e^x - e^{-x}) / (e^x + e^{-x})$$

x liegt zwischen -Unendlich und Unendlich, TANHYP(x) zwischen -1 und 1.

Der hyperbolische Cotangens errechnet sich analog zu den Winkelfunktionen als  $1/\text{TANHYP}()$ .

## TEILERGEBNIS()

**Syntax:** TEILERGEBNIS(Funktion; Bezug1; Bezug2; ...)

Die Funktion gibt ein Teilergebnis aus einer Liste oder Datenbank zurück. Gleiche Ergebnisse erzielen Sie auch mit dem Menübefehl *Daten/Teilergebnisse*. Funktion gibt mittels einer Nummer von 1 bis 11 eine Funktion an, mit der die Teilergebnisse berechnet werden sollen:

1	MITTELWERT
2	ANZAHL
3	ANZAHL2
4	MAX
5	MIN
6	PRODUKT
7	STABW
8	STABWN
9	SUMME
10	VARIANZ
11	VARIANZEN

Bezug1 gibt den Bereich an, für den das Teilergebnis berechnet werden soll. Ausgeblendete Zeilen werden dabei ignoriert, Teilergebnisse von Teilergebnissen ebenfalls, weil sie sonst mehrfach in die Berechnung eingehen würden.

## UNGERADE()

**Syntax:** UNGERADE(Zahl)

Die Funktion rundet Zahlen zur nächsten ungeraden Ganzzahl weg von 0 und ist damit komplementär zur Funktion GERADE. Wenn Zahl schon eine ungerade ganze Zahl ist, dann wird dieser Wert wiedergegeben. Das eingegebene Argument wird vom Komma weg gerundet. Für positive Zahlen ein Beispiel.

UNGERADE(5,3) ergibt 7.

## UNTERGRENZE()

**Syntax:** UNTERGRENZE(Zahl; Schritt)

Die Funktion rundet den Wert Zahl ab auf das nächste Vielfache von Schritt und ist damit komplementär zu OBERGRENZE. Dadurch ist es möglich, Kalkulationsergebnisse so abzurunden, dass nicht nur der Wert der letzten Stelle, die angegeben wurde, gerundet wird. Mit dem Wert 0,05 für Schritt kann z. B. bestimmt werden, dass die Hundertstel-Stelle beim Abrunden immer nur eine 5 oder eine 0 sein kann. Mit einem Wert 0,5 für Schritt wird z. B. dafür gesorgt, dass nicht mehr in Pfennigen, sondern nur noch für 5-Pfennig-Stücke ausgepreist wird.

Aufrunden meint im Sinne dieser Funktion, dass immer zur Null hin gerundet wird, z. B. wird UNTERGRENZE(-2,54542;0,05) gerundet zu -2,5. Bei unterschiedlichen Vorzeichen für Zahl und Schritt wird eine Fehlermeldung ausgegeben.

## VORZEICHEN()

**Syntax:** VORZEICHEN(Zahl)

Die Funktion VORZEICHEN() ergibt bei positiven Zahlen 1, bei negativen Zahlen -1 und bei Null 0. Sie läßt sich immer dann anwenden, wenn Operationen vom Vorzeichen abhängig sein sollen.

Wenn etwa in der Zelle B22 die Gewinne errechnet werden, dann liefert (eingetragen z. B. in C22) die Formel WENN(VORZEICHEN(B22) = -1; *Achtung, Verluste!*; OK) ein OK, solange der Gewinn 0 oder größer ist. Bei negativen Gewinnen erscheint dagegen der Hinweis: *Achtung, Verluste!*

## VRUNDEN()

**Syntax:** VRUNDEN(Zahl; Vielfaches)

Rundet eine Zahl auf das nächste erreichbare ganzzahlige Vielfache einer mit Vielfaches gegebenen Zahl. Liegt die Zahl genau zwischen zwei Vielfachen, wird aufgerundet.

## WURZEL()

**Syntax:** WURZEL(Zahl)

Die Quadratwurzel einer Zahl wird ermittelt. Die Funktion ergibt immer den positiven Wert, der negative Wert (auch  $-6^*-6$  ist 36) muß im Bedarfsfall ergänzt werden. Für „höhere“ Wurzeln muß die Tatsache genutzt werden, dass statt der n-ten Wurzel auch geschrieben werden kann: hoch 1/n. Die dritte Wurzel (Kubikwurzel) aus einer Zahl wird demnach geschrieben:  $Zahl^{(1/3)}$ .

Zahl darf bei der Funktion WURZEL() keine negative Zahl sein. Soll verhindert werden, dass Zahl einen negativen Wert annimmt, kann mit der ABS-Funktion gearbeitet werden.

## WURZELPI()

**Syntax:** WURZELPI(Zahl)

Liefert die Quadratwurzel aus (Zahl \* PI).

## ZUFALLSBEREICH()

**Syntax:** ZUFALLSBEREICH(Min\_Wert; Max\_Wert)

Mit dieser Funktion erzeugen Sie eine ganzzahlige Zufallszahl zwischen Min\_Wert und Max\_Wert (beide eingeschlossen).

## ZUFALLSZAHL()

**Syntax:** ZUFALLSZAHL()

Diese Funktion produziert Zufallszahlen zwischen 0 und 1. Wenn Sie mit Zufallszahlen arbeiten wollen, die sich im nachhinein nicht mehr verändern, also nicht bei jeder Neuberechnung der Tabelle ebenfalls neu berechnet werden, dann drücken Sie die Taste [F9], wenn Sie die Funktion in der Bearbeitungszeile eingegeben haben (bevor Sie die [Enter]-Taste drücken).

Zufallszahlen anderer Größenordnungen können Sie durch entsprechende Rechenoperationen mit der generierten Zufallszahl oder mit der Funktion ZUFALLSBEREICH() erreichen:

ZUFALLSZAHL\*100 eine Zufallszahl zwischen 0 und 100

ZUFALLSZAHL\*100+50 eine Zufallszahl zwischen 50 und 150

## ZWEIFAKULTÄT()

**Syntax:** ZWEIFAKULTÄT(Zahl)

Liefert die „Altheffnersche Fakultät“ einer Zahl. Sie wird berechnet als

1 \* 3 \* 5 \* ... \* Zahl (bei ungerader Zahl)  
2 \* 4 \* 6 \* ... \* Zahl (bei gerader Zahl)

## Statistische Funktionen

Für Berechnungen aus dem Bereich der Statistik braucht Excel 2003 von der Funktionsvielfalt her gesehen den Vergleich mit professionellen Statistikprogrammen kaum zu scheuen. Um die Orientierung in diesem Bereich etwas zu erweitern, hier wenigstens ein paar kurze Bemerkungen.

### Überblick über die Statistikfunktionen

Die Statistikfunktionen decken mehrere Bereiche ab: die Analyse einzelner Stichproben, bei denen ein oder mehrere Größen erfaßt wurden, Analyse und Vergleich mehrerer Stichproben, Vergleich von Stichproben mit einer Grundgesamtheit und Aspekte wie Wahrscheinlichkeitsrechnung und Wahrscheinlichkeitsverteilungen von Zufallsvariablen.

### Stichproben und Grundgesamtheiten

Fast alle statistischen Verfahren haben entweder mit Stichproben oder mit Grundgesamtheiten zu tun, häufig auch direkt oder indirekt mit beidem. Eine Stichprobe ist eine Untergruppe von Elementen aus einer Grundgesamtheit, die zufällig aus dieser Grundgesamtheit ausgewählt wurden. Zufällig heißt hier, dass bei der Auswahl darauf geachtet wird, dass nicht bestimmte Elemente der Grundgesamtheit bevorzugt werden. Die Grundgesamtheit ist die Menge der Elemente, aus denen die Stichprobe entnommen wird.

Soll etwa für die weit verbreiteten Meinungsumfragen eine Stichprobe der Wahlberechtigten genommen werden, genügt es nicht, ein oder alle Telefonbücher zufällig aufzuschlagen (Telefonbesitzer werden bevorzugt) oder auf der Straße Leute anzusprechen (zu Hause Bleibende werden benachteiligt). Entsprechend haben die Institute, die Meinungsumfragen durchführen, wohlgehütete Geheimnisse, wie sie zu ihren repräsentativen Stichproben kommen.

### Zufallsvariable und Wahrscheinlichkeit

Auf der Grundlage statistischer Erhebungen oder zuweilen auch theoretischer Überlegungen läßt sich häufig für das Auftauchen bestimmter Größen eine bestimmte Wahrscheinlichkeit angeben. So ist etwa die Wahrscheinlichkeit, beim Münzwurf eine Zahl zu werfen  $1/2$  (0,5 oder 50 %), beim Würfeln eine 6 zu schaffen  $1/6$  (0,166666), aus einem Skatspiel eine bestimmte Karte zu ziehen  $1/32$  (0,03125) usw.

Für die Errechnung derartiger Wahrscheinlichkeiten, die nicht ganz so trivial sind, hält Excel 2003 eine Anzahl von Funktionen bereit.

In anderen Fällen wird die Wahrscheinlichkeit durch Abzählen der Grundgesamtheit ermittelt. Ist z. B. bekannt, dass in einem Land 51 % der Bevölkerung weiblich ist, dann ist die Wahrscheinlichkeit, dass ein zufällig ausgewählter Mensch weiblich ist, 0,51.

Eine Größe dieser Art heißt eine Zufallsvariable. Ist diese Variable so wie in den genannten Beispielen diskret, dann läßt sich direkt eine Wahrscheinlichkeit dafür angeben, dass sie einen bestimmten Wert oder einen von mehreren Werten annimmt. Anders ist es bei nichtdiskreten Variablen. Sind etwa die Körpergrößen einer Grundgesamtheit von Menschen erfaßt, dann stellt sich sowohl bei der Erfassung als auch bei der Angabe der Wahrscheinlichkeiten die Frage nach der Meßgenauigkeit und nach der Einordnung. Die Frage, wie wahrscheinlich ist die Größe von z. B. 1,73m, kann so gar

nicht beantwortet werden: Ist 1,7299999 mitgemeint oder nicht? Die Werte müssen also in Klassen eingeteilt werden (z. B. 172,5 bis 173,49999 ...).

Für alle derartigen Größen arbeitet die Statistik mit sogenannten Wahrscheinlichkeitsverteilungen für kontinuierliche Variablen, von denen Excel 2003 mehrere zur Verfügung stellt.

## Untersuchung von Stichproben

Bei der Untersuchung von Stichproben stellen sich meist zwei Fragen: Was hat die Stichprobe ergeben und welche Schlüsse erlaubt sie auf die Grundgesamtheit. Für die erste Frage gibt es zunächst zwei Größen: den Mittelwert und die Streuung. Als Maße sind hier eigentlich nur zwei gebräuchlich: das arithmetische Mittel und die Standardabweichung (oder das Quadrat der Standardabweichung, die Varianz). Aus diesen beiden Größen lassen sich dann auch die entsprechenden Parameter der Grundgesamtheit schätzen, wobei die Schätzung um so verlässlicher wird, je größer die Stichprobe ist, vgl. hierzu MITTELWERT(), VARIANZ() und STABW().

Eine andere Fragestellung bei einer Stichprobe ist, ob die ermittelten Werte einer bestimmten Gesetzmäßigkeit gehorchen. Soll etwa untersucht werden, ob es einen Zusammenhang zwischen dem persönlichen Einkommen und der Größe des genutzten Wohnraums gibt, dann ist anzunehmen, dass eine Beziehung besteht: je mehr Einkommen um so mehr Quadratmeter. Maße hierfür sind der Korrelationskoeffizient (KORREL()) und die Kovarianz (KOVAR()), die Angaben darüber liefern, ob und wie stark die Daten zusammenhängen.

Kann darüber hinaus vermutet werden, dass der Zusammenhang linear oder exponential ist, dann läßt sich dieser Zusammenhang weitgehend durch Regression, d. h. durch Rückführung der Werte auf eine Gerade oder eine Exponentialkurve klären. Hierfür stehen die mächtigen Funktionen RGP() und RKP() zur Verfügung.

## Statistische Tests

Bei Aufgabe statistischer Tests ist, ganz allgemein gesprochen, festzustellen, mit welcher Sicherheit oder Unsicherheit von Werten einer Stichprobe auf Werte der Grundgesamtheit geschlossen werden kann. Hierfür gibt es in Excel 2003 zwei Funktionsgruppen. Die eine bietet die Möglichkeit, direkt Tests anhand von Stichproben durchzuführen, die andere liefert Werte aus Wahrscheinlichkeitsverteilungen, anhand derer aus den Stichproben gewonnene Parameter überprüft werden können.

Beim t-Test (TTEST()) wird die Frage geprüft, ob zwei Stichproben sich in ihrem Mittelwert zufällig unterscheiden (dann wären beide Stichproben derselben Grundgesamtheit zufällig entnommen), oder ob sie sich nicht zufällig unterscheiden (dann stammen sie entweder aus verschiedenen Grundgesamtheiten oder sind nicht zufällig entnommen). Hier liefert die Funktion TTEST() direkt einen Wahrscheinlichkeitswert.

Ebenfalls mit dem t-Test läßt sich klären, ob die relative Häufigkeit eines Merkmals in einer Stichprobe zufällig von der Wahrscheinlichkeit dieses Merkmals in der Grundgesamtheit abweicht oder nicht. Leider ist dieser Fall nicht von einer Funktion erfaßt, so dass hier auf die t-Verteilung (TVERT()) zurückgegriffen werden muss.

Mit dem F-Test (FTEST()) wird geprüft, ob zwei Stichproben sich in ihrer Varianz zufällig unterscheiden oder nicht. Auch hier ist wieder der Umweg über die F-Verteilung (FVERT()) gangbar und bei manchen Fragestellungen notwendig.

Der Chi-Test (CHITEST()) schließlich dient der Überprüfung der Frage, ob eine Stichprobe, mit der mehrere Werte erfaßt sind, mit einer Grundgesamtheit übereinstimmt, aus der für diese Werte Erwartungswahrscheinlichkeiten bekannt sind. Auch hier steht zusätzlich die Verteilungsfunktion zur Verfügung.

## Verteilungsfunktionen

Von Zufallsgrößen war oben schon die Rede. Es wurde die Frage gestellt, mit welcher Wahrscheinlichkeit ein bestimmtes Ereignis auftritt. Für die Beantwortung dieser Frage

stehen in Excel 2003 eine Anzahl von Funktionen zur Verfügung, die das sonst notwendige Nachschlagen in umfangreichen Tabellenwerken ersparen können.

Gemäß der Unterscheidung in diskrete und stetige Zufallsgrößen lassen sich auch die zugehörigen Verteilungen in diskrete und stetige unterscheiden. Hier kurz ein Überblick mit einigen Hinweisen zur Anwendung:

Binomial-Verteilung (BINOMVERT()). Grundlage ist ein Ereignis, das jeweils mit einer bestimmten Wahrscheinlichkeit eintreten kann oder auch nicht. Beispiele hierfür sind Münzwürfe, Würfeln, Kartenziehen (wobei die Karte anschließend zurückgesteckt werden muß); aber auch männlich/weiblich, berufstätig/nicht berufstätig usw.

