

## Weitere Tabellenkalkulation/NX Funktionen

- ▶ NX biete folgende weitere Funktionen:
- ▶ Vektor bezogene Funktionen
  - ▶ Vektorfunktionen verwenden Tabellenzellenbereiche als Vektorkomponenten und geben entweder Skalarwerte in der aktuellen Zelle oder Vektorwerte als horizontale Reihe von Zellen aus.
- ▶ Punkt bezogene Funktionen
  - ▶ Punktbezogene Funktionen bearbeiten Punktkoordinaten und Vektoren.
- ▶ Objekt bezogene Funktionen
  - ▶ Objektbezogene Funktionen werden im Allgemeinen für spezifische Unigraphics NX Objekte verwendet. Bevor eine dieser Funktionen für ein Unigraphics NX Objekt verwendet werden kann, muss dem Objekt ein Unigraphics NX Namensattribut zugewiesen werden.
- ▶ Matrix Funktionen
  - ▶ Matrix-Funktionen arbeiten mit 3x3 Matrizen, benötigen als Eingabe einen bestimmten Bereich von Zellen und geben eine Gruppe von 3x3 Zellen als Werte einer Matrix aus.
- ▶ Ausdrucksbezogene Funktionen
  - ▶ Ausdrucksbezogene Funktionen liefern Informationen über Unigraphics NX Ausdrücke. Auf der linken Seite des Gleichheitszeichen steht der Parametername für den Ausdruck. Rechts vom Gleichheitszeichen steht die Zeichenfolge des Ausdrucks.

## Punktbezogene Funktionen

- ▶ Punktekoordinaten- und Vektorenargumente, beispielsweise pkt oder vektor, werden durch einen Bereich von 3 Zellen festgelegt (z. B. A1..A3).
- ▶ DELTAPT(distanz, pkt, vektor)
  - ▶ Verschiebt einen Punkt um eine bestimmte Distanz in Richtung eines Vektors und liefert als Ergebnis den neuen Punkt.  $\text{neu\_pkt} = \text{pkt} + (\text{distanz} * \text{vektor})$
- ▶ DIRPT(pkt1, pkt2)
  - ▶ Gibt die Einheitsvektorrichtung von Punkt 1 zu Punkt 2 an.
- ▶ DISTPERP(pkt, linie\_pkt, linie\_vek)
  - ▶ Liefert den senkrechten (kürzesten) Abstand zwischen dem Punkt "pkt" und der durch den Punkt linie\_pkt und den Vektor linie\_vek definierten Linie.
- ▶ DISTPROJ(pkt1, pkt2, vektor1)
  - ▶ Liefert den Abstand zwischen zwei Punkten in der durch den Normalen-Vektor1 definierten Ebene.
- ▶ DISTPT(pkt1, pkt2)
  - ▶ Liefert den Abstand zweier Punkte voneinander.
- ▶ INTERVEC(pkt1, vek1, pkt2, vek2)
  - ▶ Liefert den Schnittpunkt zweier Linien, der durch einen Punkt und einen Vektor definiert wird. Ergibt sich kein Schnittpunkt, wird eine Meldung in der Zelle angezeigt.

## Punktbezogene Funktionen

- ▶ ROTPT(winkel, pkt, vektor)
  - ▶ Dreht einen Punkt in einem bestimmten Winkel um einen Vektor und liefert als Ergebnis den neuen Punkt (prinzipiell die Kombination der Funktionen MTXROT und XFORMPT).
- ▶ XFORMPT(pkt, matrix)
  - ▶ Transformiert einen Punkt durch eine Matrix und liefert als Ergebnis den neuen Punkt.

