Fließkommazahlen:

Dieser Abschnitt behandelt ausschließlich IEEE 754 32-Bit Fließkommazahlen:

Addition ist Kommutativ (a+b=b+a). ☺

Multiplikation ist Kommutativ. ☺

Addition ist Assoziativ [(a+b)+c=a+(b+c)]. ☹

Multiplikation ist Assoziativ. ☹

Die Auswertung der Addition ist stabil. ☹

Die Auswertung der Mult. ist stabil. ☺

Polynominterpolation:

Die Summe aller Bernstein Polynome ist 1. ☺

Die Summe aller Hermite Polynome ist 1. ☹

Die Summe aller Legendre Polynome ist 1. ☹

Die Summe aller Lagrange Polynome ist 1. ☺

Lösen Sie das gegebene Aitken Neville Schema für den Punkt $x = 2$:

x_i	$i \setminus k$	0	1	2	3
0	0	1	7	5	5
1	1	4	4	5	
3	2	4	7		
4	3	1			

Lösen Sie das gegebene Newton Schema:

x_i	$i \setminus k$	0	1	2	3
0	0	1	3	1	0
1	1	4	0	-1	
3	2	4	-3		
4	3	1			

Aitken Neville kann mit 5 Stützpunkten Polynome vom Grad 5 exakt interpolieren. ☹

Die Interpolation in der Nähe der Intervallgrenzen wird besser wenn man doppelt so viele äquidistante Stützpunkte verwendet. ☹

Die Spline-Interpolation wird instabil für viele Stützstellen (>8). ☹

Die Stückweise Interpolation kann verbessert werden, wenn man statt Hermite Polynome Bezierkurven verwendet. ☹

Numerische Integration:

Für viele äquidistant verteilte Stützstellen werden die Gewichte der Integrationsverfahren zum Teil negativ (>9). ☺

Es gibt Integrationsformeln, die mit 11 Stützpunkten ein Polynom vom Grad 11 exakt integrieren. ☺

Die Position der Stützstellen ist relevant für den Genauigkeitsgrad. ☺

Die Romberg-Quadratur ist konvergent. ☺

Die Gauß-Quadratur ist konvergent. ☺

Die Stützstellen der Gauß-Quadratur liegen symmetrisch um den Intervallmittelpunkt. ☺

Die Gewichte der Gauß-Quadratur werden für viele Stützstellen zum Teil negativ. ☹

Die Gewichte einer Quadraturformel werden immer mit Hilfe der Lagrange Polynome berechnet. ☺

Die Trapezsumme ist quadratisch konvergent ☺

Die Simpson-Summe ist quadratisch konvergent ☺

Die Gauß Quadratur liefert stets ein besseres Ergebnis als die Trapezsumme ☹

Fourier Transformation:

Die Diskrete Fourier Transformation ist linear $\left(DFT(a+b) = DFT(a) + DFT(b)\right)$. ☺

Die Inverse Fourier Transformation ist linear $\left(IDFT(a+b) = IDFT(a) + IDFT(b)\right)$. ☺

Die Schnelle Fourier Transformation ist linear $\left(FFT(a+b) = FFT(a) + FFT(b)\right)$. ☺

Die Fourier Transformation ist gut geeignet, exponentiell wachsende Funktionen zu interpolieren. ☹

Die Fourier Transformation wird zur Kompression verwendet, weil das Frequenzspektrum weniger Speicher benötigt als die Funktion selbst. ☹

Anfangswertprobleme:

Einschritt Verfahren sind konvergent. ☹

Mehrschritt Verfahren sind konvergent. ☹

Zu einem beliebigen konvergenten Quadraturverfahren kann eine konvergente Formel zum Lösen eines Anfangswertproblem gefunden werden. ☺

Das Newton-Verfahren ist für eine stetig differenzierbare Funktion stets global quadratisch konvergent ☹

Bei einem steifen Problem konvergieren explizite Verfahren nicht. ☹

Benennen Sie das folgende Verfahren:

$x_{k+1} = x_k + \delta t \cdot f(t_{k+1}, y_{k+1})$

Implizites Eulerverfahren

Allgemeines - Lineare Gleichungssysteme:

$\kappa_A = \frac{\varrho(A)}{\varrho(A^{-1})}$ ☺

$\varrho(A^{-1}) \leq 1 \Rightarrow Ax = b$ gut Konditioniert ☺

$A = A^T \Rightarrow \lambda_i \in \mathbb{R}$ (wenn gilt: $A \in \mathbb{R}^{n \times n}$) ☺

$A = \bar{A} \Rightarrow \lambda_i \in \mathbb{R}$ ☹

$x^T A x > 0 \Rightarrow$ A invertierbar ☺

$\det(A) > 0 \Rightarrow x^T A x > 0$ ☹

A positiv definit $\Rightarrow \varrho(A) > 0$ ☺

$A = A^T \Leftrightarrow \langle Ax, y \rangle = \langle x, Ay \rangle$ ☺

Algorithmen - Lineare Gleichungssysteme:

Splitting Verfahren benötigen weniger Rechenschritte bis sie ein exaktes Ergebnis liefern als direkte Löser. ☹

Der Gauß Algorithmus mit Pivot Suche verbessert die Kondition des Problems. ☹

Das CG Verfahren liefert das exakte Ergebnis nach n Iterationsschritten (falls die Systemmatrix symmetrisch positiv definit ist). ☺

Benennen Sie folgende Verfahren und was für x_∞ errechnet wird:

$x_{k+1} = x_k + \omega r_k$

Jacobi Verfahren, $Ax_\infty = b$

$x_{k+1} = (A - \mu I)x_k$

Power Iteration mit Shift, x_∞ EV zum größten EW

$\alpha_k = \frac{r_k^T r_k}{r_k^T A r_k}$

$x_{k+1} = x_k + \alpha_k r_k$

Verfahren des Steilsten Abstiegs, $Ax_\infty = b$