Hypergeometrische Verteilung (HYPERGEOMVERT()). Wird bei einem Beispiel wie Kartenziehen die Karte nicht zurückgesteckt, dann ändert sich beim nächsten Versuch die theoretische Wahrscheinlichkeit. In solchen Fällen wird die hypergeometrische Verteilung benutzt.

Poisson-Verteilung (POISSONVERT()). Diese Verteilung wird normalerweise als Näherung für die Binomial-Verteilung bei sehr großen Zahlen und sehr kleinen Wahrscheinlichkeiten genommen. Da Excel 2003 aber genauso gut mit der Binomial-Verteilung rechnen kann, ist dieser Ausweg nicht unbedingt erforderlich.

Normalverteilung (NORMVERT() und STANDNORMVERT()). In all den Fällen, wo einer Zufallsvariablen eine Grundgesamtheit zugrunde liegt, die sehr groß ist (ab 1000), und wo eine stetige Größe gemessen wird, können Sie davon ausgehen, dass sie normalverteilt ist. Das ist in zahlreichen Beispielen der Fall, so dass die Normalverteilung im Prinzip die wichtigste der stetigen Verteilungen ist.

Zusätzlich stellt Excel 2003 noch einige weniger gebräuchliche Verteilungen zur Verfügung, die gleichwohl für Spezialanwendungen nützlich sind.

Um die Handhabung der Verteilungsfunktionen zu erleichtern, hier noch abschließend ein Hinweis. Den Wahrscheinlichkeitsverteilungen liegt mathematisch immer eine Dichtefunktion zugrunde, bei der Normalverteilung etwa die berühmte Glockenkurve. Der jeweilige y-Wert sagt aber noch nichts über die Wahrscheinlichkeit des zugehörigen x-Wertes. Erst die Fläche zwischen zwei x-Werten (mathematisch das bestimmte Integral) ist ein Maß für die Wahrscheinlichkeit.

In Anlehnung an die diskreten Verteilungen verwendet Excel 2003 hier einen Wahrheitswert Kumuliert, der die Funktion veranlaßt, entweder die Dichte (FALSCH) oder die Wahrscheinlichkeit (WAHR) zu berechnen; das erste liefert also den Wert der Dichtefunktion, das zweite den Wert der Wahrscheinlichkeitsverteilung. Praktisch wird die Dichtefunktion so gut wie nie benötigt.

Für die meisten Verteilungsfunktionen steht obendrein eine inverse Funktion zur Verfügung (...INV()). Das Verhältnis der beiden Funktionen zueinander ist folgendes:

- Die Verteilungsfunktion liefert die Wahrscheinlichkeit dafür, dass eine Zufallsvariable einen Wert gleich oder kleiner als einen vorgegebenen Wert (als Argument x) annimmt.
- Die inverse Funktion liefert zu einer angegebenen Wahrscheinlichkeit den Wert, der gleich oder kleiner dem der Zufallsvariable mit der angegebenen Wahrscheinlichkeit ist. Da dieser Wert Quantil genannt wird, läßt sich der Zusammenhang auch so angeben:

...VERT(q) = p

...INV(p) = q

mit p = Wahrscheinlichkeit und q = Quantil.

# Referenz der statistischen Funktionen

## **ANZAHL()**

**Syntax:** ANZAHL(Wert1;Wert2;...)

Die Funktion ANZAHL ergibt die Anzahl der numerischen Werte, die in der Argumentenliste bzw. in dem Bereich, auf den sie sich bezieht, enthalten sind. Von den Werten bzw. Zellen insgesamt werden also Texteinträge und leere Zellen abgezogen.

## **ANZAHL2()**

**Syntax:** ANZAHL2(Wert1;Wert2;...)

Ermittelt werden die Werte insgesamt, die in der Argumentenliste bzw. in einem Bereich enthalten sind. Texteinträge werden mitgezählt, nur wirklich leere Zellen (oder Argumente) werden abgezogen. Die Funktionen ANZAHL und ANZAHL2 sind nicht nur statistisch zu verwenden, sondern auch als Zähler für Makros einzusetzen.

## **ANZAHLLEEREZELLEN()**

**Syntax:** ANZAHLLEEREZELLEN(Bereich)

Ermittelt die Anzahl der leeren Zellen im angegebenen Bereich. Zellen, die Leerzeichen enthalten, gelten nicht als leer.

## **BESTIMMTHEITSMASS()**

**Syntax:** BESTIMMTHEITSMASS(Y-Werte;X\_Werte)

Ermittelt das Quadrat des Pearsonschen Korrelationskoeffizienten, vgl. PEARSON. Das Bestimmtheitsmaß ist auch als Determinationskoeffizient bekannt und ist ein Maß für die Güte der Anpassung, die eine Regression erzielt.

## **BETAINV()**

**Syntax:** BETAINV(Wahrsch;Alpha;Beta;A;B)

Die Funktion liefert das Quantil der Betaverteilung und ist die Umkehrung zu BETAVERT().

Als notwendige Argumente sind Wahrsch (Wahrscheinlichkeit), Alpha (Parameter) und Beta (Parameter) einzutragen. A und B sind optionale Argumente, die die Intervallgrenzen bezeichnen. Werden sie nicht angegeben, dann wird  $A = 0$  und  $B = 1$  gesetzt. Vgl. BETAVERT().

Für den Zusammenhang zwischen BETAINV() und BETAVERT() gilt:

Wenn  $\text{Wert} = \text{BETAVERTEIL}(x; \dots)$ , dann ist  $x = \text{BETAINV}(\text{Wert}; \dots)$ .

## **BETAVERTEIL()**

**Syntax:** BETAVERTEIL(x;Alpha;Beta;A;B)

Die Funktion liefert die Wahrscheinlichkeitsverteilung für eine Beta-verteilte Zufallsvariable. Sie steht in engem Zusammenhang mit der Gammaverteilung und kann bei Berechnung der Verteilung von Größen aus beliebigen gleichmäßig stetig verteilten Grundgesamtheiten verwendet werden.

Es wird berechnet, mit welcher Wahrscheinlichkeit die Zufallsvariable einen Wert zwischen A und x annimmt.

Das Argument x ist die Größe der Zufallsvariablen im Intervall A bis B, Alpha und Beta – beide müssen größer als 0 sein – sind Parameter der Verteilung (in der Literatur normalerweise als p und q geführt – zumindest die Bezeichnung Beta ist irreführend).

A und B sind optionale Argumente und bezeichnen die untere und obere Grenze des Intervalls. Werden für A und B keine Werte angegeben, dann gilt die standardmäßige Betaverteilung ( $A = 0$  und  $B = 1$ ).

Ist x kleiner A oder größer als B, liefert Excel einen Fehlerwert, ist A gleich B ebenso.

## **BINOMVERT()**

**Syntax:** BINOMVERT(Zahl\_Erfolge;Versuche;Erfolgswahrsch;Kumuliert)

Die Funktion gibt die Wahrscheinlichkeit dafür an, dass bei alternativen diskreten Versuchsergebnissen bei einer mit Versuche angegebenen Anzahl von Versuchen ein bestimmtes Ergebnis mit einer durch Zahl\_Erfolge angegebenen Häufigkeit auftritt. Die



(vorweg ermittelte) Wahrscheinlichkeit für das Einzelergebnis wird mit Erfolgswahrsch (zwischen 0 und 1) angegeben.

Beispiele sind Münzwürfe (Erfolgswahrscheinlichkeit 1/2), Würfel (1/6) etc.

Kumuliert verlangt einen Wahrheitwert und beschreibt den Typ der Funktion. Wird das Argument mit FALSCH belegt, wird der Wert der Wahrscheinlichkeitsfunktion geliefert. Das oben angeführte Beispiel liefert die Wahrscheinlichkeit dafür, dass bei zehn Würfeln mit einem Würfel genau dreimal die Sechs gewürfelt wird. Wird das Argument mit WAHR belegt, wird die Verteilungsfunktion berechnet, im Beispiel die Wahrscheinlichkeit, dass die Sechs bis zu dreimal gewürfelt wird.

## CHIINV()

**Syntax:** CHIINV(Wahrsch;Freiheitsgrade)

Die Funktion liefert die (z. B. in statistischen Tabellenwerken tabellierten) Quantile der Chi-Quadrat-Verteilung.

Als Argumente verlangt diese Funktion die Irrtumswahrscheinlichkeit Wahrsch und die Anzahl der Freiheitsgrade, vgl. CHITEST() und CHIVERT().

## CHITEST()

**Syntax:** CHITEST(Beob\_Meßwerte;Erwart\_Werte)

Die Funktion liefert direkt den Wahrscheinlichkeitswert für den Chi-Quadrat-Test beim Vergleich zwischen beobachteten und erwarteten Größen.

Als Argumente werden je eine Matrix für die beobachteten Werte Beob\_Meßwerte und die theoretisch erwarteten Werte Erwart\_Werte eingetragen.

Den gleichen Wert würde auch die Funktion CHIVERT() liefern, wenn für x der Chi-Quadratwert und für die Freiheitsgrade 5 (6 Möglichkeiten - 1) eingetragen wird.

## CHIVERT()

**Syntax:** CHIVERT(x;Freiheitsgrade)

Die Funktion berechnet aus dem Wert für  $\chi^2$  und den Freiheitsgraden die Irrtumswahrscheinlichkeit für die Übereinstimmung von beobachteten und erwarteten Werten, vgl. CHITEST().

$\chi^2$  wird ermittelt als die Summe aus

$$\frac{(\text{Beobachtungswert} - \text{Erwartungswert})^2}{\text{Erwartungswert}}$$

für alle Werte.

Die Chi-Quadratverteilung ist eine Wahrscheinlichkeitsverteilung, die sich über die Summe von n unabhängigen, quadrierten, standardnormalverteilten Variablen und einer Anzahl von Freiheitsgraden definiert. Die Funktion wird für den Chi-Quadrat-Test benötigt, der beim Vergleich von empirischen zu theoretisch erwarteten Häufigkeiten zum Einsatz kommt.

Je nach Anzahl der Freiheitsgrade ändert sich der Charakter der Verteilung, mit steigender Anzahl wird die Funktion flacher und verschiebt sich nach rechts. Die Freiheitsgrade lassen sich am einfachsten ermitteln mit:

Anzahl der Möglichkeiten (bei kontinuierlichen Größen die Klassen) - 1 bei einer Datenspalte oder Zeile;

$$(\text{Anzahl der Zeilen} - 1) * (\text{Anzahl der Spalten} - 1) \text{ bei zweidimensionalen Wertetabellen.}$$

## EXPONVERT()

**Syntax:** EXPONVERT(x;Lambda;Kumuliert)

Die Funktion liefert die Werte für eine exponentialverteilte Zufallsvariable. Die Haltbarkeit von Bauteilen, die Halbwertszeiten radioaktiver Elemente etc. können mit einer Exponentialverteilung dargestellt werden.

Mit x wird das Quantil angegeben, für das der Wert ermittelt werden soll. Lambda ist ein Parameter der bei der Dichtefunktion den Anfangswert bei  $x=0$  sowie den Grad des Abfalls bestimmt.

Kumuliert ist ein Wahrheitwert, mit dem der Typ der Funktion bestimmt wird. Ist Kumuliert mit WAHR belegt, wird der Wert der Verteilungsfunktion geliefert (die Fläche bis zum Quantil), mit FALSCH belegt, ergibt sich der Wert für die Dichtefunktion (der Wert auf der y-Achse). Normalerweise wird die Verteilungsfunktion benötigt, deren Wert aussagt, wie groß die Wahrscheinlichkeit ist, dass die Zufallsvariable einen Wert zwischen 0 und x annimmt.

## **FINV()**

**Syntax:** FINV(Wahrsch; Freiheitsgrade1; Freiheitsgrade2)

Die Funktion liefert das Quantil der F-Verteilung (d. h. die Werte, die in statistischen Tabellenwerken tabelliert sind). Sie ist die Umkehrung von FVERT() (siehe dort). Die Funktion geht von einer zweiseitigen Verteilung aus.

Mit Wahrsch wird die Wahrscheinlichkeit angegeben. Die Freiheitsgrade sind die Größen der beiden miteinander verglichenen Stichproben minus 1.

## **FISHER()**

**Syntax:** FISHER(x)

Durch die Fisher-Transformation läßt sich die Korrelation r in eine annähernd normalverteilte Größe überführen und so anhand der Normalverteilung untersuchen.

## **FISHERINV()**

**Syntax:** FISHERINV(y)

Die Umkehrfunktion zu FISHER() (siehe dort).

## **FTEST()**

**Syntax:** FTEST(Matrix1;Matrix2)

Die Funktion liefert unmittelbar die Wahrscheinlichkeit der Übereinstimmung zweier Stichproben hinsichtlich ihrer Varianzen. Mit dem F-Test läßt sich also ermitteln, ob sich zwei Stichproben in ihren Varianzen nur zufällig unterscheiden. Matrix1 und Matrix2 sind Einzelwerte der beiden Stichproben. Die Argumente müssen nicht denselben Umfang haben.

Im Beispiel wird getestet, ob die beiden Stichproben aus derselben Grundgesamtheit stammen können. Die Wahrscheinlichkeit 0,73 ist dafür nicht hinreichend groß (0,95 wären hinreichend). Den gleichen Wert erhalten Sie, wenn Sie den F-Wert berechnen (größere Varianz/kleinere Varianz) und dann für diesen Wert FVERT() berechnen und das Ergebnis mit 2 multiplizieren.

Hinweis: Die Funktion FTEST() geht wie allgemein gebräuchlich von einem einseitigen Test aus (Abweichungen nur in einer Richtung), die Funktionen FVERT() und FINV() von zweiseitigen Tests. Das kann zu Verwirrungen führen.

## **FVERT()**

**Syntax:** FVERT(x;Freiheitsgrade1;Freiheitsgrade2)

Die Funktion liefert die Irrtumswahrscheinlichkeit der F-Verteilung. Die wichtigste Anwendung der F-Verteilung liegt in Signifikanztests für zwei unabhängige Stichproben. Je nach der Anzahl der Freiheitsgrade1 (Größe der ersten Stichprobe -1) und der Anzahl der Freiheitsgrade2 (Größe der zweiten Stichprobe -1) unterscheiden sich die F-Verteilungen und nehmen verschiedene Gestalt an.

Mit x wird das Quantil der Verteilung eingegeben. Für den Zusammenhang zwischen FVERT() und FINV() gilt:

Wenn Wert = FINV(a;...), dann ist a = FVERT(Wert;...).

## **GAMMAINV()**

**Syntax:** GAMMAINV(Wahrsch; Alpha; Beta)

Die Funktion liefert das Quantil der Gammaverteilung. Sie ist die Umkehrfunktion zu GAMMAVERT() (siehe dort).

Alpha und Beta sind Funktionsparameter (in der Literatur werden als Parameter meist b und p angegeben). Beta = 1 liefert die standardisierte Gammaverteilung.

