Task 1

Convert your mean and standard deviation program from last week into a function or subroutine. The function/subroutine should read in a 1D array as an argument and return the mean and standard deviation.

Write a main program that reads in the values, then calls the function/subroutine and finally displays the result.

Note: A function can return only one value directly, so either the mean or the standard deviation would have to be an argument.

Task 2

Convert your finite differences program from last week into a subroutine. The subroutine should take a 1D array and the grid spacing as input arguments and return the second derivative of the array as output argument. Make sure that the result is still the same as in the last exercise. Save the subroutine in a module, so you can use it in Task 3.

One-dimensional heat diffusion equation

$$\frac{\partial T}{\partial t} = \kappa \cdot \frac{\partial^2 T}{\partial x^2}$$

T: Temperature
t: Time
x: Spatial coordinate
κ: Diffusion coefficient (assumption: $\kappa = 1$)

Subroutine from Task 2

Finite differences

$$\frac{T_i^{n+1} - T_i^n}{\Delta t} = \left(\frac{T_{i+1}^n - 2T_i^n + T_{i-1}^n}{\Delta x^2}\right) \rightarrow \quad T_i^{n+1} = T_i^n + \Delta t \cdot \left(\frac{T_{i+1}^n - 2T_i^n + T_{i-1}^n}{\Delta x^2}\right)$$

Time index $n = 1 \ldots n_t$ Space index $i = 1 \ldots n_x$

Task 3

Write a Fortran program that solves the one-dimensional heat diffusion equation.

$$\frac{\partial T}{\partial t} = \kappa \cdot \frac{\partial^2 T}{\partial x^2}$$

- For simplicity, use the assumption $\kappa = 1$.
- Initialize the length of the model domain (L), the number of grid points (nx) and the integration time (total_time) with values of your choice.
- Calculate the grid spacing $h = L/(nx - 1)$, and the time step, e.g. $\Delta t = 0.4 \cdot h^2$.
- Initialize the temperature on the grid, either with a delta function (i.e. 1 in the center, 0 everywhere else) or with random numbers: `CALL RANDOM_NUMBER(T)`
- Use the boundary condition $T = 0$ (for $i = 1$ and $i = nx$).

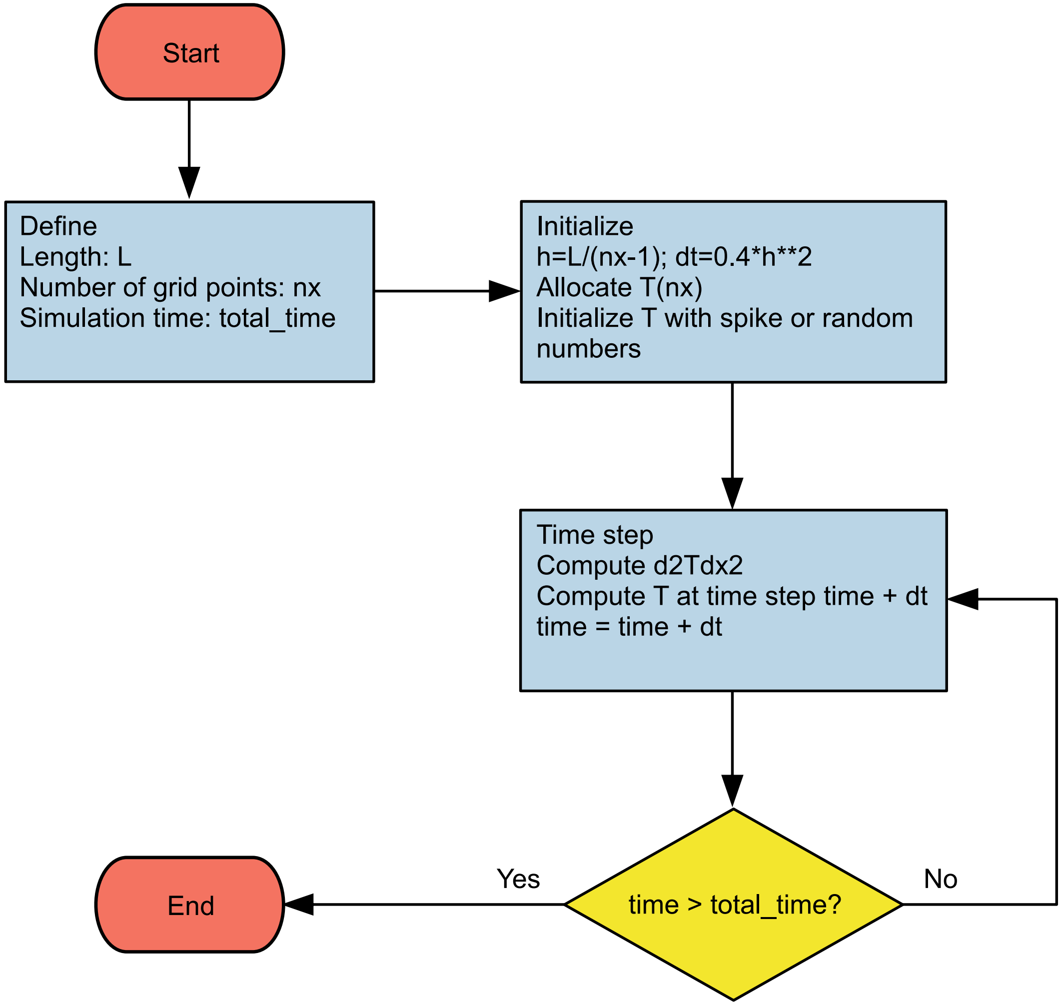
Task 3

- Integrate forward in time by calculating the temperature at each next time step for several time steps using the finite difference method:

$$T_i^{n+1} = T_i^n + \Delta t \cdot \left(\frac{T_{i+1}^n - 2T_i^n + T_{i-1}^n}{\Delta x^2}\right)$$

- For this purpose, use your module from Task 2 to calculate the second derivative of temperature with respect to x.
- Write out the temperature at the first and last time step. For now, you can use `PRINT*` and redirection of stdout (>) to write to a text file: `./a.out > output.txt`
- Plot the temperature data with your favorite plotting tool.

Flowchart for Task 3



Exercise 3

Deadline:

- Please hand in your solutions (.f90 files and plots of T) until Tuesday, **9 April 2024, 23:59** at the latest.

Questions?

- Email me (marina.duetsch@univie.ac.at)
  Or pass by my office (UZA II, 2G551).