F04

Tidigare $\begin{cases} \dot{u}(t) = f(t), & t \in (0,T] \\ u(0) = u^0 \end{cases}$ (2)

Euler framåt: $u((n+1)k) = u(nk) + f(nk) \cdot k$

Felet $\leq \frac{LT}{2} k$, $u(t) = u^0 + \int_0^t f(s)\,ds$

Ordinär differentialekvation (ODE)

$$\begin{cases} \dot{u}(t) = f(u(t)), & t > 0 \\ u(0) = u^0 \end{cases}$$

Framåt Euler (Explicit Euler): $u((n+1)k) = u(nk) + f(u(nk)) \cdot k$

Bakåt Euler (Implicit Euler): $u((n+1)k) = u(nk) + f(u((n+1)k)) \cdot k$

Trapetsmetoden: $u((n+1)k) = u(nk) + \frac{k}{2}\left(f(u((n+1)k)) + f(u(nk))\right)$

Python: "ref.py"

1/ $\begin{cases} \dot{u}(t) = u \\ u(0) = 1 \end{cases}$

2/ $\begin{cases} \dot{u}(t) = -u \\ u(0) = 1 \end{cases}$

3/ $\begin{cases} \dot{u}(t) = 1/u \\ u(0) = 0 \end{cases}$

Viba implementera av implicita metoder!

Ex: Implicit Euler

```
u0 = u
for i in range(0,5):
    fval = f(t,u)
    u = u0 + dt*fval
```