Zwischen GAMMAINV() und GAMMAVERT() besteht folgender Zusammenhang:

Wenn Wert = GAMMAINV(x;...), dann ist x = GAMMAVERT(Wert;...WAHR).

## **GAMMALN()**

**Syntax:** GAMMALN(x)

Die Funktion liefert den natürlichen Logarithmus zur Gammafunktion.

Es gilt folgende Beziehung für die Funktion: EXP(GAMMALN(x)) ergibt FAKULTÄT(x-1).

## **GAMMAVERT()**

**Syntax:** GAMMAVERT(x; Alpha; Beta; Kumuliert)

Die Funktion liefert die Werte für eine gammaverteilte Zufallsvariable (bei der Verteilungsfunktion die Wahrscheinlichkeit dafür, dass eine Zufallsgröße einen Wert zwischen 0 und x annimmt).

Diese Funktion findet vor allem in der Bedienungs- und Zuverlässigkeitstheorie Anwendung.

Von den Argumenten bezeichnet x das Quantil, für das die Wahrscheinlichkeit berechnet werden soll, Alpha und Beta sind Parameter der Verteilung (vgl. GAMMAINV()). Kumuliert bestimmt den Typ der Verteilung: mit WAHR wird der Wert der Verteilungsfunktion berechnet, mit FALSCH der Wert der Dichtefunktion.

Wird Beta = 1 gesetzt, ergibt dies die Werte für die standardisierte Gammaverteilung.

Durch geeignete Wahl der Parameter läßt sich die Gammaverteilung in andere Verteilungen überführen (Chi-Quadrat-Verteilung, Weibullverteilung etc.). Wird Alpha = 1 gesetzt, ergibt sich eine Exponentialverteilung mit Lambda = 1/Beta.

## **GEOMITTEL()**

**Syntax:** GEOMITTEL(Zahl1; Zahl2; ...)

Die Funktion berechnet das geometrische Mittel für eine Reihe von Daten, die positive Zahlen sein müssen. Berechnet wird das geometrische Mittel, indem alle Beobachtungen miteinander multipliziert werden und dann die n-te Wurzel aus dem Ergebnis gezogen wird. Das Ergebnis von GEOMITTEL ist immer kleiner als das Ergebnis für MITTELWERT. Das wichtigste Anwendungsgebiet ist die Errechnung von Durchschnitten bei einer Abfolge von Veränderungen.

Ist einer der für die Berechnung aufgenommenen Werte 0 oder kleiner, so gibt Excel eine Fehlermeldung aus.

## **GESTUTZMITTEL()**

**Syntax:** GESTUTZMITTEL(Matrix; Prozent)

Die Funktion gibt das arithmetische Mittel eines Datensatzes zurück, bei dem aber die niedrigsten und die höchsten Werte nicht berücksichtigt werden.

Wieviel der mit Matrix angegebenen Daten jeweils oben und unten abgeschnitten werden, wird durch das Argument Prozent bestimmt. Wird für Prozent 0.1 (10 %) eingegeben, werden 5 % der niedrigsten und 5 % der höchsten Werte für die Berechnung nicht berücksichtigt.

Das Verfahren ist dazu gedacht, „Ausreißer“ aus einer Mittelwertberechnung herauszuhalten, im obigen Beispiel würde MITTELWERT zu einem Ergebnis von 6,71 führen.

## **GTEST()**

**Syntax:** GTEST(Matrix; x; Sigma)

Die Funktion liefert die zweiseitige Wahrscheinlichkeit für einen Gauss-Test (normalverteilte Daten). Mit diesem Test kann die Wahrscheinlichkeit dafür geschätzt werden, dass ein bestimmter Wert aus derselben (normalverteilten) Grundgesamtheit stammt wie eine gegebene Stichprobe.

Mit Matrix wird der Datenbereich der Stichprobe angegeben, mit der der Wert x verglichen werden soll. Das optionale Argument Sigma bezeichnet die bekannte Standardabweichung der Grundgesamtheit. Wird Sigma nicht angegeben, dann verwendet die Funktion die Standardabweichung der Stichprobe als Schätzwert für Sigma.

Der Test liefert als Wahrscheinlichkeit 0,46 für die Hypothese, dass der Einzelwert zur gleichen Grundgesamtheit gehört. Der Test wird brauchbarer, wenn die Stichprobe größer ist und die Standardabweichung der Grundgesamtheit vorliegt.

## **HARMITTEL()**

**Syntax:** HARMITTEL(Zahl1; Zahl2; ...)

Die Funktion gibt das harmonische Mittel zurück. Das harmonische Mittel wird zuweilen im Rahmen varianzanalytischer Problemstellungen benötigt. Das Ergebnis fällt kleiner aus als bei GEOMITTEL() und MITTELWERT().

Beträgt der Wert eines Elements der Daten 0 oder weniger, dann gibt das Programm eine Fehlermeldung aus.

## HÄUFIGKEIT()

**Syntax:** HÄUFIGKEIT(Daten;Klassen)

Die Funktion berechnet die Wahrscheinlichkeiten einer hypergeometrisch verteilten Zufallsvariablen. Die Funktion wird in Fällen angewendet, in denen es durch Entnahme aus der Grundgesamtheit zu einer Änderung ihrer Zusammensetzung führt, so dass hier die Binomial-Verteilung nicht eingesetzt werden kann. Vgl. BINOMVERT().

Mit Umfang\_S und Umfang\_G werden die Größe der entnommenen Stichprobe und die Größe der Grundgesamtheit angegeben. Erfolge\_G gibt an, wie oft das zu testende Ereignis in der Grundgesamtheit enthalten ist, Erfolge\_S wie oft es in der Stichprobe enthalten sein soll.

Im Beispiel wird die Wahrscheinlichkeit dafür ermittelt, dass aus einem Skatblatt mit 32 Karten sich einer der vier Buben im Skat befindet.

## KGRÖSSTE()

**Syntax:** KGRÖSSTE(Matrix;k)

Die Funktion gibt den k-größten Wert zurück. Das Argument k bestimmt, der wievieltgrößte Wert aus der Matrix gesucht wird.

Ist k = 5, so gibt die Funktion den fünftgrößten Wert als Ergebnis aus; mit k = 1 wird der größte Wert ausgegeben.

Für den Fall, dass das Argument k = 0 ist oder die Anzahl der Datensätze übersteigt, wird eine Fehlermeldung ausgegeben, vgl. auch KKLEINSTE().

## KKLEINSTE()

**Syntax:** KKLEINSTE(Matrix;k)

Die Funktion gibt den k-kleinsten Wert zurück. Das Argument k bestimmt, der wievieltkleinste Wert aus der Matrix gesucht wird.

Ist k = 5, so gibt die Funktion den fünftkleinsten Wert als Ergebnis aus; mit k = 1 wird der kleinste Wert ausgegeben.

Für den Fall, dass das Argument k = 0 ist oder die Anzahl der Datensätze übersteigt, wird eine Fehlermeldung ausgegeben, vgl. auch KGRÖSSTE().

## KONFIDENZ()

**Syntax:** KONFIDENZ(Alpha;Standabwn;Umfang\_S)

Die Funktion berechnet das Konfidenzintervall (auch Vertrauensbereich, Mutungsintervall) für den Mittelwert einer (normalverteilten) Grundgesamtheit anhand einer Stichprobe aus dieser Grundgesamtheit. Bei ein- wie zweiseitigen Fragestellungen wird ein bestimmter Prozentsatz (Alpha) extremer Fälle der Stichprobenverteilung als unwahrscheinlich ausgeschlossen. Diese Extremwerte liegen an den beiden Enden der Verteilung, der Bereich zwischen den beiden Extremwerten beidseitig vom Mittelwert ist das Konfidenzintervall.

Alpha ist die Irrtumswahrscheinlichkeit (gewählt wird zumeist 0,05, 0,01 oder 0,001), Standabwn ist die Standardabweichung, Umfang\_S die Größe der Stichprobe.

Für den Mittelwert der Grundgesamtheit gilt

$$M_{gg} = M_{st} \pm k \cdot (s / \text{WURZEL}(n))$$

wobei M<sub>gg</sub> und M<sub>st</sub> die Mittelwerte von Grundgesamtheit und Stichprobe sind, k der von der Funktion KONFIDENZ() ermittelte Wert, s die Standardabweichung der Stichprobe und n die Größe der Stichprobe.

Ergibt sich etwa im obigen Beispiel bei einer Werkstoffprüfung bei 200 Prüflingen eine durchschnittliche Länge von 102 mm mit einer Standardabweichung von 2,6, dann liegt das arithmetische Mittel mit einer Wahrscheinlichkeit von 90 % ( $0,9 = 1 - 2 \cdot 0,05$ ) im Bereich

$$102 - 0,3603 \cdot 2,6 / \text{WURZEL}(200) \text{ bis} \\ 102 + 0,3603 \cdot 2,6 / \text{WURZEL}(200)$$

also zwischen 101,934 und 102,066.

## KORREL()

**Syntax:** KORREL(Matrix1;Matrix2)

Liefert den Korrelationskoeffizienten (ein Maß für den linearen Zusammenhang) zweier Datenreihen aus verbundenen Stichproben (paarweise ermittelte Daten), die mit Matrix1 und Matrix2 angegeben werden.

Die Funktion ergibt den Wert 1 bei direktem linearen Zusammenhang (die beiden Regressionsgeraden der Daten sind direkt proportional), -1 bei indirektem Zusammenhang (die beiden Regressionsgeraden sind umgekehrt proportional), 0 wenn kein Zusammenhang besteht.

## **KOVAR()**

**Syntax:** KOVAR(Matrix1;Matrix2)

Liefert ähnlich wie die Funktion KORREL() ein Maß für den Zusammenhang zwischen den Daten zweier Datenreihen aus verbundenen Stichproben. Die Funktion ermittelt, in welchem Maß die Daten der beiden Datenreihen gemeinsam von ihrem jeweiligen Mittelwert abweichen.

Die Kovarianz kann benutzt werden zur Berechnung der Korrelation.

## **KRITBINOM()**

**Syntax:** KRITBINOM(Versuche;Erfolgswahrsch;Alpha)

Liefert die Anzahl der erfolgreichen Versuche, die mit einer Irrtumswahrscheinlichkeit von Alpha zu erwarten sind. Voraussetzung ist, dass die Zufallsgröße binomialverteilt ist (vgl. BINOMVERT()).

Mit Versuche wird die Zahl der Versuche angegeben; mit Erfolgswahrsch die Wahrscheinlichkeit für den erfolgreichen Ausgang eines Versuchs.

Im Beispiel sei gegeben, dass bei einer gegebenen Maschineneinstellung von 200 Prüflingen im Durchschnitt 180 (= 90 %) korrekt sind, die Wahrscheinlichkeit für einen korrekten Prüfling also 0,9 ist. Die Fragestellung: Mit wie vielen korrekten Prüflingen können Sie mit einer Irrtumswahrscheinlichkeit von 0,01 mindestens rechnen? Das Ergebnis lautet 170, d. h., in 99 % aller 200-Stück-Lieferungen werden mindestens 170 korrekte Produkte erhalten sein.

## **KURT()**

**Syntax:** KURT(Zahl1;Zahl2;...)

Die Funktion liefert die Kurtosis bzw. den Exzeß einer Häufigkeitsverteilung. Verglichen wird mit einer Normalverteilung mit gleichem Mittelwert und gleicher Streuung. Ein negatives Maß wie im Beispiel weist dabei auf einen stumpferen Verlauf, ein positives auf einen steileren Verlauf hin.

Zusammen mit der Funktion SCHIEFE() (vgl. dort) läßt sich das Verhältnis zu einer Normalverteilung bestimmen.

Bei Angabe von weniger als vier Werten wird als Fehlerwert #DIV/0! ausgegeben.

## **LOGINV()**

**Syntax:** LOGINV(Wahrsch;Mittelwert;Standabwn)

Die Funktion liefert das Quantil einer logarithmischen Normalverteilung. Die Funktion ist die Umkehrung von LOGNORMVERT().

Mit Wahrsch wird die Wahrscheinlichkeit, mit Mittelwert das Mittel und mit Standabwn die Standardabweichung der Stichprobe angegeben.

## **LOGNORMVERT()**

**Syntax:** LOGNORMVERT(x;Mittelwert;Standabwn)

Die Funktion liefert die Wahrscheinlichkeitsverteilung für eine logarithmische Normalverteilung. Bei einigen Experimenten, z. B. über Reaktionszeiten, ergibt sich als Häufigkeitsverteilung ein asymmetrischer, linkssteiler Kurvenzug. Durch Logarithmieren lassen sich daraus häufig normalverteilte Meßwerte herstellen.

Das Argument x bezeichnet den Wert des Quantils, Mittelwert das arithmetische Mittel und Standabwn die Standardabweichung der Stichprobe.

## **MAX()**

**Syntax:** MAX(Zahl1;Zahl2;...)

Mit MAX wird der größte Wert aus der Liste von Argumenten oder aus einem Bereich ermittelt.

## **MAXA()**

**Syntax:** MAXA(Wert1;Wert2;...)

Mit MAXA wird der größte Wert aus der Liste von Argumenten oder aus einem Bereich ermittelt. Auch Zellen mit Texten oder Wahrheitswerten werden berücksichtigt. Textzellen haben dabei den Wert 0.

## **MEDIAN()**

**Syntax:** MEDIAN(Zahl1;Zahl2;...)

Ermittelt den Median einer Reihe von Daten. Der Median ist derjenige Wert, der genau auf der Mitte einer Skala liegt, deren untere und obere Grenze durch den tiefsten und den höchsten Wert der Zahlenreihe gebildet wird.

Über dem Median liegen also genauso viele Werte wie unter ihm. Bei einer geraden Anzahl von Werten ermittelt Excel den Mittelwert der beiden mittleren Werte, um ihn als Median auszugeben.

## **MIN()**

**Syntax:** MIN(Zahl1;Zahl2;...)

Mit MIN wird der kleinste Wert aus der Liste von Argumenten oder aus einem Bereich ermittelt.

## **MINA()**

**Syntax:** MINA(Wert1;Wert2;...)

Mit MINA wird der kleinste Wert aus der Liste von Argumenten oder aus einem Bereich ermittelt. Auch Zellen mit Texten oder Wahrheitswerten werden berücksichtigt. Textzellen haben dabei den Wert 0.

## **MITTELABW()**

**Syntax:** MITTELABW(Zahl1;Zahl2;...)

Die Funktion liefert die mittlere Abweichung einer Reihe von Daten. Sie gehört damit zu den Dispersionsmaßen in der Statistik, die den Durchschnitt der in Absolutbeträgen gemessenen Abweichungen aller Meßwerte vom arithmetischen Mittel angibt. Die mittlere Abweichung wird in der Statistik eher selten verwendet.

## **MITTELWERT()**

**Syntax:** MITTELWERT(Zahl1;Zahl2;...)

Die Funktion MITTELWERT liefert das arithmetische Mittel der Argumente oder aller numerischen Werte des angegebenen Bereichs. Hierzu werden alle Werte aufsummiert und durch die Zahl der Werte geteilt. Zellen, die Text enthalten, werden ebenso wenig berücksichtigt wie leere Zellen. Die Funktion kann bis zu 30 numerische Argumente enthalten.