## Vektor Bezogene Funktionen

- ▶ Ein Vektor, beispielsweise vektor1, wird durch einen Bereich von 3 Zellen (z. B. B1...B3). entweder in horizontaler oder vertikaler Ausrichtung angegeben. Ein Skalarwert kann als ein literaler Wert (0.5) oder als Zellreferenz zu einer Formel oder einem Wert angegeben werden.
- ▶ **ADDVEC(vektor1, vektor2)**
  - ▶ Bildet die Summe zweier Vektoren.  $\text{vektor} = \text{vektor1} + \text{vektor2}$
- ▶ **AFFINEVEC(skalar, vektor1, vektor2)**
  - ▶ Bildet die affine Vektorkombination.  $\text{vektor} = \text{vektor1} + (\text{skalar} * \text{vektor2})$
- ▶ **ANGLEVEC(vektor1, vektor2, vektor3)**
  - ▶ Berechnet den Winkel zwischen vektor1 und vektor2 in der durch den Normalen-Vektor3 definierten Ebene. Der Winkelbereich liegt zwischen Null und  $2 * \text{PI}$ .
- ▶ **CONVEXVEC(skalar, vektor1, vektor2)**
  - ▶ Bildet die konvexe Vektorkombination.  $\text{vektor} = (\text{skalar} * \text{vektor2}) + ((1 - \text{skalar}) * \text{vektor1})$
- ▶ **CROSSVEC(vektor1, vektor2)**
  - ▶ Bildet das Kreuzprodukt zweier Vektoren.

## Vektor Bezogene Funktionen

- ▶ DOTVEC(vektor1, vektor2)
  - ▶ Bildet das Skalarprodukt zweier Vektoren.
- ▶ LINEARVEC(skalar1, skalar2, vektor1, vektor2)
  - ▶ Bildet die lineare Vektorkombination.  $\text{vektor} = (\text{skalar1} * \text{vektor1}) + (\text{skalar2} * \text{vektor2})$
- ▶ NEGVEC(vektor1)
  - ▶ Bildet den Umkehrvektor (Vektornegation).
- ▶ SCALEVEC(skalar, vektor)
  - ▶ Multipliziert einen Vektor mit einem Skalarwert.
- ▶ SUBVEC(vektor1, vektor2)
  - ▶ Bildet die Differenz zweier Vektoren.
- ▶ UNITVEC(vektor1)
  - ▶ Ergibt einen Einheitsvektor.
- ▶ VECMAG(vektor1)
  - ▶ Bildet die Länge (Betrag) eines Vektors.

## Matrix Funktionen

- ▶ Die Vektorargumente für Matrix-Funktionen, wie beispielsweise `norm_vek`, werden durch einen zweidimensionalen Zellenbereich festgelegt (z. B. `A1..C3`).
- ▶ `MTXROT(winkel, norm_vek)`
  - ▶ Liefert die Matrix der Drehung in einem bestimmten Winkel um einen Vektor.

## Ausdrucksbezogene Funktionen

- ▶ Parameterbezeichnungen wie `parameter_name` sind Zeichenfolgenargumente und können durch eine in doppelte Anführungszeichen eingeschlossene Zeichenfolge, wie z. B. "Höhe", oder durch einen Zellverweis auf diese Zeichenfolge festgelegt werden.
- ▶ `UGEXPR("parameter_name")`
  - ▶ Liefert für den Parameternamen eines Ausdrucks in Unigraphics NX die *Zeichenfolge* des Ausdrucks. Dabei kann es sich um einen teileübergreifenden Ausdruck handeln, sofern das Teil geladen wurde. Wurde das Teil nicht geladen, wird in dieser Zelle eine Fehlermeldung ausgegeben.
- ▶ `EXPRVAL("parameter_name")`
  - ▶ Liefert für den Parameternamen eines Ausdrucks in Unigraphics NX den *Wert* des Ausdrucks. Diese Funktion lässt sich auch auf teileübergreifende Ausdrücke anwenden, die im betreffenden Teil verwendet werden.

## Objektbezogene Funktionen

- ▶ Objektbezogenen Funktionen können numerische Argumente oder Zeichenfolgen zugewiesen werden. Zeichenfolgenargumente wie "object\_name" können durch eine in doppelte Anführungszeichen eingeschlossene Zeichenfolge wie zum Beispiel "LINE1" oder durch eine Zellreferenz auf die Zeichenfolge angegeben werden.
- ▶ UGATTR("object\_name", "attribut\_title", attribute\_type)
  - ▶ Liefert einen Unigraphics NX Attributwert, wenn ein Unigraphics NX Objektname, ein anwenderdefinierter Attributtitel und ein numerischer Attributtyp angegeben wurde. Bei dem numerischen Attributtyp kann es sich um folgende Typen handeln:
    - ▶ 1 = Ganzzahlattribut
    - ▶ 2 = Reelles Attribut
    - ▶ 3 = Zeichenfolgeattribut
    - ▶ 4 = Referenzattribut
  - ▶ Entspricht der Objektname der Zeichenfolge "PART", werden der angegebene Attributtitel und der Attributtyp für das aktive Teil ausgegeben.

