

Eulerschen Zahlen

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

1. Bevor Sie zum programmieren schreiten, lesen Sie sich unbedingt die [Einführung](#) in die Berechnung von Reihen.
2. Die Funktion soll mit `[e] = reihe_eulernum(k, lmax)` aufgerufen werden können.

Die Eulerschen Zahlen E_k sind durch folgende Reihe definiert

$$E_k = \frac{(-1)^k 2^{2k+2} (2k)!}{\pi^{2k+1}} \sum_{l=1}^{\infty} \frac{(-1)^{l-1}}{(2l-1)^{2k+1}} . \quad (1)$$

3. Hier wird nun über l summiert und der k -Vektor spielt die Rolle, die vorher x gespielt hat, d.h. für jeden Wert von k bekommt man einen Wert von E_k .
4. Als Defaultwert für `lmax` soll 10 verwenden werden.
5. Wird kein `k` übergeben, dann soll mit der Fehlermeldung `Not enough input arguments.` abgebrochen werden.
6. Da alle Eulerzahlen ganzzahlig sind, empfiehlt es sich, am Ende der Funktion den Befehl `round` zu verwenden. Damit werden leichte numerische Ungenauigkeiten ausgeglichen, und man muss `lmax` nicht größer wählen.
7. Alle Outputwerte müssen die gleiche Größe wie `k` haben. Dazu merkt man sich am Anfang der Funktion die Größe von `k` (`size`), macht aus dem Array `k` einen Vektor (`colon`). Am Ende des Programms kann man dann mit `reshape` sicherstellen, dass die richtige Größe zurückgegeben wird. Beachten Sie also, dass `k` ein beliebig dimensionierte Matrix sein kann.

Weitere Informationen zu den Eulerschen Zahlen gibt es auf der [Wikipedia](#).

Hinweis:

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

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

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.

Gesucht:

Funktion `eulernum.m`

```
[e] = reihe_eulernum(k, lmax)
    k       : Vektor der Indizes der Eulerzahlen
    lmax    : Obergrenze der Aufsummierung (default: 10)
    e       : Rückgabevektor mit Eulerzahlen
```

Anschaungsbeispiel:

```
>> reihe_eulernum([1,2;3,4])
ans =
     -1         5
    -61       1385
```