## **MITTELWERTA()**

**Syntax:** MITTELWERTA(Wert1;Wert2;...)

Die Funktion MITTELWERTA liefert das arithmetische Mittel der Argumente oder aller Werte des angegebenen Bereichs. Hierzu werden alle Werte aufsummiert und durch die Zahl der Werte geteilt. Zellen, die Text enthalten, werden ebenso berücksichtigt wie Wahrheitswerte. Die Funktion kann bis zu 30 Argumente enthalten. Textwerte zählen 0.

## **MODALWERT()**

**Syntax:** MODALWERT(Zahl1;Zahl2;...)

Liefert den in einer Datenreihe am häufigsten vorkommenden Wert. Damit gehört die Funktion zu den grundlegenden statistischen Kennwerten der Maße der zentralen Tendenz. Mit dem Modalwert lassen sich schnell Informationen über den Schwerpunkt der Verteilung gewinnen.

Betrachten Sie eine Verteilung, so ist das Maximum der Verteilung gleich dem Modalwert. Der Modalwert einer Häufigkeitsverteilung (s. HÄUFIGKEIT) liegt in der Kategorienmitte der am häufigsten besetzten Kategorie.

Kann die Funktion keinen Modalwert angeben, weil keiner der Werte zumindest zweimal vorkommt, wird ein Fehlerwert ausgegeben. Bei gleich häufigem Vorkommen verschiedener Werte wird der in der Liste zuerst vorkommende ausgegeben.

## NEGBINOMVERT()

**Syntax:** NEGBINOMVERT(Zahl\_MiBerfolge; Zahl\_Erfolge; Erfolgswahrsch)

Die Funktion benutzt als Grundlage ihrer Berechnungen ebenso wie BINOMVERT() die Binomial-Verteilung und wird auch als negative Binomial-Verteilung bezeichnet.

Sie berechnet, mit welcher Wahrscheinlichkeit ein zusammengesetztes Ereignis auftritt. Als Argumente werden Zahl\_Erfolge und Zahl\_MiBerfolge angegeben. Zusammen mit der Angabe von Erfolgswahrsch ermittelt die Funktion die Wahrscheinlichkeit dafür, dass das zusammengesetzte Ereignis (erst die angegebene Zahl an MiBerfolgen, dann die angegebene Zahl der Erfolge) auftritt.

Im Beispiel wird die Wahrscheinlichkeit ermittelt, hintereinander genau fünfmal nicht die Sechs und dann die Sechs zu werfen.

## NORMINV()

**Syntax:** NORMINV(Wahrsch; Mittelwert; Standabwn)

Die Funktion liefert das Quantil der Normalverteilung und ist die Umkehrung zu NORMVERT (siehe dort).

Als Argumente werden Wahrsch (die Wahrscheinlichkeit, zu der das Quantil gesucht wird) sowie der Mittelwert und die Standabwn (Standardabweichung) der Verteilung angegeben. Wie bei der Normalverteilung gilt auch hier, dass bei Mittelwert = 0 und Standardabweichung = 1 eine Standardnormalverteilung vorliegt. In diesem Fall kann auch STANDNORMINV aufgerufen werden.

## NORMVERT()

**Syntax:** NORMVERT(x; Mittelwert; Standabwn; Kumuliert)

Die Funktion liefert die Werte für eine Normalverteilung. Wird die Funktion grafisch dargestellt, ergibt sich immer ein glockenförmiger Verlauf. Wie er im einzelnen ausfällt, hängt von den Argumenten *Mittelwert* und *Standabwn* ab. Der Mittelwert (Erwartungswert) gibt die Lage der Funktion auf der x-Achse an und markiert dabei den Gipfel dieser Funktion. *Standabwn* (Standardabweichung) gibt die Streuung an und bestimmt damit, wie flach oder steil die Funktion verläuft.

Mit *Mittelwert* = 0 und *Standabwn* = 1 erhalten Sie die Standardnormalverteilung, die Sie auch mit STANDNORMWERT abfragen können. *x* bezeichnet den Wert, dessen Wahrscheinlichkeit berechnet werden soll (in der Grafik der Wert auf der x-Achse). Mit *Kumuliert* = WAHR erhalten Sie die Verteilungsfunktion (die Wahrscheinlichkeit dafür, dass die Zufallsvariable einen Wert von *x* oder kleiner annimmt). Mit FALSCH erhalten Sie die Werte der Dichtefunktion.

## PEARSON()

**Syntax:** PEARSON(Matrix1; Matrix2)

Liefert den Pearsonschen Korrelationskoeffizienten zweier Datenreihen aus verbundenen Stichproben. Für die Ergebnisse und die Argumente vgl. KORREL().

## POISSON()

**Syntax:** POISSON(x; Mittelwert; Kumuliert)

Die Funktion liefert die Werte für eine Poisson-Verteilung. Die Poisson-Verteilung ist wie die Binomial- und die hypergeometrische Verteilung eine Verteilung, die nur jeweils diskrete Werte annehmen kann. Die Poisson-Verteilung ist für große Zahlen eine gute Näherung für die Binomial-Verteilung.

An Argumenten verlangt die Funktion *x* (die Anzahl der Fälle) und *Mittelwert* (Erwartungswert). Mit *Kumuliert* = FALSCH wird die Wahrscheinlichkeit dafür berechnet, dass die Zufallsvariable den Wert *x* annimmt, mit *Kumuliert* = WAHR die Wahrscheinlichkeit dafür, dass die Zufallsvariable einen Wert von *x* oder kleiner annimmt.

Da die Poisson-Verteilung normalerweise dazu verwendet wurde, die bei großen Zahlen schwer zu handhabende Binomial-Verteilung anzunähern, gibt es kaum einen Grund, sie noch zu verwenden. Schließlich bietet Excel auch jene Funktion an.

## QUANTIL()

**Syntax:** QUANTIL(Matrix; Alpha)

Liefert denjenigen Wert einer Datenreihe, unterhalb dessen ein mit Alpha angegebener Bruchteil der Daten liegt.

Mit dieser Funktion wird eine Verteilung nach einer Skala unterteilt, deren unterster und oberster Punkt der tiefste und höchste Wert der Daten bildet.

Matrix sind die zu unterteilenden Daten. Durch das Argument Alpha wird ein Lage-Maß (Quantil) angegeben. Das Maß 0,25 (25 %) bezeichnet z. B. den Punkt, unterhalb dessen ein Viertel aller Beobachtungen liegen. Einige Quantile, die besonders oft verwendet werden, haben eigene Bezeichnungen wie Quartil für 25 %-Abschnitte, Dezil für 10 %-Abschnitte. Das zweite Quartil oder ein Quantil von 0,5 bezeichnet dann den Median.

Das Argument Alpha kann jeden Wert zwischen 0 und 1 annehmen; liegt ein Quantil zwischen zwei Beobachtungen, wird durch Interpolation der entsprechende Wert ermittelt. Enthält Matrix mehr als 8191 Datensätze, wird eine Fehlermeldung ausgegeben.

## QUANTILSRANG()

**Syntax:** QUANTILSRANG(Matrix;x;Genauigkeit)

Liefert die Angabe des Anteils von Daten, die unterhalb des angegebenen Wertes liegen.

Das Argument x bezeichnet den Wert, dessen relative Position ermittelt werden soll; Matrix sind die Daten. Wenn x selbst als Wert nicht in der Matrix auftaucht, wird der entsprechende Wert interpoliert. Mit Genauigkeit läßt sich die Anzahl der Stellen für die Ausgabe des Ergebnisses bestimmen. Wird Genauigkeit nicht angegeben, wird drei angenommen.

Der Zusammenhang mit QUANTIL sieht so aus:

Wenn  $x = \text{QUANTIL}(\text{Matrix};0,2)$ , dann ist  $0,2 = \text{QUANTILSRANG}(\text{Matrix};x)$ .

## QUARTILE()

**Syntax:** QUARTILE(Matrix;Quartil)

Die Funktion unterteilt die Daten von Matrix in vier Bereiche mit je gleichen Anteilen von Daten und ist damit ein Spezialfall von QUANTIL() (siehe dort).

Für Quartil sind vier Belegungen möglich: 0 (liefert den niedrigsten Wert); 1 (25 % Quantil), 2 (50 % Quantil, das ist der Median); 3 (75 % Quantil) und 4 (der höchste Wert).

Enthält Matrix mehr als 8191 Datensätze, wird eine Fehlermeldung ausgegeben.

## RANG()

**Syntax:** RANG(Zahl;Bezug;Reihenfolge)

Liefert die Position, die ein Wert in einer Datenreihe in bezug auf seine Größe einnimmt.

Mit Zahl wird der Wert angegeben, dessen Position bestimmt werden soll; Bezug ist die Datenreihe. Mit Reihenfolge wird angegeben, ob in fallender oder steigender Ordnung gezählt wird. Vorgegeben ist die fallende Ordnung, die dann verwendet wird, wenn das Argument nicht oder mit 0 belegt ist. Bei jedem anderen Wert zählt Excel in steigender Ordnung.

## RGPO()

**Syntax:** RGP(Y\_Werte;X\_Werte;Konstante;Stats)

Liefert für den vorgegebenen Wert x einen Schätzwert von y anhand einer linearen Regression (vgl. RKP).

## SCHIEFE()

**Syntax:** SCHIEFE(Zahl1;Zahl2;...)

Die Funktion liefert ein Maß für die Asymmetrie der Häufigkeitsverteilung einer Stichprobe. Verglichen wird mit einer Normalverteilung mit gleichem Mittelwert und gleicher Streuung.

Ist das Ergebnis größer als 0, dann ist die linke Seite steiler, die Verteilung heißt „rechts-schief“; ist das Ergebnis kleiner 0 ist die Verteilung „linksschief“. Vgl. auch KURT().

## STABW()

**Syntax:** STABW(Zahl1;Zahl2;...)

Die Funktion errechnet die Standardabweichung der Werte in der Argumentenliste. Die Argumentenliste wird dabei als Stichprobe genommen. Handelt es sich um eine Grundgesamtheit, ist die Funktion STABWN zu verwenden.



Die Standardabweichung ist die Wurzel aus der Varianz, die für Stichproben von der Funktion VARIANZ geliefert wird (siehe dort).

### **STABWA()**

**Syntax:** STABWA(Wert1;Wert2;...)

Die Funktion errechnet die Standardabweichung der Werte in der Argumentenliste. Die Argumentenliste wird dabei als Stichprobe genommen. Handelt es sich um eine Grundgesamtheit, ist die Funktion STABWNA zu verwenden. Textwerte und logische Werte werden mit ausgewertet. Textwerte zählen dabei 0.

Die Standardabweichung ist die Wurzel aus der Varianz, die für Stichproben von der Funktion VARIANZA geliefert wird (siehe dort).

### **STABWN()**

**Syntax:** STABWN(Zahl1;Zahl2;...)

Die Funktion errechnet die Standardabweichung der Werte in der Argumentenliste. Die Argumentenliste wird dabei als Grundgesamtheit genommen. Handelt es sich um eine Stichprobe, ist die Funktion STABW zu verwenden.

Die Standardabweichung ist die Wurzel aus der Varianz, die für Grundgesamtheiten von der Funktion VARIANZEN geliefert wird (siehe dort).

### **STABWNA()**

**Syntax:** STABWNA(Wert1;Wert2;...)

Die Funktion errechnet die Standardabweichung der Werte in der Argumentenliste. Die Argumentenliste wird dabei als Grundgesamtheit genommen. Handelt es sich um eine Stichprobe, ist die Funktion STABWA zu verwenden. Textwerte und logische Werte werden mit ausgewertet.

Die Standardabweichung ist die Wurzel aus der Varianz, die für Grundgesamtheiten von der Funktion VARIANZENA geliefert wird (siehe dort).

### **STANDARDISIERUNG()**

**Syntax:** STANDARDISIERUNG(x;Mittelwert;Standabwn)

Rechnet Werte, die einer Normalverteilung zugerechnet werden können, in Werte einer Standardnormalverteilung um.

Eine Standardnormalverteilung ist eine Normalverteilung mit einem arithmetischen Mittel von 0 und einer Standardabweichung von 1, vgl. STANDNORMVERT und NORMVERT.

### **STANDNORMINV()**

**Syntax:** STANDNORMINV(Wahrsch)

Die Funktion liefert den Wert auf der x-Achse für eine Standardnormalverteilung (Quantil). Die Funktion ist die Umkehrung zu STANDNORMVERT.

Die Standardnormalverteilung ist eine Variante der Normalverteilung und dadurch gekennzeichnet, dass der Mittelwert (Erwartungswert) gleich 0 ist und die Standardabweichung gleich 1.

### **STANDNORMVERT()**

**Syntax:** STANDNORMVERT(z)

Die Funktion gibt die Wahrscheinlichkeit dafür, dass eine Zufallsvariable aus einer Standardnormalverteilung den Wert z oder kleiner annimmt.

Die Standardnormalverteilung ist eine Variante der Normalverteilung und dadurch gekennzeichnet, dass der Mittelwert (Erwartungswert) gleich 0 ist und die Standardabweichung gleich 1. Die von dieser Funktion ermittelten Werte ließen sich auch über NORMVERT(z;0;1;WAHR) berechnen. Soll die Dichtefunktion berechnet werden, muß mit NORMVERT(z;0;1;FALSCH) gearbeitet werden.

In der Spalte S stehen die von der Funktion STANDNORMVERT berechneten Werte, in NK die Werte der kumulierten Normalverteilung (Verteilungsfunktion) und in N die der nichtkumulierten (Dichtefunktion).

### **STEIGUNG()**

**Syntax:** STEIGUNG(Y\_Werte;X\_Werte)

Die Funktion liefert die Steigung für die aus Y\_Werte und X\_Werte errechneten Regressionsgeraden, vgl. hierzu die Funktion RGP.

Die Regressionsgerade hat die Gleichung

$$y = b + m \cdot x$$

wobei b der Schnittpunkt der Geraden mit der y-Achse ist und m die Steigung. b kann mit der Funktion ACHSENABSCHNITT berechnet werden.

## STFEHLERYX()

**Syntax:** STFEHLERYX(Y\_Werte;X\_Werte)

Liefert den Standardschätzfehler für mittels linearer Regression aus den angegebenen Daten berechneten y-Werte, vgl. hierzu RGP.

## SUMQUADABW()

**Syntax:** SUMQUADABW(Zahl1;Zahl2;...)

Die Funktion gibt die Summe der quadratischen Abweichungen der Einzelwerte von ihrem arithmetischen Mittel an.

Das Ergebnis dieser Funktion wird häufig in der Statistik verwendet, z. B. ist es Bestandteil und Ausgangspunkt von Varianz und Standardabweichung.

