Kepler : $U = \begin{bmatrix} x \\ y \\ \dot{x} \\ \dot{y} \end{bmatrix} = \begin{bmatrix} \vec{r} \\ \dot{\vec{r}} \end{bmatrix} \rightarrow$ Vector de estado

$$F(U,t) = \frac{dU}{dt} = \begin{bmatrix} \dot{\vec{r}} \\ \ddot{\vec{r}} \end{bmatrix}$$

Como la aceleración se debe a la fuerza gravitacional:

$$\ddot{\vec{r}} = \frac{-\vec{r}}{|\vec{r}|^3}$$

```python
from numba import njit
from numpy import array, sqrt


#@jit(nopython=True)
@njit
def Kepler(U, t):

    x = U[0]; y = U[1]; dxdt = U[2]; dydt = U[3]
    d = ( x**2  +y**2 )**1.5

    return  array( [ dxdt, dydt, -x/d, -y/d ] )

@njit
def Arenstorf_equations(U, t):

     mu = 0.012277471

     x = U[0]; y = U[1]; vx = U[2];  vy = U[3]

     D1 = sqrt( (x+mu)**2      + y**2 )**3
     D2 = sqrt( (x-(1-mu))**2 + y**2 )**3

     dxdt  = vx
     dydt  = vy
     dvxdt = x + 2 * vy - (1-mu)*( x + mu )/D1 - mu*(x-(1-mu))/D2
     dvydt = y - 2 * vx - (1-mu) * y/D1 - mu * y/D2

     return array( [ dxdt, dydt, dvxdt, dvydt ] )
```

Modela un problema de 3 cuerpos

mu = 0.012277471 ⟶ $\mu \equiv$ Parámetro que representa la relación entre las masas de los 2 cuerpos principales

Las ecs del movimiento se derivan de las fuerzas gravitacionales de los 2 cuerpos principales. Se meten las distancias cúbicas D_1 y D_2

$$D_1 = \left|(x+\mu)^2 + y^2\right|^{3/2} \qquad D_2 = \left|[x-(1-\mu)]^2 - y^2\right|^{3/2}$$

Código

Este código define dos funciones que describen **sistemas dinámicos** mediante ecuaciones diferenciales:

1. **Kepler(U, t)**:
- Describe el movimiento orbital de un objeto bajo la influencia de la **gravedad** (problema de Kepler).
- Toma un estado que contiene posición y velocidad, y devuelve cómo cambian estas en función del tiempo.

2. **Arenstorf_equations(U, t)**:
- Define un sistema más complejo de movimiento, el **problema de órbitas de Arenstorf**, que involucra la interacción entre tres cuerpos (por ejemplo, la Tierra, la Luna y un satélite).
- Usa la masa reducida y devuelve las derivadas de posición y velocidad.

Ambas funciones se pueden usar en esquemas numéricos para integrar en el tiempo y simular estos sistemas.

Resumen código:

El código define dos sistemas de ecuaciones diferenciales ordinarias (EDO): uno para el **problema de Kepler** (movimiento orbital bajo gravedad) y otro para las **ecuaciones de Arenstorf** (un sistema de tres cuerpos). Ambas funciones calculan la tasa de cambio de las posiciones y velocidades del sistema, devolviendo las derivadas del estado en cada instante de tiempo. Estas funciones pueden ser usadas en métodos de integración numérica para simular la evolución de sistemas dinámicos.