

# Sinus Reihenentwicklung

Schreiben Sie eine MATLAB-Funktion `reihe_sin.m`.

1. Bevor Sie zum programmieren schreiten, lesen Sie sich unbedingt die [Einführung](#) in die Berechnung von Reihen.
2. Die Funktion soll mit `[r,u] = reihe_sin(x,n)` aufgerufen werden können.
3. Übergabeparameter: `x` ein Array von  $x$ -Werten und `n` der Maximalwert für die  $k$ -Werte.
4. Rückgabewerte: die Teilsummen `r`

$$r(x) = x - \frac{x^3}{3!} + \frac{x^5}{5!} - \frac{x^7}{7!} + \dots \quad (1)$$

$$= \sum_{k=0}^n (-1)^k \frac{x^{2k+1}}{(2k+1)!}, \quad (2)$$

die analytischen Werte der Funktion `u`

$$u = \sin(x). \quad (3)$$

5. Alle Outputwerte müssen die gleiche Größe wie `x` haben. Dazu merkt man sich am Anfang der Funktion die Größe von `x` (`size`), macht aus dem Array `x` einen Vektor (`colon`). Am Ende des Programms kann man dann mit `reshape` sicherstellen, dass die richtige Größe zurückgegeben wird. Beachten Sie also, dass `x` ein beliebig dimensionierte Matrix sein kann.
6. Verwenden Sie für die Berechnung der Reihe den `meshgrid` Befehl. `for`-Schleifen werden nicht benötigt.

## Hinweis:

Die Berechnung von  $k!$  (Fakultät) kann mit Hilfe der  $\Gamma$ -Funktion durchgeführt werden

$$\Gamma(n+1) = n! \quad (n = 0, 1, 2, \dots), \quad (4)$$

wobei dafür in MATLAB die Funktion `gamma(k)` zur Verfügung steht. Diese funktioniert natürlich wieder für Arrays von  $k$ -Werten. Für große  $k > 171$  liefert  $\Gamma(k)$  den Wert unendlich, da das Ergebnis jenseits der Darstellungsgrenze für den Datentyp `double` liegt.

## Hinweis:

In der Matlab-Console können sie `x` definieren, `reihe_sin` ausführen, und danach mit einem Plotbefehl `r` und `u` über `x` darstellen. So sehen Sie sofort ob die Rückgabeparameter Sinn machen.

Gesucht:

Funktion `reihe_sin.m`

```
[r,u] = reihe_sin(x,n)
    x      : Matrix mit x-Werten
    n      : Obergrenze der Aufsummierung
    r      : Rückgabevektor der Teilsumme der Taylorreihe des Sinus
    u      : analytische Werte des Sinus
```

---

## Anschauungsbeispiel:

```
>> [r,u] = reihe_sin([0,4;0.5,-0.5],3)
```

```
r =
```

```
    0    -1.3841  
0.4794  -0.4794
```

```
u =
```

```
    0    -0.7568  
0.4794  -0.4794
```