## TINV()

**Syntax:** TINV(Wahrsch;Freiheitsgrade)

Die Funktion liefert das Quantil der t-Verteilung und ist damit die Umkehrung von TVERT mit dem Parameter 2 für Seiten. Die wiedergegebenen Werte sind in statistischen Tabellenwerken als t für zweiseitige Tests (Tests, bei denen die Werte nach beiden Seiten abweichen können) tabelliert.

Mit Wahrsch wird die Irrtumswahrscheinlichkeit angegeben, die Freiheitsgrade werden aus den zu vergleichenden Größen ermittelt.

Der prinzipielle Ablauf des t-Tests umfaßt folgende Schritte:

1. Aus den zu vergleichenden Größen wird ein rechnerischer t-Wert ermittelt (im folgenden tr).
2. Die Freiheitsgrade (im folgenden df) werden ermittelt.
3. Der errechnete tr-Wert wird mit dem von TINV gelieferten verglichen. Soll der Test einseitig sein, muß für die Funktion das Maß der Irrtumswahrscheinlichkeit halbiert werden.

Benötigt wird der von TINV() gelieferte Wert u. a. bei folgenden Tests:

Vergleich des Mittelwertes einer Stichprobe mit dem Mittelwert der Grundgesamtheit.

$$\begin{aligned} tr &= \text{WURZEL}(n) \cdot \text{ABS}(Ms - Mg) / Ss \\ df &= n - 1 \end{aligned}$$

mit n = Stichprobengröße; Ms = Mittelwert Stichprobe; Mg = Mittelwert Grundgesamtheit; Ss = Standardabweichung Stichprobe.

### *Vergleich der Mittelwerte zweier Stichproben*

$$\begin{aligned} tr &= (M1 - M2) / Sg \\ Sg^2 &= ((n1 - 1) \cdot S1^2 + (n2 - 1) \cdot S2^2) \cdot (n1 + n2) / ((n1 + n2 - 2) \cdot (n1 \cdot n2)) \\ df &= n1 + n2 - 2 \end{aligned}$$

mit M1 und M2 für die Mittelwerte der beiden Stichproben, S1 und S2 für die Standardabweichungen, n1 und n2 für die Stichprobengrößen.

Ist der so errechnete tr-Wert kleiner als der von TINV gelieferte, kann davon ausgegangen werden, dass die Unterschiede zwischen den zu testenden Größen zufällig sind. Die Wahrscheinlichkeit, dass diese Annahme falsch ist, wird mit der Irrtumswahrscheinlichkeit angegeben.

## TREND()

**Syntax:** TREND(Y\_Werte;X\_Werte;Neue\_x\_Werte;Konstante)

Die Funktion berechnet auf der Basis der linearen Regression (vgl. RGP) geschätzte y-Werte.

Y\_Werte sind die vorhandenen y-Werte; X\_Werte sind die vorhandenen x-Werte. Werden sie nicht angegeben, numeriert Excel die y-Werte durch. Mit Neue\_x\_Werte lassen sich von den vorhandenen x-Werte verschiedene x-Werte angeben, für die y-Werte geschätzt werden sollen (z. B. zum Hochrechnen von Werten).

TREND muß als Matrixfunktion eingegeben werden: Ausgabebereich markieren, Funktion eingeben, mit [Strg]+[Umschalt]+[Enter] beenden.

## TTEST()

**Syntax:** TTEST(Matrix1;Matrix2;Seiten;Typ)

Die Funktion gestattet den direkten Vergleich zweier Stichproben, ohne dass so viele rechnerische Zwischenschritte nötig wären wie bei dem unter TINV geschilderten Verfahren.

Die beiden Stichproben werden mit Matrix1 und Matrix2 angegeben. Mit Seiten wird vorgegeben, ob Abweichungen nach beiden Seiten (2) oder nur nach einer Seite (1) möglich sind. Mit Typ wird der Charakter der Stichproben angegeben:

- 1 zwei Stichproben gleicher Größe
- 2 zwei Stichproben mit unterschiedlicher Größe aber gleicher Standardabweichung
- 3 zwei Stichproben mit unterschiedlicher Größe und unterschiedlicher Standardabweichung

## TVERT()

**Syntax:** TVERT(x;Freiheitsgrade;Seiten)

Die Funktion liefert die Irrtumswahrscheinlichkeit für eine t-verteilte Zufallsvariable. TVERT ist die Umkehrung zu TINV (siehe dort).

Ein Beispiel für die Anwendung ist der Vergleich der Häufigkeit eines Merkmals in einer Stichprobe mit der Wahrscheinlichkeit dieses Merkmals in der Grundgesamtheit. Die Testgröße t ist

$$t = \text{ABS}(z - n \cdot p) / \text{WURZEL}(n \cdot p \cdot (1 - p))$$

Mit z = Häufigkeit des Merkmals in der Stichprobe, p = Wahrscheinlichkeit in der Grundgesamtheit und n = Größe der Stichprobe. Die Zahl der Freiheitsgrade df = n - 1. Setzen Sie diese beiden Größen (t und df) in die Funktion ein, dann erhalten Sie direkt die Wahrscheinlichkeit dafür, dass der Unterschied zwischen Stichprobe und Grundgesamtheit zufällig ist.

$$\text{Wenn } x = \text{TINV}(W;df), \text{ dann ist } W = \text{TVERT}(W;df;2).$$

## VARIANZ()

**Syntax:** VARIANZ(Zahl1;Zahl2;...)

Liefert das Quadrat der Standardabweichung bei einer Stichprobe (vgl. STABW).

## VARIANZA()

**Syntax:** VARIANZA(Wert1;Wert2;...)

Liefert das Quadrat der Standardabweichung bei einer Stichprobe (vgl. STABWA). Textwerte und logische Werte werden mit ausgewertet. Textwerte zählen 0.

## VARIANZEN()

**Syntax:** VARIANZEN(Zahl1;Zahl2;...)

Liefert das Quadrat der Standardabweichung bei einer Grundgesamtheit (vgl. STABWN).

## VARIANZENA()

**Syntax:** VARIANZENA(Wert1;Wert2;...)

Liefert das Quadrat der Standardabweichung bei einer Grundgesamtheit (vgl. STABWNA). Textwerte und logische Werte werden mit ausgewertet. Textwerte zählen 0.

## VARIATION()

**Syntax:** VARIATION(Y\_Werte;X\_Werte;Neue\_x\_Werte;Konstante)

Die Funktion berechnet die Reihe von geordneten Folgen, die mit den Argumenten möglich sind. k ist die Anzahl der Elemente, die aus einer Menge von n Elementen gewählt werden. Im Gegensatz zu KOMBINATIONEN werden in dieser Funktion Reihenfolgen berücksichtigt.

Im Beispiel wird aus einem Patience-Kartenspiel eine bestimmte Folge von vier Karten gezogen. Wie viele mögliche Folgen ständen zur Verfügung? Die Wahrscheinlichkeit, die Folge von Kreuz-As, Pik-As, Herz-As und Karo-As (genau in dieser Reihenfolge) zu ziehen, liegt damit bei 1/6497400.

## WAHRSCHBEREICH()

**Syntax:** WAHRSCHBEREICH(Beob\_Werte; Beob\_Wahrsch; Untergrenze; Obergrenze)

Die Funktion berechnet auf der Basis beobachteter Werte und ihrer Wahrscheinlichkeiten die Wahrscheinlichkeit dafür, dass ein (neuer) Beobachtungswert in ein bestimmtes Intervall fällt.

Für Beob\_Wahrsch wird üblicherweise angenommen

$$p(\text{wert}) = h(\text{wert})/\text{beobachtungen}$$

also der Quotient aus der Häufigkeit, mit der ein Wert auftrat und der Zahl der Beobachtungen. Die Summe aller Wahrscheinlichkeiten muß naturgemäß immer 1 sein.

Das Intervall wird mit Untergrenze und Obergrenze (beide einschließlich) angegeben. Wird Obergrenze weggelassen, dann berechnet die Funktion die Wahrscheinlichkeit dafür, dass ein Beobachtungswert die Größe Untergrenze annimmt.

## WEIBULL()

**Syntax:** WEIBULL(x; Alpha; Beta; Kumuliert)

Die Funktion liefert Werte für eine Zufallsvariable, die einer Weibull-Verteilung gehorcht. Diese Verteilung wird für Haltbarkeitsstatistiken benützt.

Das Quantil wird mit x angegeben, Alpha ist ein Skalenparameter, Beta ein Form- oder Gestaltparameter der Verteilung. Mit Kumuliert läßt sich festlegen, ob die Dichtefunktion (FALSCH) oder die Verteilungsfunktion (WAHR) ausgegeben wird.

## ZÄHLENWENN()

**Syntax:** ZÄHLENWENN(Bereich; Suchkriterien)

Die Funktion gibt die Anzahl der nichtleeren Zellen in einem Bereich wieder, die den angegebenen Kriterien entsprechen. Suchkriterien können eine Zahl, ein Ausdruck oder eine Zeichenfolge sein.

# Matrix- und Bereichsfunktionen

Die unter dieser Rubrik aufgeführten Funktionen dienen in erster Linie der Behandlung von Bezügen: dem Ermitteln der Adressen von Zellen, der Größe von Bereichen, dem Durchsuchen von Bereichen etc. Obwohl diese Funktionen alle in Tabellen verwendet werden können, finden sie zum größten Teil eher in der Makroprogrammierung Anwendung.

## BEREICH.VERSCHIEBEN()

**Syntax:** BEREICH.VERSCHIEBEN(Bezug; Zeilen; Spalten; Höhe; Breite)

Die Funktion liefert einen Bereichsbezug, der um eine mit Zeilen und Spalten festgelegte Zahl von Zeilen und Spalten gegenüber dem mit Bezug festgelegten Bereich verschoben ist. Als Ausgangspunkt dient die linke obere Eckzelle des Bereichs.

Bei positiven Werten werden Zeilen und Spalten nach unten bzw. nach rechts versetzt, bei negativen umgekehrt.

Höhe und Breite sind optionale Argumente und bezeichnen die Größe des neuen Bezugs als Anzahl der Zeilen und Spalten. Wenn Zeilen- und Spaltenanzahl mit dem ursprünglichen Bezug übereinstimmen, müssen für diese Argumente keine Werte eingetragen werden.

## BEREICHE()

**Syntax:** BEREICHE(Bezug)

Die Funktion ermittelt die Anzahl der Bereiche (Zellen oder Matrizen), die in Bezug enthalten sind.

## HYPERLINK()

**Syntax:** HYPERLINK(Hyperlink\_Adresse; Freundlicher\_Name)

**Syntax2:** INDEX(Matrix; Zeile; Spalte)

Die Funktion INDEX liegt in zwei unterschiedlichen Ausprägungen vor. In der ersten Form

dient sie der Ermittlung eines Bezugs aus Bereichen, in der zweiten der Abfrage einer Matrix.

1. Mit der ersten Form der Funktion läßt sich ein Bezug aus Bereichen durch Angabe der entsprechenden Zeile, Spalte und (bei Mehrfachauswahl) dem Bereich erfahren. Mit der Nummer von Zeile und Spalte wird der Ort bezeichnet, dessen Bezug ermittelt werden soll. Bereich als optionales Argument wird dann verwendet, wenn Bezug eine Mehrfachauswahl enthält. Die einzelnen Bereiche werden in Bezug in Klammern gesetzt. Mit Bereich wird durch die Eingabe der entsprechenden Nummer auf diesen Bereich für die Abfrage verwiesen. Wird Bereich nicht angegeben, wird immer der erste Teilbereich genommen. Das Ergebnis der Abfrage wird von Funktionen, die einen Bezug verlangen, als Bezug interpretiert. Funktionen, die einen Wert verlangen, interpretieren das Ergebnis als Wert.
2. Mit der zweiten Form der Funktion lassen sich Werte aus einer Matrix abfragen. Das Ergebnis kann ein einzelner Wert sein oder wiederum eine Matrix. Mit der Angabe von Zeile und Spalte wird der Ort der Matrix festgelegt, dessen Wert ermittelt werden soll. Wollen Sie nicht nur einen einzelnen Wert ermitteln, so läßt sich auch eine Matrix in Form einer Spalte oder einer Zeile auslesen. Wollen Sie eine Spalte auslesen, muß die Angabe für Zeile weggelassen werden, wollen Sie eine Zeile auslesen, gilt das analoge Verfahren. Allerdings muß in diesen Fällen die Funktion selber wie eine Array-Formel eingegeben werden. (Ausgabebereich markieren, Funktion eingeben, beenden mit [Strg]+[Umschalt]+[Enter]).

## **INDIREKT()**

**Syntax:** INDIREKT(Bezug;A1)

Die Funktion ermittelt indirekt den Inhalt einer Zelle, auf die in einer anderen Zelle verwiesen wird. Mit A1 wird angegeben, wie der Eintrag in Bezug steht: in der A1-Schreibweise (WAHR oder weggelassen) oder in der Z1S1-Schreibweise (FALSCH)MTRANS().

## **MTRANS()**

**Syntax:** MTRANS(Matrix)

Gibt einzelne Werte oder die Werte eines angegebenen Bereichs aus einer Pivot-Tabelle zurück.

## **SPALTE()**

**Syntax:** SPALTE(Bezug)

Die Funktion liefert die Spaltennummer des mit Bezug angegebenen Bereichs. Wird Bezug nicht angegeben, ist das Ergebnis die Spaltennummer der Zelle, in der die Funktion steht.

Wird die Funktion als horizontale Matrix eingegeben und ist Bezug ein Bereich, dann werden die entsprechenden Spaltennummern ausgegeben.

## **SPALTEN()**

**Syntax:** SPALTEN(Matrix)

Liefert die Anzahl der Spalten eines Bereichs oder einer Matrix.

## **SVERWEIS()**

**Syntax:** SVERWEIS(Suchkriterium;Matrix;Spaltenindex;Bereich\_Verweis)

Die Funktion liefert einen Wert aus einer Liste von Werten.

Mit Index wird festgelegt, der wievielte Wert als Ergebnis zurückgegeben werden soll. Da bis zu 29 verschiedene Werte eingetragen werden können, ist auch der Eintrag für Index auf 29 begrenzt.

## **WVERWEIS()**

**Syntax:** WVERWEIS(Suchkriterium;Matrix;Zeilenindex;Bereich\_Verweis)

Die Funktion entspricht exakt der Funktion SVERWEIS(); lediglich Zeilen und Spalten sind vertauscht.

## **ZEILE()**

**Syntax:** ZEILE(Bezug)

Die Funktion liefert die Zeilennummer des unter Bezug angegebenen Bezugs. Wird für

Bezug keine Angabe gemacht, wird als Ergebnis die Zeilennummer der Zelle ausgegeben, in der die Funktion steht.

Wird die Funktion als vertikale Matrix eingegeben und ist Bezug ein Bereich, dann werden die entsprechenden Zeilennummern ausgegeben.

## **ZEILEN()**

**Syntax:** ZEILEN(Matrix)

Liefert die Anzahl der Zeilen eines Bereichs oder einer Matrix.

