```
(*                          Parabolic and Hyperbolic PDE

  Task1: 1.1. Consider the heat equation ∂t u(x,t)-c²∂x,x u(x,t)=
 0 for c²=1.The length of the rod is L=
   1.Assume that the ends of the rod are held at the temperature t=
    0 (u(0,t)=u(1,t)=0).Assume that the initial temperature distribution is
    u(x,0) = φ(x) = sin(π x)+sin(3π x). Apply the Crank-
        Nicolson method for a squared grid and obtain temperature
         distributions for t=0,0.01.,0.02,0.03,⋯,0.10.
          1.2 Compare the solution with the exact solution:u(x,t) =
 sin(π x)ⅇ^(-tπ²)+  sin(3π x)ⅇ^(-9tπ²).    *)
Clear["`*"];
a = 0; b = 1; c = 0; d = 0.10; coef = 1;
f[x_, y_] = 0;
m = 11; n = 11;
h = (b - a)/m; k = (d - c)/n; r = (k * coef^2)/h^2;
ω = 2/(1 + Sqrt[1 - (r/(1+r) Cos[π/m])^2]);
Print["The optimal weighted average is ", N[ω]];
(* 1.1 *) (* The list of points,
where x0 and y0 are replaced by x1 and y1, respectively: *)
x1 = a;  y1 = c;
For[i = 1, i ≤ m, i++,
  x(i+1) = x(i) + h; Print["x"(i+1), "=", x(i+1)];];
For[j = 1, j ≤ n, j++,
  y(j+1) = y(j) + k; Print["y"(j+1), "=", y(j+1)];];
(* As a consequence of the increased initial index,
the last point will be given by (x(m+1),y(n+1)). *)
(* Initial conditions: *)
u = Table[5.00, {m +1}, {n +1}];
For[ i = 1, i <= m +1, i++, u[[i,1]] = Sin[π x(i)] + Sin[3 π x(i)] ];
For[ j = 1, j <= n +1, j++, u[[1,j]] = 0];
For[ j = 1, j <= n +1, j++, u[[m +1,j]] = 0 ];
u0 = u; Print["u=", MatrixForm[N[u]]];
Err = 1.0;
While[ And[Err > 0.001 , k ≤ 50],
  Err = 0.0;
  For[ j = 1, j ≤ n, j++,
   For[ i = 2, i ≤ m, i++,
     u[[i,j+1]] = r/(2 (1 + r)) ω (u0[[i+1,j]] + u[[i-1,j]] + u0[[i,j+1]] +
          u[[i-1,j+1]] + (2 (1 - r))/r u0[[i,j]] + 2 h² r/coef² f[x(i), y(j)]) + (1 - ω) u0[[i,j+1]];
     Err = Max[Err, Abs[u[[i,j]] - u0[[i,j]]]]; ]; ];
  u0 = u;
  Print["Max grid change = ", Err];
  k = k + 1; ];
```

```
Print["The numerical solution to the P.D.E."];
Print["∂tu(x,t)-∂x,xu(x,t)=0"];
ListPlot3D[Transpose[u], AxesLabel → {"t", "x", "u   "},
  ViewPoint → {4, 2, 3}, Lighting → False, ColorFunction → Hue];

Print["The numerical solution to the P.D.E."];
Print["∂tu(x,t)-∂x,xu(x,t)=0"];
Print[NumberForm[TableForm[N[u]], 4]];

Print["The numerical solution to the P.D.E."];
Print["∂tu(x,t)-∂x,xu(x,t)=0"];
ListContourPlot[Reverse[u], ColorFunction → Hue];

(* 1.2 *) (* The analytic solution: *)
U[x_, t_] = Sin[π x] e^(-t π^2) + Sin[3 π x] e^(-9 t π^2);
Print["The analytic solution to the P.D.E."];
Print["ut(x,t) - uxx(x,t)=0 is"];
Print["u[x,t] = ", U[x, t] ];
SetOptions[Plot3D, PlotPoints → {n, m}];
Plot3D[U[x, t], {t, 0, 0.2}, {x, 0, 1.0},
  ViewPoint → {4, 2, 3}, Lighting → False, ColorFunction → Hue];
Uex = Table[U[xi, yj], {i, 1, m + 1}, {j, 1, n + 1}];
Print["The exact solution is: "];
Print["Uex= ", N[Uex // MatrixForm]];
Print["The largest error is: ", Max[Abs[u - Uex]]];

(* Observation.The numerical solution to
 the P.D.E.was computed on a "grid" in a matrix.Hence,
we have "lost" the connection between the "x" and "t" variables when
 plotting the solution.This problem is independent of the software
 used. If "tick marks" corresponding to "x" and "t" are necessary,
then it would be necessary to rewrite these commands in the
 software to handle this particular situation. Otherwise,
the automatic graphing of lists of data will automatically
 decide what numbers to put on the "axes." *)

(*   Task2: 2.1. Consider the wave equation ∂t,tu(x,t)-c²∂x,xu(x,t)=
 0 for c²=4.The length of the string at rest is L=1.Assume that the initial
     conditions are u(x,0) = φ1(x) = sin(π x)+ sin(2π x) and ∂t u(x,t)|t=0=
      φ2(x)=0. The boundary conditions u(0,t) and u(1,t) are set to 0.