## Objektbezogene Funktionen

- ▶ UGNAMES()
  - ▶ Liefert den Objektnamen, den Objekt-Typ und den Untertyp sowie den Layer aller benannten Geometrieobjekte des aktiven Teils. Die Informationen werden für jedes Objekt in jeweils einer Zeile ausgegeben.
- ▶ ALLATTR("object\_name")
  - ▶ Liefert sämtliche Titel und Typen von Ganzzahl-, Fließkomma- und Zeichenfolgeattributen, die mit einem bestimmten Objekt verknüpft sind. Entspricht der Objektname der Zeichenfolge "PART", werden alle Titel und Typen von Ganzzahl-, Fließkomma- und Zeichenfolgeattributen ausgegeben.
- ▶ ARCLLEN("object\_name", param1, param2, units)
  - ▶ Liefert die Bogenlänge einer Kurve zwischen den beiden Parameterwerten param1 und param2. Diese Funktion kann für alle Kurventypen wie Bogen, Kreise, Kegelschnitte und Linien verwendet werden.
  - ▶ Die Parameterwerte liegen in einem Bereich von 0.0 bis 1.0. Die Maßeinheitenkonvertierung wird durch eine Zahl zwischen 1 und 4 angegeben. Dabei gilt:
    - ▶ 1 = Zoll
    - ▶ 2 = Millimeter
    - ▶ 3 = Zentimeter
    - ▶ 4 = Meter

## Objektbezogene Funktionen

- ▶ `EVALCRV("object_name", eval_type, param1, param2, steps)`
  - ▶ Liefert Auswertungsdaten für eine bestimmte Anzahl Schritte entlang einer Kurve in einem Bereich zwischen den Parameterwerten `param1` und `param2`.
  - ▶ Die Parameterwerte liegen in einem Bereich zwischen 0.0 und 1.0. Für die Anzahl der Schritte kann ein Wert zwischen 1 und 100 festgelegt werden. Es folgt eine Liste der Typen für die Kurvenauswertung:

Typ	Anzahl Werte	Beschreibung
1	1	Gaußsche Krümmung
2	1	Mittlere Krümmung
3	1	Minimaler Radius
4	1	Maximaler Radius

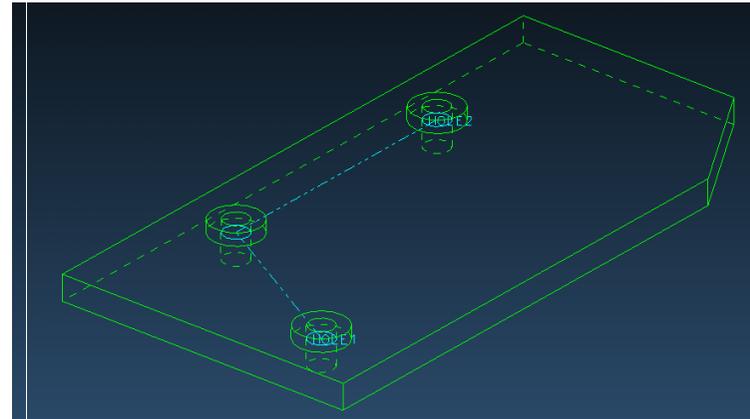
- ▶ Werden im Bereich zwischen den Parameterwerten mehrere Schritte verwendet, ist das Ergebnis der Form nach mit den Funktionen für die Graphenerstellung kompatibel. sich lediglich einer der Parameter `u` oder `v`, entspricht die Ausgabe der Form nach dem über die Funktion `=EVALCRV` erzielten Ergebnis. sich hingegen beide Parameter, ist das erzielte Ergebnis mit einem Oberflächengraphen kompatibel.