## Referenz der Datenbankfunktionen

### **DBANZAHL()**

**Syntax:** DBANZAHL(Datenbank; Datenbankfeld; Suchkriterien)

Die Funktion ermittelt die Anzahl der Datensätze, die mit den Suchkriterien übereinstimmen.

Wird für Datenbankfeld eine Angabe gemacht, dann wird auch das angegebene Feld nach dem Kriterium durchsucht, selbst wenn es im Kriterienbereich nicht unter diesem Feld eingetragen ist.

### **DBANZAHL2()**

**Syntax:** DBANZAHL2(Datenbank; Datenbankfeld; Suchkriterien)

Die Funktion zählt die Datensätze, die mit den Suchkriterien übereinstimmen und unter Datenbank keine leeren Zellen haben.

### **DBAUSZUG()**

**Syntax:** DBAUSZUG(Datenbank; Datenbankfeld; Suchkriterien)

Die Funktion ermittelt den Inhalt des mit Datenbankfeld angegebenen Feldes des Datensatzes, der mit den Suchkriterien übereinstimmt. Findet die Funktion keinen Datensatz, der mit dem Suchkriterium übereinstimmt, wird die Fehlermeldung *#WERT* ausgegeben. Findet die Funktion mehr als einen Datensatz, der mit dem Kriterium übereinstimmt, wird die Fehlermeldung *#Zahl!* ausgegeben.

### **DBMAX()**

**Syntax:** DBMAX(Datenbank; Datenbankfeld; Suchkriterien)

Die Funktion liefert den größten Wert in einem Datenbankfeld, der den angegebenen Kriterien entspricht.

### **DBMIN()**

**Syntax:** DBMIN(Datenbank; Datenbankfeld; Suchkriterien)

Wie DBMAX(), nur dass nach dem kleinsten Wert gesucht wird.

### **DBMITTELWERT()**

**Syntax:** DBMITTELWERT(Datenbank; Datenbankfeld; Suchkriterien)

Die Funktion liefert den Mittelwert aller Werte in der Spalte Datenbankfeld, deren Datensätze die Suchkriterien erfüllen.

Leere Zellen bleiben unberücksichtigt.

### **DBPRODUKT()**

**Syntax:** DBPRODUKT(Datenbank; Datenbankfeld; Suchkriterien)

Die Funktion multipliziert alle Werte in der Spalte Datenbankfeld derjenigen Datensätze, die die Suchkriterien erfüllen.

### **DBSTDABW()**

**Syntax:** DBSTDABW(Datenbank; Datenbankfeld; Suchkriterien)

Die Funktion berechnet die Standardabweichung der Spalte Datenbankfeld für diejenigen Datensätze, die die Suchkriterien erfüllen. Die Datensätze werden als Stichprobe behandelt.

## **DBSTDABWN()**

**Syntax:** DBSTDABWN(Datenbank; Datenbankfeld; Suchkriterien)  
Wie DBSTDABW(), die Daten werden aber als Grundgesamtheit behandelt.

## **DBSUMME()**

**Syntax:** DBSUMME(Datenbank; Datenbankfeld; Suchkriterien)

Die Funktion summiert alle Werte derjenigen Datensätze in der Spalte Datenbankfeld, die die Suchkriterien erfüllen.

## **DBVARIANZ()**

**Syntax:** DBVARIANZ(Datenbank; Datenbankfeld; Suchkriterien)

Die Funktion berechnet die Varianz der Spalte Datenbankfeld für diejenigen Datensätze, die die Suchkriterien erfüllen. Die Datensätze werden als Stichprobe behandelt.

## **DBVARIANZEN()**

**Syntax:** DBVARIANZEN(Datenbank; Datenbankfeld; Suchkriterien)

Wie DBVARIANZ(), die Daten werden aber als Grundgesamtheit behandelt.

# Textfunktionen

Gelegentlich ist es sinnvoll, eine Zeichenfolge in einer Zelle durch eine Formel zu erzeugen, die verschiedene Zeichen oder Zeichenfolgen verknüpft. Angenommen, Sie wollen Artikelnummern um zwei Zeichen erweitern, die die Warengruppe beinhalten. Wenn Sie in die Zelle C9

=C5&C8

eintragen, wird der Inhalt von Zelle C5 mit dem Inhalt von Zelle C8 verkettet. Wenn C5 das Warengruppenkennzeichen "PX" enthält und C8 die Artikelnummer 3370086', ist das Ergebnis in Zelle C9 "PX3370086". Wenn die bisherige Artikelnummer in C8 als Zahl und nicht als Zeichenfolge eingetragen worden ist, wird die Zahl in eine Zeichenfolge umgewandelt. Anschließend können Sie diese Formel noch in ihr Ergebnis verwandeln. Wollen Sie zwischen den beiden Textelementen eine Leertaste sehen, müssen Sie folgendes schreiben:

=C5&" "&C8

## Sortierung durch Textfunktionen

Sollen umgekehrt bei Artikelbezeichnungen, die immer auf den zwei ersten oder letzten Stellen ein Kennzeichen der Warengruppe enthalten, diese abgelesen werden, etwa um eine Liste danach zu sortieren, können Sie mit einer Textfunktion die beiden Stellen herausziehen.

=RECHTS(C9;2)

liefert die beiden letzten Stellen,

=LINKS(C9;2)

die beiden ersten Stellen.

## Logische Werte in Texte aufnehmen

Auch eine Zelle mit einem logischen Wert kann mit dem Inhalt einer anderen Zelle zu einem Text verkettet werden.

="Die Behauptung, der Umsatz steigt, ist "&C9

ergibt den Satz "Die Behauptung ... ist FALSCH", wenn die Bedingung in C9 nicht erfüllt ist.

## Verknüpfung von Text mit Datum

Etwas schwieriger ist die Verknüpfung von Text mit einem Datum. Wenn Sie einen Bezug auf ein Datumsfeld benutzen, liefert Excel anstelle des Datums die serielle Zahl, mit der das Datum intern dargestellt wird. Das Problem ist lösbar, wenn Sie mit den anderen Datumsfunktionen arbeiten. Hier ein `84`

Diese Funktion liefert den ANSI-Code des ersten Zeichens in der Zeichenkette, die als Argument eingegeben wird (direkt oder als Bezug). Bezieht sich das Argument auf eine leere oder eine numerische Zelle, liefert die Funktion die Meldung *#WERT*.

### DM()

**Syntax:** DM(Zahl;Dezimalstellen)

Die Funktion wandelt eine Zahl in eine Zeichenkette im Währungsformat um. Durch das optionale Argument Dezimalstellen kann die Anzahl der Ziffern rechts vom Komma festgelegt werden, die dabei gerundet und nicht abgeschnitten werden. Wird das Argument nicht angegeben, dann wird es mit 2 angenommen. Wird ein negativer Wert eingegeben, wird auf entsprechende Stellen vor dem Komma gerundet.

### ERSETZEN()

**Syntax:** ERSETZEN(Alter\_Text;Erstes\_Zeichen;Anzahl\_Zeichen;Neuer\_Text)

Mit dieser Funktion können in einer Zeichenfolge ein oder mehrere Zeichen durch neue Zeichen überschrieben werden. Die Funktion hat vier Argumente:

1. Alter\_Text ist die Zeichenfolge, die durch eine andere ersetzt werden soll.
2. Erstes\_Zeichen legt eine Zahl fest, an welcher Stelle in der Zeichenkette mit dem Ersetzen zu beginnen ist. Das erste Zeichen der Zeichenfolge wird mit 1 gezählt.
3. Anzahl\_Zeichen gibt an, wie viele Zeichen von Beginn an ersetzt, d. h. aus der alten Zeichenfolge gelöscht werden sollen.
4. Neuer\_Text ist die Zeichenfolge, die anstelle der herausgenommenen Zeichen der alten Zeichenfolge eingefügt werden soll.

### FEST()

**Syntax:** FEST(Zahl;Dezimalstellen;Keine\_Punkte)

Die Funktion wandelt einen numerischen Wert in eine Zeichenfolge um. Die Anzahl der gewünschten Dezimalstellen muß eingegeben werden, das Ergebnis erscheint gerundet. Geben Sie das Argument Dezimalstellen nicht an, so wird es als 2 angenommen, wird es negativ eingegeben, dann wird auf die entsprechenden Stellen vor dem Komma gerundet.

Keine\_Punkte ist ein Wahrheitswert, der darüber entscheidet, ob eine Tausender-Abtrennung mittels Punkt vorgenommen werden soll. Wird kein Wahrheitswert oder FALSCH angegeben, dann wird die Tausenderabtrennung vorgenommen, andernfalls wird der Punkt nicht mit ausgegeben.

Mit der FEST-Funktion ist es möglich, Zeichenfolgen zusammenzustellen, die Text und Zahlen kombinieren.

### FINDEN()

**Syntax:** FINDEN(Suchtext;Text;Erstes\_Zeichen)

Die Funktion überprüft, ob eine Suchzeichenfolge (Suchtext) in einer anderen Zeichenfolge (Text) vorkommt.

Die Suche beginnt bei der Position in der Zeichenfolge, die durch den Wert für Erstes\_Zeichen bestimmt ist. Ergebnis ist die Position von Suchtext in Text. Wird der Suchtext in der Zeichenfolge nicht gefunden, wird die Meldung *#WERT* ausgegeben. Die Funktion unterscheidet zwischen Groß- und Kleinschreibung.

### GLÄTTEN()

**Syntax:** GLÄTTEN(Text)

Entfernt führende Leerzeichen und Leerzeichen zwischen Wörtern in Text, so dass jeweils nur ein Leerzeichen zur Worttrennung stehenbleibt.



Die Funktion ist nützlich, um überflüssige Leerzeichen aufzuspüren und um mögliche Fehler beim Sortieren zu vermeiden.

## **GROSS()**

**Syntax:** GROSS(Text)

Die Funktion GROSS() wandelt alle Buchstaben aus Text in Großbuchstaben um.

## **GROSS2()**

**Syntax:** GROSS2(Text)

Die Funktion wandelt jeweils den ersten Buchstaben eines jeden Wortes in der Zeichenfolge in Großbuchstaben um, die weiteren Zeichen in Kleinbuchstaben. Die Funktion kann zur Erhöhung der Lesbarkeit von Texteinträgen benutzt werden. Sie ist von Nutzen, wenn z. B. Daten von anderen Programmen oder Rechnern übernommen worden sind.

## **IDENTISCH()**

**Syntax:** IDENTISCH(Text1;Text2)

Vergleicht zwei Zeichenfolgen und prüft, ob sie exakt übereinstimmen. Nur dann wird als logischer Wert WAHR geliefert, im anderen Fall FALSCH. Leerzeichen und Groß- bzw. Kleinschreibung werden in den Vergleich einbezogen.

## **KLEIN()**

**Syntax:** KLEIN(Text)

Wandelt alle Buchstaben von Text in Kleinbuchstaben um.

## **LÄNGE()**

**Syntax:** LÄNGE(Text)

Liefert die Zahl der Zeichen (einschließlich Leerzeichen) von Text.

Ist das Argument der Funktion eine Zelladresse, wird als Ergebnis 0 ausgegeben, wenn die Zelle leer ist. Numerische Zeichen werden wie Text behandelt und auf gleiche Weise gezählt. Zur Verwendung kommt die Funktion bei der Prüfung von Einträgen, für die eine bestimmte Länge vorgesehen ist, z. B. Postleitzahlen.

## **LINKS()**

**Syntax:** LINKS(Text;Anzahl\_Zeichen)

Die Funktion liefert als Ergebnis die eingetragene Anzahl\_Zeichen vom Beginn der Zeichenfolge an, der Rest wird abgeschnitten.

Dies ist eine einfache Möglichkeit, Zeichenfolgen zu kürzen, etwa um Kurzbezeichnungen zu produzieren.

Wird für Anzahl\_Zeichen kein Wert angegeben, so wird 1 vorgegeben.

## **RECHTS()**

**Syntax:** RECHTS(Text;Anzahl\_Zeichen)

Liefert die angegebene Anzahl\_Zeichen vom rechten Ende der Zeichenfolge und gibt diese als Ergebnis aus.

## **SÄUBERN()**

**Syntax:** SÄUBERN(Text)

Die Funktion entfernt aus Text alle nicht druckbaren Zeichen. In der Regel sind dies Steuerzeichen, die mit Texten aus anderen Programmen übernommen wurden und die nicht ausgedruckt werden sollen.

## **SUCHEN()**

**Syntax:** SUCHEN(Suchtext;Text;Erstes\_Zeichen)

Die Funktion liefert die Position des Zeichens oder der Zeichenfolge (Suchtext), die in einem Text gesucht wird. Die Suche startet bei der Position Erstes\_Zeichen, wenn nichts angegeben wird bei der Position 1.

Im Gegensatz zu FINDEN ignoriert diese Funktion die Groß- und Kleinschreibung.

## TO

**Syntax:** T(Wert)

Die Funktion überprüft, ob sich in einer Zelle eine Zahl oder eine Zeichenfolge befindet. Wird eine Zahl gefunden, dann gibt die Funktion Leertext aus, wird Text gefunden, wird dieser wieder ausgegeben.

## TEILO

**Syntax:** TEIL(Text;Erstes\_Zeichen;Anzahl\_Zeichen)

Die Funktion ermittelt aus einer Zeichenkette (Text) eine Unterkette ab einer festgelegten Position (Erstes\_Zeichen) und gibt sie in der gewünschten Länge (Anzahl\_Zeichen) aus.

## TEXT()

**Syntax:** TEXT(Wert;Textformat)

Die Funktion wandelt einen numerischen Wert in einen Text um und verwendet dabei ein Excel-Zahlenformat (einschließlich Datumsformaten) aus dem Menü Format.

## VERKETTEN()

**Syntax:** VERKETTEN(Text1;Text2;...)

Text1, Text2 etc. (bis zu 29 Zeichenfolgen) können mittels dieser Funktion miteinander verknüpft werden. Die Funktion erfüllt somit die gleichen Aufgaben wie der Operator „&“.

## WECHSELN()

**Syntax:** WECHSELN(Text;Alter\_Text;Neuer\_Text;Häufigkeit)

Die Funktion sucht in einem Text nach einer Zeichenkette (Alter\_Text), um diese Zeichenkette dann durch eine andere (Neuer\_Text) zu ersetzen.

Das optionale Argument Häufigkeit bestimmt, wie oft der Vorgang des Ersetzens ablaufen soll. Wird das Argument mit 1 gesetzt, dann wird nur die erste Fundstelle geändert; wird das Argument nicht belegt, dann werden alle Fundstellen geändert.

## WERT()

**Syntax:** WERT(Text)

Die Funktion wandelt einen Text in einen Wert um und gibt als Ergebnis dann die entsprechende Zahl aus. Der Analysetext darf dabei die entsprechenden Zeichen aus einem Excel-Zahlenformat enthalten. Zeichenketten, die als Datums- oder Zeitformate interpretiert werden können, werden als die jeweilige serielle Zahl ausgegeben. Läßt sich der Text nicht als Zahl interpretieren, gibt die Funktion #WERT aus.

