Assignment 2.

Given September 20, due October 4.

Objective: To practice the basics of finite difference computations.

We are interested in the solution of the heat equation in the interval $[0,6]$ with Dirichlet boundary conditions $u(0,t) = 0$, $u(6,t) = 0$. Write a program that applies the forward Euler method, with parameters λ, the CFL ratio, and L, the number of grid points in the interval ($\Delta x = 6/L$). Implement the Dirichlet boundary condition by taking $U_0^n = 0$ and $U_L^n = 0$ for all n. Compute to time $T = 2$.

a. Take initial data

$$u_0(x) = \exp(-3(x-3)^2) \quad .$$

Do a convergence study to show that the method is second order accurate for general λ and fourth order accurate when $\lambda = 1/3$.

b. For non smooth initial data, it is common to take the values U_k^0 to be "cell averages" of the given initial data:

$$U_k^0 = \frac{1}{\Delta x} \int_{x_k - \Delta x/2}^{x_k + \Delta x/2} u_0(x)dx \quad . \tag{1}$$

This is supposed to give more accurate results than the pointwise formula

$$U_k^0 = u_0(x_k) \quad . \tag{2}$$

Use initial data

$$u_0(x) = \begin{cases} 1 \text{ if } |x-3| \leq \sqrt{2} \\ 0 \text{ if } |x-3| > \sqrt{2}. \end{cases}$$

Which of the formulae (1) or (2) leads to more accurate results?

c. Take

$$u_0(x) = \begin{cases} 1 + \cos(x-3) & \text{if } |x-3| \leq \sqrt{\pi} \\ 0 & \text{if } |x-3| > \sqrt{\pi}. \end{cases}$$

and recompute the order of accuracy with $\lambda \neq 1/3$ and $\lambda = 1/3$. Comment on the difference with part a.

d. Make some plots to illustrate the qualitative properties of the solution of the heat equation. These properties are

i. The smoothing property: $u(x,t)$ is a smooth function of x even when $u(x,0)$ is not.

ii. The maximum principle: the maximum of $u(x,t)$, as a function of x, is a decreasing function of t.

iii. The integral conservation property: $\int u(x,t)dx$ is independent of t. In this case, because of Dirichlet boundary conditions, that is only approximately true for small t.

Note on programming. This is partly an exercise in software engineering. If you code from scratch for each part, it will be tedious. Plan the work so that routines (such as that for convergence analysis) can be used over.