        Apply the Crank-Nicolson method for a 20x20 grid and t∈[0,0.1].
            2.2 Compare the solution with the exact solution:u(x,t) =
        u(x,t) = sin(π x)cos(π t)+ sin(2π x)cos(4π t).   *)
Clear["`*"];
a = 0; b = 1; c = 0; d = 0.1; coef = 2;
f[x_, y_] = 0;
m = 20; n = 20;
h = (b - a)/m; k = (d - c)/n; r = (k^2 * coef^2)/h^2;
ω = 2/(1 + Sqrt[1 - (r/(1+r) Cos[π/m])^2]);

Print["The optimal weighted average is ", N[ω]];
```

```
(* 2.1 *) (* The list of points,
where x0 and y0 are replaced by x1 and y1, respectively: *)
x1 = a;  y1 = c;
For[i = 1, i ≤ m, i++,
  xi+1 = xi + h; Print["x"i+1, "=", xi+1];];
For[j = 1, j ≤ n, j++,
  yj+1 = yj + k; Print["y"j+1, "=", yj+1];];
(* As a consequence of the increased initial index,
the last point will be given by (xm+1,yn+1). *)
(* Initial conditions: *)
u = Table[5.00, {m + 1}, {n + 1}];
For[ i = 1,  i <= m + 1,  i++, u[[i,1]] = Sin[π xi] + Sin[2 π xi] ];
For[ j = 1,  j <= n + 1,  j++, u[[1,j]] = 0];
For[ j = 1,  j <= n + 1,  j++, u[[m +1,j]] = 0 ];
φ2[x_] = 0;
For[ i = 2,  i <= m,  i++,
  u[[i,2]] = u[[i,1]] + k * φ2[ xi] + k^2/2 (coef^2 (u[[i+1,1]] - 2 u[[i,1]] + u[[i-1,1]])/h^2 + f[xi, c])];
u0 = u;  Print["u=", MatrixForm[N[u]]];
Err = 1.0;
While[ And[Err > 0.001 , k ≤ 50],
  Err  =  0.0;
  For[ j = 2,  j ≤ n,  j++,
   For[ i = 2,  i ≤ m,  i++,
      u[[i,j+1]]  =  r/(2 (1 + r)) ω (u0[[i+1,j+1]] + u[[i-1,j+1]] + (u0[[i+1,j-1]] + u[[i-1,j-1]] - 2 u0[[i,j-1]]) +
           2/r (-u0[[i,j-1]] + 2 u0[[i,j]] + k^2 * f[xi, yj])) + (1 - ω) u0[[i,j+1]];
      Err  =  Max[Err, Abs[u[[i,j+1]] - u0[[i,j+1]]]];  ];  ];
  u0 = u;
  Print["Max grid change  =  ", Err];
  k = k + 1;  ];
Print["The numerical solution to the P.D.E."];
Print["∂t,tu(x,t)-4∂x,xu(x,t)=0"];
ListPlot3D[Transpose[u], AxesLabel → {"t", "x", "u    "},
  ViewPoint → {4, 2, 3}, Lighting → False, ColorFunction → Hue];
Print["The numerical solution to the P.D.E."];
Print["∂t,tu(x,t)-4∂x,xu(x,t)=0"];
Print[NumberForm[TableForm[N[u]], 4]];
Print["The numerical solution to the P.D.E."];
Print["∂t,tu(x,t)-4∂x,xu(x,t)=0"];
ListContourPlot[Reverse[u], ColorFunction → Hue];
```

```
(* 2.2 *) (* The analytic solution: *)
U[x_, t_] = Sin[π x] Cos[π t] + Sin[2 π x] Cos[4 π t];
Print["The analytic solution to the P.D.E."];
Print["∂t,t u(x,t) - 4∂x,x u(x,t)=0 is"];
Print["u[x,t] = ", U[x, t] ];
SetOptions[Plot3D, PlotPoints → {n, m}];
Plot3D[U[x, t], {t, 0, 0.2}, {x, 0, 1.0},
  ViewPoint → {4, 2, 3}, Lighting → False, ColorFunction → Hue];
Uex = Table[U[xi, yj], {i, 1, m + 1}, {j, 1, n + 1}];
Print["The exact solution is: "];
Print["Uex= ", N[Uex // MatrixForm]];
Print["The largest error is: ", Max[Abs[u - Uex]]];
```