
1. Determine whether the following partial differential equations, in which \( p \) and \( q \) are arbitrary real constants, are elliptic, parabolic, or hyperbolic.

(a) \( p^2 \frac{\partial^2 \psi}{\partial x^2} + q^2 \frac{\partial^2 \psi}{\partial y^2} = 0 \)

(b) \( (p \frac{\partial}{\partial x} + q \frac{\partial}{\partial y})(p \frac{\partial}{\partial x} - q \frac{\partial}{\partial y})\psi = 1 \)

(c) \( \frac{\partial^2 \psi}{\partial x^2} + 4 \frac{\partial^2 \psi}{\partial y^2} + \frac{\partial \psi}{\partial x \partial y} \frac{\partial^2 \psi}{\partial y^2} = 0 \)

(d) \( \frac{\partial^2 \psi}{\partial x^2} + \frac{\partial^2 \psi}{\partial y^2} + \frac{\partial^2 \psi}{\partial x^2} = \psi \)

(e) \( \frac{\partial^2 \psi}{\partial x^2} + p \frac{\partial \psi}{\partial y} = \psi \)

(f) \( \frac{\partial^2 \psi}{\partial y^2} + \frac{\partial^2 \psi}{\partial x^2} = x \)

2. Write a computer code function\(^1\) to evaluate the difference stencil in two dimensions for the anisotropic partial differential operator, \( \mathcal{L} = 4 \frac{\partial^2}{\partial x^2} + \frac{\partial^2}{\partial y^2} \). The code function is to operate on a quantity \( f(x, y) = f_{ij} \), represented as a matrix of the values at discrete points on a structured, equally-spaced, 2-D mesh with \( N_x \) and \( N_y \) nodes in the \( x \) and \( y \) directions, spanning the intervals \( 0 \leq x \leq L_x \), \( 0 \leq y \leq L_y \). The function should accept parameters \( N_x, N_y, L_x, L_y, i, j, f \) and return the corresponding finite-difference expression for \( g_{ij} = \mathcal{L}f \) at mesh point \( i, j \).

Write also a test program to construct \( f(x, y) = (x^2 + y^2) \) on the mesh nodes, giving \( f_{ij} \), and call your stencil function, with \( f \) and the corresponding \( N_x, N_y, L_x, L_y \) as arguments, to evaluate \( g_{ij} \) and print it.

Submit the following as your solution:

a. Your code in a computer format that is capable of being executed, citing the language it is written in.

b. A brief answer to the following. Will your function work at the boundaries, \( x = 0, L_x \), or \( y = 0, L_y \)? If not, what is needed to make it work there?

c. The values of \( g_{ij} \) for four different nodes corresponding to two different interior \( i \) and two different interior \( j \), when \( N_x = N_y = 10, L_x = L_y = 10 \).

d. Brief answer to: Are there inefficiencies in using a code like this to evaluate \( \mathcal{L}f \) everywhere on the mesh? If so, how might those inefficiencies be avoided?

\(^1\)For OO purists, this could be a “method”.