## Objektbezogene Funktionen

- ▶ POINT("object\_name")
  - ▶ Gibt die das Objekt "definierende" xyz-Position in der aktuellen Zelle und den daneben angeordneten zwei Spalten aus. Bei Kreisen, Bogen, sowie elliptischen Kegelschnitten wird der Mittelpunkt des Kreises als Ergebnis ausgegeben. Bei Ebenen und Koordinatensystemen wird als Ergebnis der Ursprung des Objekts ausgegeben.
  - ▶ Die Funktion POINT wird wie folgt verwendet:
    - ▶ Die Kalkulationstabelle aufrufen und die drei benachbarten Reihen auswählen.
    - ▶ In Excel Folgendes eingeben:  
=POINT("<Name>")  
In Xess Folgendes eingeben:  
=@POINT("<Name>")
    - ▶ Zur Auswertung der Koordinaten und für die Übernahme der Zellen Strg+Umschalt+Eingabetaste drücken.

## Beispiel für Objekt (Punkt) und Punkt-Funktion

- ▶ Öffnen Sie das Teil punkt\_vektor.prt



- ▶ Wechseln Sie in die Tabellenkalkulation
- ▶ Fragen Sie in der Tabellenkalkulation die Punkte für HOLE1 und HOLE2 ab
  - ▶ Beachten Sie die vorher beschriebene Vorgehensweise
  - ▶ 1.Drei Zelle markieren
  - ▶ 2.Eingeben der Funktion =POINT("HOLE1")
  - ▶ 3.Drücken von Strg+Umschalt+Enter

J35		fx										
	A	B	C	D	E	F	G	H	I	J	K	
1		<b>Ausruecke</b>			<b>Bohrungspunkte</b>					<b>Abstand</b>		
2		<b>Parameter</b>	<b>Werte</b>		<b>Bohrung</b>	<b>XC</b>	<b>YC</b>	<b>ZC</b>		<b>Bohrung</b>	<b>Abstand</b>	
3		<b>h1_lgth</b>	4		<b>Hole1</b>	4,714101615	1	0,5				
4		<b>h2_lgth</b>	5,230943		<b>Hole2</b>	1,25	8,230943	0,5				
5		<b>hole_dia</b>	0,5									
6		<b>angle</b>	120									
7		<b>blk_lgth</b>	12									
8		<b>blk_thk</b>	0,5									
9		<b>blk_wdt</b>	6									

## Beispiel für Objekt (Punkt) und Punkt-Funktion

- ▶ Fragen Sie nun über die DISTPT Funktion den Abstand zwischen den Bohrungen ab
  - ▶ Eingeben von =DISTPT(F3:H3;F4:H4)

A	B	C	D	E	F	G	H	I	J	K
	<b>Ausruecke</b>			<b>Bohrungspunkte</b>					<b>Abstand</b>	
	<b>Parameter</b>	<b>Werte</b>		<b>Bohrung</b>	<b>XC</b>	<b>YC</b>	<b>ZC</b>		<b>Bohrung</b>	<b>Abstand</b>
	h1_lgth	4		Hole1	4,714101615	1	0,5		Hole1-->Hole2	8,017888542
	h2_lgth	5,230943		Hole2	1,25	8,230943	0,5			
	hole_dia	0,5								
	angle	120								

- ▶ Optimieren Sie nun das Modell so dass der Abstand zwischen den Bohrungen genau 8 in beträgt. Der Variable Wert soll „angle“ sein

	<b>Ausruecke</b>			<b>Bohrungspunkte</b>				<b>Abstand</b>	
	<b>Parameter</b>	<b>Werte</b>		<b>Bohrung</b>	<b>XC</b>	<b>YC</b>	<b>ZC</b>	<b>Bohrung</b>	<b>Abstand</b>
	h1_lgth	4		Hole1	4,729770727	1,027388612	0,5	Hole1-->Hole2	8,000000008
	h2_lgth	5,230943		Hole2	1,25	8,230943	0,5		
	hole_dia	0,5							
	angle	119,5480206							
	blk_lgth	12							
	blk_thk	0,5							
	blk_wdt	6							