Tiefpassfilter erster Ordnung ("Exponentialfilter")

$T\_max := 5\cdot s$ Eingestellte Simulationszeit (in Sekunden) $dt := 0.05\cdot s$ Zeitschritt der Simulation

$n\_max := \mathrm{rund}\left(\frac{T\_max}{dt}\right)$ Berechung der Anzahl der Simulationsschritte $n\_max = 100$

$n := 0 .. (n\_max - 1)$ Erzeugung eines Laufzeit-Indizes "n" und des Zeitvektors "t" $t_n := n\cdot dt$

$In_n := 1$ Erzeugung und Initialisierung eines Vektors "In" mit n_max Elementen als Einheitssprung

$Tau := 1\cdot s$ Festlegung der Zeitkonstante für die Filterfunktion

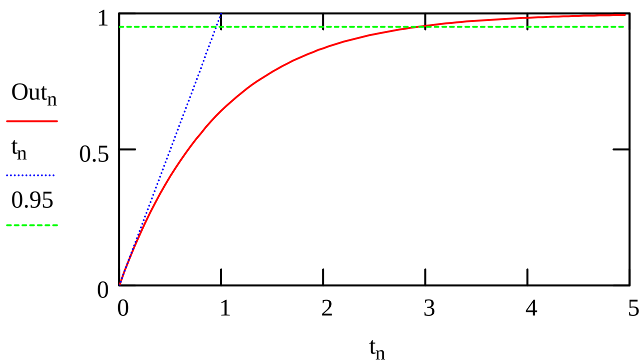
$dt\_div\_Tau := \frac{dt}{Tau}$ Berechnung einer Hilfsgröße für den Algorithmus $dt\_div\_Tau = 0.05$

Beschreibung des Algorithmus

```
ExpFi(In) := | Erg_0 ← 0
             | z ← 1
             | while z < n_max
             |   | Erg_z ← Erg_(z-1) + dt_div_Tau·(In_z − Erg_(z-1))
             |   | z ← z + 1
             | Erg
```

Berechnung des Ergebnisvektors: $Out := ExpFi(In)$

Darstellung des Ergebnisses:



Bemerkungen:

Es ist gut erkennbar, wie die Nullpunkt-Tangente an den Anstieg den Endwert beim Zeitwert Tau (1s) schneidet.
Ebenso gut ist erkennbar, dass bei 3 Tau 95% vom Endwert erreicht sind.
Nach allem, was wir in der Schule gelernt haben:
die Reaktion auf den Einheitssprung ist klar die Exponentialfunktion 1-exp(- t/Tau).

Nun kann dieser Algorithmis zur Verarbeitung verrauschter Eingangssignalverläufe verwendet werden.

$In\_cos_n := \mathrm{rnd}(\sin(n\cdot 0.025))$ $Out\_cos := ExpFi(In\_cos)$