Programmieraufgabe VI (21 Punkte)

Abgabe in der Woche 1.-5. Juli

In [44]:
```
import numpy as np
import matplotlib.pyplot as plt
```

Teilaufgabe i: 10 Punkte

Implementieren Sie das allgemeine m-stufige Runge-Kutta-Verfahren mit den Parametern $A \in \mathbb{R}^{m\times m}$, $b, c \in \mathbb{R}^m$ (vgl. Butcher-Tableau in Aufgabe 1 auf Blatt 10) zur Lösung des Anfangswertproblems

$$\dot{y}(t) = f(t, y(t)) \quad \forall t \in [t_0, t_1], \qquad y(t_0) = y_0.$$

Das nichtlineare Gleichungsystem im Runge-Kutta-Verfahren soll mittels Fixpunktiteration gelöst werden. Als Argumente übergeben werden sollen:

- A,b,c
- die Funktion f und der Startwert y_0
- die Anzahl der Zeitschritte $nsteps$, Anfangs- und Endzeit t_0 bzw. t_1
- maximale Anzahl an Schritten der Fixpunktiteration $maxit$, Abbruchtoleranz der Fixpunktiteration tol

In [ ]:
```

```

Teilaufgabe ii: 6 Punkte

Im Folgenden sind die Parameter für vier Testverfahren gegeben:

- Implizites Eulerverfahren ("ie")
- Implizite Mittelpunktsregel ("mp")
- Radau3-Verfahren ("rad3")
- Klassisches Runge-Kutta Verfahren ("rk")

In [45]:
```
A_ie = np.array([[1.]])
b_ie = np.array([1.])
c_ie = np.array([1.])

A_mp = np.array([[0.5]])
b_mp = np.array([1.])
c_mp = np.array([0.5])

A_rad3 = np.array([[5/12, -1/12],[3/4, 1/4]])
b_rad3 = np.array([3/4,1/4])
c_rad3 = np.array([1/3, 1])

A_rk = np.array([[0,0,0,0],[0.5,0,0,0],[0,0.5,0,0],[0,0,1,0]])
b_rk = np.array([1/6,1/3,1/3,1/6])
c_rk = np.array([0,0.5,0.5,1])

Verfahren = [[A_ie, b_ie, c_ie, 'Impl.Euler'],[A_mp,b_mp,c_mp,'Impl. Mittelp
unktsr.'], [A_rad3, b_rad3, c_rad3,'Radau3'], [A_rk, b_rk, c_rk,'Klass. Rung
e-Kutta']]
```

Testen Sie Ihre Implementierung für die vier oben gegebenen Verfahren an den Problemen

$$\dot{y} = \begin{pmatrix} 0.1 & 0 \\ 0 & -0.1 \end{pmatrix} y, \qquad y(0) = \begin{pmatrix} 1 \\ 1 \end{pmatrix},$$

bzw.

$$\dot{y} = \begin{pmatrix} 0 & -\pi \\ \pi & 0 \end{pmatrix} y, \qquad y(0) = \begin{pmatrix} 1 \\ 0 \end{pmatrix},$$

jeweils mit $t_0 = 0, t_1 = 10$.

- Plotten Sie jeweils die numerisch berechnete Lösung für verschiedene Schrittweiten und die exakte Lösung. Was können Sie beobachten?
- Plotten Sie, wie sich der maximale (auf $[t_0, t_1]$) Abstand zwischen der diskreten und der exakten Lösung in Abhängigkeit von der Schrittweite h verhält. Welche Konvergenzordnung können Sie jeweils beobachten?

In [ ]:

Teilaufgabe iii: 5 Punkte

In dieser Teilaufgabe wollen wir die oben besprochenen Verfahren auf ein (zumindest ansatzweise) reales Problem anwenden: Die Bewegung eines Objekts mit vernachlässigbarer Masse (z.B. Satellit) im Schwerefeld von Erde und Mond lässt sich (unter Verwendung eines mit Erde und Mond mitrotierenden Koordinatensystems) annähernd durch

$$\ddot{x}(t) = x(t) + 2\dot{y}(t) - \hat{\mu}\frac{x(t)+\mu}{N_1} - \mu\frac{x(t)-\hat{\mu}}{N_2},$$
$$\ddot{y}(t) = y(t) + 2\dot{x}(t) - \hat{\mu}\frac{y(t)}{N_1} - \mu\frac{y(t)}{N_2},$$

beschreiben, wobei $(x(t), y(t))$ die Orts-Koordinaten des Objekts zum Zeitpunkt t angeben und wir abkürzend

$$N_1 = ((x(t)+\mu)^2 + y(t)^2)^{3/2}, \qquad N_2 = ((x(t)-\hat{\mu})^2 + y(t)^2)^{3/2}$$

benutzt haben. Die Position der Erde ist in $(0, 0)$ fixiert, die des Mondes in $(1, 0)$. Mit μ wird das Verhältnis der Mondmasse zur Erdmasse beschrieben und es gilt $\hat{\mu} = 1 - \mu$.

Finden Sie für die vier oben angegebenen Verfahren jeweils passende Schrittweiten, um die Bewegung eines Objekts mit Anfangsbedingung

$$x(0) = 0.994, \quad \dot{x}(0) = 0, \quad y(0) = 0, \quad \dot{y}(0) = -2.001585106$$

für $t_0 = 0, t_1 = 17.0652166$ zu simulieren.

Hinweis: Als Lösung ergibt die in der png-Grafik "satellit.png" (siehe Webseite) dargestellte Kurve, die periodisch mit Periodendauer t_1 durchlaufen wird. Versuchen Sie Schrittweiten zu finden, sodass dieses Verhalten zumindest optisch erreicht wird...

Welche (absolute) Genauigkeit wäre notwendig, um in diesem Modell die Bewegung eines Satelliten mit 1km Genauigkeit zu simulieren?

In [ ]:

In [ ]: