Unidad 4

Ecuaciones diferenciales parciales. (b) Ecuaciones parabólicas.

La determinación de la variación de la temperatura a lo largo de una barra de longitud L, cuya temperatura inicial es de 100°C en tanto sus extremos se mantienen a 0°C, conduce al siguiente *problema de Cauchy para la ecuación de calor*: Hallar la función $u(x,t)$ que satisface la ecuación

$$u_t - \alpha^2 u_{xx} = 0,$$

para $0 \le x \le L$, $t \ge 0$, y que cumple la condición inicial:

$$u(x, t=0) = 100, \quad 0 \le x \le L,$$

y las condiciones homogéneas en los bordes:

$$u(x=0,t) = u(x=L,t) = 0, \quad t \ge 0.$$

Ejercicio 1. Mostrar que la solución *analítica* del problema, obtenida por el *método de separación de variables*, está dada por

$$u(x,t) = \sum_{k=0}^{\infty} \frac{400}{(2k+1)\pi} \exp\left\{ -\left[\frac{(2k+1)\pi\alpha}{L}\right]^2 t \right\} \times \operatorname{sen}\left(\frac{n\pi x}{L}\right).$$

Ejercicio 2. Una solución *numérica* del problema es obtenida por el *método de diferencias finitas*. Considerando una grilla espacio-temporal de paso Δx y Δt, tenemos la red de puntos:

$$x_i = i\Delta x, \quad i = 0, 1, \dots, n = L/\Delta x,$$
$$t_j = j\Delta t, \quad j = 0, 1, \dots.$$

Mostrar que el algoritmo de *diferencia finita progresiva* determina la aproximación $u_{i,j+1}$ de $u(x_i, y_{j+1})$ según la fórmula:

$$u_{i,j+1} = u_{i,j} + \lambda(u_{i+1,j} + u_{i-1,j} - 2u_{i,j}),$$

donde $\lambda = \alpha^2 \Delta t/(\Delta x)^2$, y se satisfacen la condición inicial:

$$u_{i,0} = 100, \quad i = 1, 2, \dots, n-1$$

y las condiciones de contorno:

$$u_{0,j} = u_{n,j} = 0, \quad j = 0, 1, \dots$$

Ejercicio 3. El programa, cuyo código se muestra más adelante, implementa la determinación de la solución numérica del problema planteado, tomando $L = 1$ y $\alpha = 1$.

a) Determine la solución numérica con $n = 49$ y un paso Δt que satisfaga la *condición de estabilidad*: $\lambda < 1/2$. Grafique la misma junto con las curvas de nivel $u =$ cte, esto es, las *isotermas*.

b) Compare la solución numérica con la solución analítica.

c) Repita el proceso para otros tamaños de paso, en particular investigue como se comporta la solución numérica cuando no se satisface la condición de estabilidad.

```
      program calor
*     -----------------------------------------------------
*     Solución numérica de la ecuación del calor
*     -----------------------------------------------------
*     Declaración de tipo
*     -----------------------------------------------------
      implicit none
      integer maxpoints ! Máximo tamaño de la red
      parameter (maxpoints = 1000)
      double precision u(0:maxpoints,0:1)
      double precision long,deltax,deltat,alpha2,lambda
      integer n,tpasos,twrite,i,j
*     -----------------------------------------------------
*     Parametros del problema
*     -----------------------------------------------------
      long    = 1.0d0  ! Longitud de la barra
      alpha2  = 1.0d0  ! Coeficiente de la ec. diferencial
      n       = 49     ! N+1 puntos de la malla en x
      deltax  = long/n ! Paso de la malla en el espacio
      deltat  = 2e-4   ! Paso de la malla en el tiempo
      tpasos  = 2000   ! Número de pasos en el tiempo
      twrite  = 100    ! Imprimir cada twrite pasos temporales
      lambda  = alpha2*deltat/deltax**2
*     -----------------------------------------------------
*     Asignar condiciones de contorno y condiciones iniciales
*     -----------------------------------------------------
      u(0,0) = 0.0d0  ! Nivel 0 del tiempo
      u(n,0) = 0.0d0
      u(0,1) = 0.0d0  ! Nivel 1 del tiempo
      u(n,1) = 0.0d0
      do i=1,n-1
          u(i,0) = 100.0d0
      enddo
*     -----------------------------------------------------
*     Imprimir condición inicial
*     -----------------------------------------------------
      do i=0,n
          write(*,*) i*deltax,0,u(i,0)
      enddo
      write(*,*)
*     -----------------------------------------------------
*     Determinar solucion al siguiente tiempo en puntos interiores
*     -----------------------------------------------------
      do j = 1,tpasos
          do i=1,n-1
              u(i,1) = u(i,0)+lambda*(u(i+1,0)+u(i-1,0)-2.0d0*u(i,0))
          enddo
         if (mod(j,twrite).eq.0) then ! Imprimir solución cada twrite pasos en el tiempo
              do i=0,n
                 write(*,*) i*deltax,j*deltat,u(i,1)
              enddo
              write(*,*)
          endif
          do i=0,n
              u(i,0) = u(i,1) ! Reasignar nivel 0 del tiempo
          enddo
      enddo
      end
```