## WIEDERHOLEN()

**Syntax:** WIEDERHOLEN(Text;Multiplikator)

Die Funktion trägt die angegebene Zeichenfolge so oft hintereinander in die Zelle ein, wie durch den Multiplikator angegeben. Die Zeichenfolge kann aus mehr als einem Zeichen bestehen.

## ZEICHEN()

**Syntax:** ZEICHEN(Zahl)

Die Funktion ist die Umkehrung der CODE-Funktion. Sie liefert für eine bestimmte Codezahl das entsprechende Zeichen.

# Logische Funktionen

Logische Funktionen werden benutzt, wenn es darum geht, zu prüfen, ob bestimmte Tatsachen oder Bedingungen gegeben sind oder nicht. Eine logische Formel kann nicht nur eine Bedingung enthalten, sondern auch mehrere gleichzeitig. Diese Bedingungen können entweder alternativ oder additiv formuliert werden. Alternative Bedingungen werden mit der ODER-Funktion verknüpft, additive Bedingungen werden mit der UND-Funktion. Hier einige Beispiele:

=UND (B6>B7 ; B9>B10)

=UND(A10>50;A10<100)

In diesen Fällen ist die Bedingung jeweils nur erfüllt, also wahr, wenn beide Teilbedingungen erfüllt sind.

=ODER (C9=12 ; C9=24)

=ODER(MONAT="Mai" ; MONAT="Juni" ; MONAT="Okt")

Bei diesen Formeln ist das Ergebnis WAHR, wenn die Zelle mit dem Namen "MONAT" einen der drei Monatsnamen enthält. Die Funktion NICHT() verneint einen Vergleich. Ein FALSCH, wenn der Gewinn = 300000

Die WENN-Funktion prüft zunächst, ob eine bestimmte Bedingung erfüllt, also wahr ist. Trifft dieses zu, wird die WAHR-Anweisung ausgeführt und bestimmt das Ergebnis der Formel. Trifft dies nicht zu, wird zur FALSCH-Anweisung verzweigt und diese ausgeführt. Die Funktion hat deshalb drei Argumente, die in der vorgeschriebenen Reihenfolge eingetragen werden müssen, weil sonst unerwünschte Ergebnisse produziert werden.

1. Eine Bedingung in Form eines logischen Ausdrucks.
2. Eine Anweisung für den Fall, dass die Bedingung erfüllt ist.
3. Eine Anweisung für den Fall, dass die Bedingung nicht erfüllt ist.

Die WENN-Funktion kann generell für zwei ganz unterschiedliche Zwecke eingesetzt werden.

1. Die Funktion erlaubt die Durchführung von Prüfungen. z. B. kann mit dieser Funktion die Frage beantwortet werden, ob bestimmte Grenzwerte überschritten, ob Abweichungen von einer Norm vorliegen oder bestimmte Zielwerte erreicht worden sind.

2. Mit der WENN-Funktion kann Excel veranlasst werden, unterschiedliche Operationen durchzuführen, je nachdem, ob eine Bedingung erfüllt ist. Das entspricht etwa einer Verzweigung, die in allen Computersprachen zu den Grundoperationen gehört.

Allerdings gilt die Verzweigung immer nur in bezug auf das Ergebnis in der Zelle selbst. Es ist nicht möglich, mit der WENN-Funktion direkt einen unterschiedlichen Wert in eine andere Zelle einzutragen. Es ist zwar möglich, etwa in der Zelle B3 zu

schreiben

WENN B1>1000;B2=100;B2=200)

Ist B1 tatsächlich größer 1000, bedeutet diese Formel aber lediglich, dass Excel nun prüft, ob die Zelle B2 tatsächlich den Wert 100 enthält. Ist das der Fall, ist das Ergebnis in Zelle B3 der Wert WAHR.

Eine direkte Wert-Zuweisung an eine Zelle, die dann eventuell auch Werte, die bisher in dieser Zelle eingetragen waren, überschreibt, kann nur über Makrobefehle geleistet werden. Das schließt natürlich nicht aus, dass eine andere Zelle durch einen Adressbezug den Wert aus der Zelle, in der die WENN-Funktion steht, übernimmt. Wenn Sie in B3

=WENN)B1>1000;100;200)

schreiben und in Zelle B2

=B3

erhalten Sie das Ergebnis, dass Sie mit der ersten Formel erreichen wollten. In den folgenden Abschnitten soll die Nützlichkeit der WENN-Funktion an einigen Beispielen verdeutlicht werden.

## Texte automatisch anpassen

In der abgebildeten Tabelle sind Kunden-Umsätze einer Firma für 2 Jahre eingetragen, in der Spalte B der Umsatz für 1999, in der Spalte C der für 2000. Die Kunden, deren

Umsatz gegenüber dem Vorjahr um mehr als 20% abgesunken ist, sollen in der Spalte D so markiert werden, dass sie auf einen Blick zu erkennen sind. Die Formel kann in der Zelle D8 eingetragen und dann die Spalte hinunterkopiert werden. Sie könnte lauten:

```
=WENN(C8<(B8*0,8); „Umsatzrückgang“;““)
```

Bei den Kunden, deren Umsatz im Jahr 2000 tatsächlich um mehr als 20% gesunken ist, erscheint in der Spalte D der Hinweis: „Umsatzrückgang“. Bei allen anderen Kunden bleibt die Zelle in der Spalte D als Ergebnis der Formel leer.

Während es im letzten Fall in erster Linie um die Prüfung der Entwicklung der Kundenumsätze ging, soll die nächste Formel zeigen, wie die WENN-Funktion benutzt werden kann, um auf verschiedene Zustände unterschiedlich zu reagieren. Die Firma könnte auf die Idee kommen, die Höhe eines Bonus zum Jahresende von der Umsatzentwicklung abhängig zu machen. Die Kunden, deren Umsatz über 250000 € liegt, sollen einen Bonus von 0,5 % erhalten, die anderen gehen leer aus. Die Formel für diesen Fall:

```
=WENN(C5>250000;C5*0,05;0)
```

Sowohl bei der Formulierung der Bedingungen, als auch bei der Formulierung der WAHR- und FALSCH-Anweisungen können natürlich alle sinnvollen Kombinationen von Operatoren und Funktionen benutzt werden.

## Bedingte Textanzeige

Besteht die Möglichkeit, in einem Arbeitsblatt Texte nur unter bestimmten Voraussetzungen zu drucken? Die WENN-Funktion bietet sich auch in diesem Fall an. Angenommen, das Arbeitsblatt wird zur Aufbereitung und zum Druck von Rechnungen benutzt. Auf den Dezember-Rechnungen soll im Rechnungsfuß ein Hinweis auf ein spezielles Weihnachtsangebot erscheinen. Die Lösung könnte folgende Formel sein:

```
=WENN(MONAT(JETZT)='Sonderrabatt von 20% für alle Orientteppiche';““)
```

## Prüfung von Texten

Die WENN-Funktion kann selbstverständlich nicht nur zur Prüfung numerischer Zellen benutzt werden. Zellen, die Texte oder Zeichenfolgen enthalten, können ebenfalls befragt werden. Der Text kann direkt eingetragen sein oder indirekt über einen Zellbezug oder als Ergebnis einer Formel, die als Ergebnis eine Zeichenfolge bereitstellt. Hier ein einfaches FALSCH

Diese Funktion legt den Wahrheitswert für FALSCH in der Zelle ab oder übergibt ihn an eine andere Funktion.

Die Funktionen WAHR und FALSCH können in einer WENN-Funktion benutzt werden, um anzuzeigen, ob die Bedingung, die diese Funktion prüft, erfüllt ist. Das kann sinnvoll sein, wenn bei einer komplexen Bedingung nicht gleich ersichtlich ist, ob die Bedingung erfüllt ist oder nicht.

### NICHT()

**Syntax:** NICHT(Wahrheitswert)

Durch diese Funktion wird der Wert des Wahrheitswertes umgekehrt.

### ODER()

**Syntax:** ODER(Wahrheitswert1;Wahrheitswert2;...)

Die Funktion vergleicht bis zu 30 Argumente miteinander. Sind alle Argumente FALSCH, so liefert die Funktion FALSCH, andernfalls WAHR.

### UND()

**Syntax:** UND(Wahrheitswert1;Wahrheitswert2;...)

Die Funktion vergleicht bis zu 30 Argumente miteinander. Sind alle Argumente WAHR, so wird als Ergebnis WAHR zurückgegeben, andernfalls ist das Ergebnis FALSCH.

## **WAHR()**

**Syntax:** WAHR()

Ebenso wie bei der Funktion FALSCH kann mit dieser Funktion der Wahrheitswert WAHR in ein Feld eingetragen werden oder als Argument an eine Funktion übergeben werden.

## **WENN()**

**Syntax:** WENN(Prüfung; Dann\_Wert; Sonst\_Wert)

Die Funktion liefert in Abhängigkeit einer Bedingung je nach Wahrheitsprüfung unterschiedliche Resultate. Die WENN-Funktion prüft zunächst, ob eine bestimmte Bedingung (Prüfung) erfüllt, also wahr ist. Trifft dies zu, gibt die Funktion den Dann\_Wert aus, trifft es nicht zu, wird der Sonst\_Wert ausgegeben.

Die Funktion erlaubt die Durchführung von Prüfungen, z. B. kann mit dieser Funktion die Frage beantwortet werden, ob bestimmte Grenzwerte überschritten oder bestimmte Zielwerte erreicht worden sind. Es kann auch veranlaßt werden, unterschiedliche Werte in eine Zelle einzutragen oder unterschiedliche Berechnungen durchzuführen, je nachdem, ob eine Bedingung erfüllt ist oder nicht.

Als Ergebnis der Wahrheitsprüfung kann eine Verschachtelung durch weitere Verwendung der WENN-Funktion erfolgen.

# Informationsfunktionen

Die meisten der Informationsfunktionen werden nur gelegentlich innerhalb von Tabellen, sonst eher in Makros benötigt. Sie sind deshalb auch dort besprochen.

Das Hauptanwendungsgebiet von Informationsfunktionen von Tabellen ist, in Verbindung mit WENN() Ergebnisse von Berechnungen von den Inhalten bestimmter Zellen abhängig zu machen.

## **FEHLER.TYP()**

**Syntax:** FEHLER.TYP(Fehlerwert)

Siehe Besprechung der Funktion bei den Makrofunktionen.

## **INFO()**

**Syntax:** INFO(Typ)

Siehe Besprechung der Funktion bei den Makrofunktionen.

## **ISTBEZUG()**

**Syntax:** ISTBEZUG(Wert)

Die Funktion testet auf einen gültigen Bereichsnamen oder eine Bereichsadresse. Ob vorhanden oder nicht vorhanden, wird durch die Wahrheitswerte WAHR oder FALSCH ausgedrückt.

## **ISTFEHL()**

**Syntax:** ISTFEHL(Wert)

Die Funktion testet, ob ein bestimmtes Feld einen Fehlerwert enthält (Ausnahme: #N/A). Ist ein Fehlerwert vorhanden, wird WAHR ausgegeben, in allen anderen Fällen FALSCH.

## **ISTFEHLER()**

**Syntax:** ISTFEHLER(Wert)

Die Funktion testet, ob ein bestimmtes Feld einen Fehlerwert enthält (einschließlich #N/A). Ist ein Fehlerwert vorhanden, wird WAHR ausgegeben, in allen anderen Fällen FALSCH.

## **ISTGERADE()**

**Syntax:** ISTGERADE(Wert)

Die Funktion testet, ob Wert durch 2 ganzzahlig teilbar ist und liefert WAHR, wenn das der Fall ist, vgl. ISTUNGERADE().

## **ISTKTEXT()**

**Syntax:** ISTKTEXT(Wert)

Die Funktion testet, ob in einem Feld Text oder kein Text vorhanden ist. Ist Text vorhanden, so wird als Ergebnis FALSCH ausgegeben, andernfalls und bei einem leeren Feld WAHR.

## **ISTLEER()**

**Syntax:** ISTLEER(Wert)

Die Funktion testet, ob ein Feld leer ist. Wenn das der Fall ist, wird WAHR ausgegeben, andernfalls FALSCH.

## **ISTLOG()**

**Syntax:** ISTLOG(Wert)

Die Funktion überprüft, ob in einem Feld ein Wahrheitswert eingetragen ist. Ist ein Wahrheitswert oder eine Funktion, die zu einem Wahrheitswert führt, eingetragen, wird WAHR ausgegeben, andernfalls FALSCH.

## **ISTNV()**

**Syntax:** ISTNV(Wert)

Die Funktion überprüft, ob eine Zelle den Fehlerwert #NV enthält. Das Ergebnis wird als Wahrheitswert ausgegeben.

## **ISTTEXT()**

**Syntax:** ISTTEXT(Wert)

Die Funktion prüft, ob der Wert in der Zelle eine Zeichenfolge ist. Ist in der Zelle ein Text vorhanden, wird WAHR ausgegeben.

## **ISTUNGERADE()**

**Syntax:** ISTUNGERADE(Wert)

Die Funktion testet, ob Wert durch 2 ganzzahlig teilbar ist und liefert FALSCH, wenn das der Fall ist, vgl. ISTGERADE().

## **ISTZAHL()**

**Syntax:** ISTZAHL(Wert)

Die Funktion prüft, ob das Feld eine Zahl enthält. Ist eine Zahl enthalten, dann wird als Ergebnis WAHR ausgegeben, enthält das Feld keine Zahl (auch ein leeres Feld), wird FALSCH ausgegeben.

## **N()**

**Syntax:** N(Wert)

Die Funktion gibt den Eintrag in der Zelle als einen in eine Zahl gewandelten Wert zurück. Findet die Funktion eine Zahl vor, so wird diese unverändert als Zahl wieder ausgegeben. Der Wahrheitswert WAHR wird als 1 wiedergegeben, FALSCH als 0, Daten und Zeiten in einem der Excel-Formate werden als entsprechende serielle Zahlen ausgegeben.

Alle anderen Einträge, Text, Fehlerwerte, leere Zellen, werden als 0 ausgegeben.

## **NV()**

**Syntax:** NV()

Die Funktion trägt die Fehlermeldung #NV (**N**icht **V**orhanden) in die Zelle ein, von der aus die Funktion aufgerufen wird. Für einen Bereich kann die Funktion auch als Matrixformel eingegeben werden. Formeln, die sich auf ein Feld beziehen, in dem #NV eingetragen ist, liefern auch #NV. Damit läßt sich die Funktion zur besonders auffälligen Kennzeichnung von Feldern einsetzen.

Obwohl die Funktion keine Argumente verlangt, ist es notwendig, die beiden Klammern mit anzugeben, damit Excel die Funktion erkennt.

## **TYP()**

**Syntax:** TYP(Wert)

Liefert eine Kennziffer für den Typus des mit Wert angegebenen Eintrags: 1 (Zahl), 2 (Text), 4 (Wahrheitswert), 16 (Fehlerwert), 64 (Matrix).

