The questions in this exercise sheet correspond to the chapter on finite differences, which is not included in the *Numerische Mathematik 2* transcript. An online textbook (Lloyd N. Trefethen: *Finite Difference and Spectral Methods for Ordinary and Partial Differential Equations*) covering that (and additional) material can be found at `http://people.maths.ox.ac.uk/trefethen/pdetext.html`. The exercises below correspond to section 3.2 of that textbook.

The notation used is the same as in that textbook, in particular, $u_t = \frac{\partial u}{\partial t}$, $u_x = \frac{\partial u}{\partial x}$, $u_{xx} = \frac{\partial^2 u}{\partial x^2}$, and $v_j^n = v(x_j, t_n)$, the discretized version of $u(x_j, t_n)$.

76. (a) Implement the **Leap frog** formula for the **first-order wave equation** $u_t = u_x$:

$$v_j^{n+1} = v_j^{n-1} + \frac{k}{h}(v_{j+1}^n - v_{j-1}^n)$$

in a MATLAB or Octave program with the following inputs:

- a space step h,
- a time step k,
- a radius M such that the space interval considered is $[-M, M]$ (M should be a multiple of h),
- a target time T, which should be a multiple of k, and
- a list `v0` of length $2\frac{M}{h} + 1$ discretizing the initial conditions $u_0(x)$, i.e. `v0`$(j) = u_0(-M + jh)$, $0 \leq j \leq 2\frac{M}{h}$

and with as output a list `vT` of length $2\frac{M}{h} + 1$ discretizing the solution $u(x, T)$ at the time T.

Note: You can assume that $u(x, t) = 0$ for $x \notin [-M, M]$, i.e. ignore the term v_{j-1}^n at $x = -M$ and the term v_{j+1}^n at $x = M$.

In addition, at $t = 0$, use $v_j^{n-1} = v_j^n =$ `v0`(j).

(b) Run your program on the hat-shaped initial data $u_0(x) = max(0, 1 - |x|)$ and the parameters $h = 0.1$, $k = 0.04$, $M = 5$ and $T = 1$.

(c) Plot the result and compare it with the exact solution

$$u(x, 1) = max(0, 1 - |x + 1|).$$

77. (a) Implement the **Lax-Wendroff** formula for the **first-order wave equation** $u_t = u_x$:

$$v_j^{n+1} = v_j^n + \frac{k}{2h}(v_{j+1}^n - v_{j-1}^n) + \frac{k^2}{2h^2}(v_{j+1}^n - 2v_j^n + v_{j-1}^n)$$

in a MATLAB or Octave program with same inputs and output as in example 76.

Note: Assume again that $u(x, t) = 0$ for $x \notin [-M, M]$.

(b) Run your program on the hat-shaped initial data $u_0(x) = max(0, 1 - |x|)$ and the parameters $h = 0.1$, $k = 0.04$, $M = 5$ and $T = 1$.

(c) Plot the result and compare it with the exact solution

$$u(x, 1) = max(0, 1 - |x + 1|)$$

and with the result from example 76 (c).

78. (a) Implement the **Euler** formula for the **heat equation** $u_t = u_{xx}$:

$$v_j^{n+1} = v_j^n + \frac{k}{h^2}(v_{j+1}^n - 2v_j^n + v_{j-1}^n)$$

in a MATLAB or Octave program with same inputs and output as in examples 76 and 77.
Note: You can assume that $u(x,t) \approx 0$ for $x \notin [-M, M]$, i.e. ignore the term v_{j-1}^n at $x = -M$ and the term v_{j+1}^n at $x = M$ as if they were exactly zero.

(b) Run your program on the hat-shaped initial data $u_0(x) = max(0, 1 - |x|)$ and the parameters $h = 0.1$, $k = 0.004$, $M = 5$ and $T = 1$.
Note: We need a smaller time step k here than in the examples 76 and 77 because we have a $\frac{k}{h^2}$ term in the formula instead of $\frac{k}{h}$.

(c) Plot the result.