## **ZELLE()**

**Syntax:** ZELLE(Infotyp;Bezug)

Die Funktion gibt vielfältige Informationen über eine Zelle bzw. die Zelle in der linken oberen Ecke eines Bezugs. Unter Infotyp wird ein Textstring eingetragen, der Informationen abfragt, die durch diesen Text definiert werden, Bezug ist die Zelle, über die Informationen eingeholt werden sollen.

# Technische Funktionen

In dieser Gruppe finden Sie verschiedene Varianten der Bessel-Funktion, die insbesondere für Schwingungsberechnungen benutzt werden. Dazu kommen eine ganze Reihe von Umwandlungsfunktionen, um Werte zwischen den verschiedenen Zahlensystemen auszutauschen. Eine andere Gruppe hat mit den komplexen Zahlen zu tun.

Ganz praktisch ist die Funktion *UMWANDELN()*, mit der zwischen verschiedenen Maßeinheiten umgerechnet werden kann. Sie können sich damit leicht eine Tabelle aufbauen, die die gewünschten Umwandlungen für die Maßeinheiten liefert, mit denen Sie öfters zu tun haben.

## Referenz der technischen Funktionen

### **BESSELI()**

**Syntax:** BESSELI(x;n)

Liefert Werte der modifizierten Besselfunktion.

Bei den Besselschen Funktionen (auch unter dem Namen Zylinderfunktion bekannt) handelt es sich um mehrere miteinander verwandte Funktionen, die in Physik und Technik besonders bei Schwingungsberechnungen benutzt werden. Die Besselschen Funktionen finden Sie in mathematischen Tabellenwerken unter

J0(x), J1(x)..	BESSELJ()
I0(x), I1(x)..	BESSELI()
K0(x), K1(x)..	BESSELK()
Y0(x), Y1(x)..	BESSELY()

meist aber nur für die nullte und erste Ordnung. Die Funktionen, die Excel 2003 bereitstellt, gestatten eine Berechnung für andere Ordnungen (mit n angegeben).

### **BESSELJ()**

**Syntax:** BESSELJ(x;n)

Vgl. BESSELI().

### **BESSELK()**

**Syntax:** BESSELK(x;n)

Vgl. BESSELI().

### **BESSELY()**

**Syntax:** BESSELY(x;n)

Vgl. BESSELI().

## **BININDEZ()**

**Syntax:** BININDEZ(Zahl)

Liefert den Dezimalwert einer binären Zahl. Die binäre Zahl darf höchstens zehn Zeichen (0 oder 1) lang sein; bei zehn Zeichen ist das erste Zeichen das Vorzeichen-Bit (1 = negativ).

Näheres zur Zahlenumwandlung siehe DEZINBIN().

## **BININHEX()**

**Syntax:** BININHEX(Zahl; Stellen)

Liefert den Hexadezimalwert einer binären Zahl, vgl. BININDEZ().

Stellen gibt an, wieviel Stellen angezeigt werden. Ohne Angabe werden nur die notwendigen Stellen angezeigt, ist Stellen größer als diese Anzahl, dann werden führende Nullen ausgegeben.

## **BININOKT()**

**Syntax:** BININOKT(Zahl; Stellen)

Liefert den Oktalwert einer binären Zahl, vgl. BININDEZ().

Stellen gibt an, wieviel Stellen angezeigt werden. Ohne Angabe werden nur die notwendigen Stellen angezeigt, ist Stellen größer als diese Anzahl, dann werden führende Nullen ausgegeben.

## **DELTA()**

**Syntax:** DELTA(Zahl1; Zahl2)

Liefert 1 wenn Zahl1 und Zahl2 gleich sind, sonst 0.

## **DEZINBIN()**

**Syntax:** DEZINBIN(Zahl; Stellen)

Liefert den Binärwert einer Dezimalzahl.

Die Zahl der ausgegebenen Stellen kann mit Stellen festgelegt werden. Ohne Angabe werden nur die notwendigen Stellen ausgegeben, ist Stellen größer als diese Zahl, dann wird die Binärzahl mit führenden Nullen ausgegeben.

Wie die anderen Funktionen zur Zahlenumwandlung (BIN..., HEX..., OKT... und die anderen DEZ ...-Funktionen) dient DEZINBIN der Konversion von Zahlen zwischen den verschiedenen Zahlensystemen.

Zur Erinnerung: Das Dezimalsystem verfügt über 10 Ziffernzeichen (0 bis 9). Kommen Sie beim Zählen einer Zahl nach der 9 an, dann wird die erste Ziffer genommen und eine 0 angehängt. Sie zählen also im Binärsystem 0, 1, 10, 11, 100, im Oktalsystem 1, 2 ... 7, 10, 11, ..., 16, 20, im Hexadezimalsystem 1, 2, ... 9; A; B; C; D; E; F; 10... Im Hexadezimalsystem müssen Sie zu den bekannten Ziffern noch A bis F hinzunehmen, um insgesamt auf 16 Ziffern zu kommen.

## **DEZINHEX()**

**Syntax:** DEZINHEX(Zahl; Stellen)

Liefert den Hexadezimalwert einer Dezimalzahl, vgl. DEZINBIN().

Die Zahl der ausgegebenen Stellen kann mit Stellen festgelegt werden. Ohne Angabe werden nur die notwendigen Stellen ausgegeben, ist Stellen größer als diese Zahl, dann wird die Hexadezimalzahl mit führenden Nullen ausgegeben.

## **DEZINOKT()**

**Syntax:** DEZINOKT(Zahl; Stellen)

Liefert den Oktalwert einer Dezimalzahl, vgl. DEZINBIN().

Die Zahl der ausgegebenen Stellen kann mit Stellen festgelegt werden. Ohne Angabe werden nur die notwendigen Stellen ausgegeben, ist Stellen größer als diese Zahl, dann wird die Binärzahl mit führenden Nullen ausgegeben.

## **ERF/GAUSSFEHLER()**

**Syntax:** GAUSSFEHLER(Untere\_Grenze; Obere\_Grenze)



Liefert Werte des Gaußschen Fehlerintegrals. Die Funktion gehört eigentlich zu den Verteilungsfunktionen der Statistik.

Mit Untere\_Grenze (0 oder größer) wird der Wert angegeben, ab dem die Funktion integriert werden soll; mit Obere\_Grenze der Wert, bis zu dem integriert wird. Wird Obere\_Grenze nicht angegeben, wird die Integration von 0 bis Untere\_Grenze durchgeführt.

## ERFC()

**Syntax:** ERFC(Untere\_Grenze)

Liefert komplementäre Werte des Gaußschen Fehlerintegrals, vgl. GAUSSFEHLER().

Es gilt die Beziehung:

$$1 - \text{GAUSSFEHLER}(x) = \text{ERFC}(x)$$

## GGANZZAHL()

**Syntax:** GGANZZAHL(Zahl; Schritt)

Liefert 1, wenn Zahl größer oder gleich ist als die mit Schritt angegebene Zahl, sonst 0.

## HEXINBIN()

**Syntax:** HEXINBIN(Zahl; Stellen)

Liefert den Binärwert einer hexadezimalen Zahl, vgl. DEZINBIN().

Stellen gibt an, wieviel Stellen angezeigt werden. Ohne Angabe werden nur die notwendigen Stellen angezeigt, ist Stellen größer als diese Anzahl, dann werden führende Nullen ausgegeben.

## HEXINDEZ()

**Syntax:** HEXINDEZ(Zahl)

Liefert den Dezimalwert einer hexadezimalen Zahl, vgl. DEZINBIN().

## HEXINOKT()

**Syntax:** HEXINOKT(Zahl; Stellen)

Liefert den Oktalwert einer hexadezimalen Zahl, vgl. DEZINBIN().

Stellen gibt an, wieviel Stellen angezeigt werden. Ohne Angabe werden nur die notwendigen Stellen angezeigt, ist Stellen größer als diese Anzahl, dann werden führende Nullen ausgegeben.

## IMABS()

**Syntax:** IMABS(Komplexe\_Zahl)

Liefert den Betrag einer komplexen Zahl. Die Komplexe\_Zahl muß als Zeichenkette der Form:

$$x+aj \text{ oder } x+ai$$

eingetragen werden, wobei x und a zwei beliebige Zahlen sein können.

Kurz einige Hinweise: Die imaginäre Einheit j (oft wird auch i geschrieben, die Funktion akzeptiert beides) ist definiert als

$$j = \text{WURZEL}(-1)$$

Auf diese Weise lassen sich aus negativen Zahlen Wurzeln ziehen: WURZEL(-25) ist 5j. Mit imaginären Zahlen läßt sich ganz normal rechnen, wenn dabei beachtet wird, dass

$$j^2 = -1; j^3 = -j; j^4 = 1; j^7 = -j \text{ usw.}$$

$$4j + 2j = 6j$$

$$4j - 4j = 0$$

$$3j * 4j = -12$$

$$10j/2j = 5$$

Komplexe Zahlen sind Zahlen, die aus einem reellen und einem imaginären Anteil zusammengesetzt sind. Sie lassen sich geometrisch darstellen als Punkte in einem rechtwinkligen Koordinatensystem, bei dem die waagerechte Achse die reellen (also ganz normalen) Zahlen und die senkrechte Koordinate den imaginären Anteil repräsentiert.

Als Betrag einer komplexen Zahl gilt dann:

$$\text{Betrag}(x + aj) = \text{WURZEL}(x^2 + a^2) \text{ (Pythagoras)}$$

Der Winkel phi heißt Argument der komplexen Zahl.

## **IMAGINÄRTEIL()**

**Syntax:** IMAGINÄRTEIL(Komplexe\_Zahl)

Liefert den imaginären Anteil einer komplexen Zahl, vgl. IMABS().

## **IMAPOTENZ()**

**Syntax:** IMAPOTENZ(Komplexe\_Zahl;Potenz)

Liefert die ganzzahlige Potenz einer komplexen Zahl, vgl. IMABS().

Für das Quadrat komplexer Zahlen läßt sich die Rechnung einfach nachvollziehen:

$$(3+2j)^2 = 9 + 12j + 4j^2 = 9 + 12j - 4$$

## **IMARGUMENT()**

**Syntax:** IMARGUMENT(Komplexe\_Zahl)

Liefert das Argument einer komplexen Zahl, vgl. IMABS(). Der Winkel wird im Bogenmaß geliefert.

## **IMCOS()**

**Syntax:** IMCOS(Komplexe\_Zahl)

Liefert den Cosinus einer komplexen Zahl, vgl. IMABS().

## **IMDIV()**

**Syntax:** IMDIV(Komplexe\_Zahl1;Komplexe\_Zahl2)

Liefert das Ergebnis der Division zweier komplexer Zahlen, vgl. IMABS().

## **IMEXP()**

**Syntax:** IMEXP(Komplexe\_Zahl)

Liefert das Resultat der Potenzierung von e (Eulersche Zahl) mit einer komplexen Zahl, vgl. IMABS() und die mathematische Funktion EXP().

## **IMKONJUGIERTE()**

**Syntax:** IMKONJUGIERTE(Komplexe\_Zahl)

Liefert zu einer komplexen Zahl das konjugiert komplexe Komplement. Beide Zahlen zusammen heißen „konjugiert komplex“. Geometrisch handelt es sich um eine Spiegelung an der reellen Achse, vgl. IMABS().

## **IMLN()**

**Syntax:** IMLN(Komplexe\_Zahl)

Liefert den natürlichen Logarithmus einer komplexen Zahl, vgl. IMABS() und die mathematische Funktion LN().

## **IMLOG10()**

**Syntax:** IMLOG10(Komplexe\_Zahl)

Liefert den dekadischen Logarithmus einer komplexen Zahl, vgl. IMABS() und die mathematische Funktion LOG10().

## **IMLOG2()**

**Syntax:** IMLOG2(Komplexe\_Zahl)

Liefert den binären Logarithmus einer komplexen Zahl, vgl. IMABS() und die mathematische Funktion LOG().

## **IMPRODUKT()**

**Syntax:** IMPRODUKT(Komplexe\_Zahl1;Komplexe\_Zahl2,...)

Liefert das Produkt von bis zu 29 komplexen Zahlen, vgl. IMABS().

## **IMREALTEIL()**

**Syntax:** IMREALTEIL(Komplexe\_Zahl)

Liefert den reellen Anteil einer komplexen Zahl, vgl. IMABS().

## **IMSIN()**

**Syntax:** IMSIN(Komplexe\_Zahl)

Liefert den Sinus einer komplexen Zahl, vgl. IMABS().

## **IMSUB()**

**Syntax:** IMSUB(Komplexe\_Zahl1;Komplexe\_Zahl2)

Liefert das Resultat der Subtraktion zweier komplexer Zahlen, vgl. IMABS().

## **IMSUMME()**

**Syntax:** IMSUMME(Komplexe\_Zahl1;Komplexe\_Zahl2;...)

Liefert die Summe von bis zu 29 komplexen Zahlen, vgl. IMABS().

## **IMWURZEL()**

**Syntax:** IMWURZEL(Komplexe\_Zahl)

Liefert die Quadratwurzel einer komplexen Zahl, vgl. IMABS().

## **KOMPLEXE()**

**Syntax:** KOMPLEXE(Realteil;Imaginärteil;Suffix)

Bildet eine komplexe Zahl aus zwei reellen Zahlen, wobei die erste (Realteil) zum reellen Anteil, die zweite (Imaginärteil) zum imaginären Anteil wird. Mit Suffix läßt sich festlegen, ob i („i“ oder keine Angabe) oder j („j“) zur Kennzeichnung des Imaginärteils verwendet wird, vgl. IMABS().

## **OKTINBIN()**

**Syntax:** OKTINBIN(Zahl;Stellen)

Liefert den Binärwert einer oktalen Zahl, vgl. BININDEZ().

Stellen gibt an, wieviel Stellen angezeigt werden. Ohne Angabe werden nur die notwendigen Stellen angezeigt, ist Stellen größer als diese Anzahl, dann werden führende Nullen ausgegeben.

## **OKTINDEZ()**

**Syntax:** OKTINDEZ(Zahl)

Liefert den Dezimalwert einer oktalen Zahl, vgl. BININDEZ().

## **OKTINHEX()**

**Syntax:** OKTINHEX(Zahl;Stellen)

Liefert den hexadezimalen Wert einer oktalen Zahl, vgl. BININDEZ().

Stellen gibt an, wieviel Stellen angezeigt werden. Ohne Angabe werden nur die notwendigen Stellen angezeigt, ist Stellen größer als diese Anzahl, dann werden führende Nullen ausgegeben.

## **UMWANDELN()**

**Syntax:** UMWANDELN(Zahl;Von\_Maßeinheit;In\_Maßeinheit)

Liefert Umrechnungen zwischen verschiedenen Maßeinheiten. Mit Zahl wird angegeben, wie viele Einheiten der Von\_Maßeinheit umgerechnet werden sollen. Von\_Maßeinheit ist die Einheit, aus der umgerechnet wird; In\_Maßeinheit ist die Einheit, in die umgerechnet werden soll. Welche Einheiten zur Umrechnung verfügbar sind, zeigt die folgende Tabelle. Die Einheiten müssen in Anführungszeichen gesetzt werden.