1. Write a C program to solve the following equation (this is the so-called one-dimensional heat equation)

$$\frac{\partial U}{\partial T} - k * \frac{\partial^2 U}{\partial X^2} = F(X,T)$$

over the interval [A,B] with boundary conditions

```
U(A,T) = UA(T),
U(B,T) = UB(T),
```

and over the time interval [T0,T1] with initial conditions

```
U(X,T0) = U0(X)
```

The above equation describes how temperature changes across a bar (say, an iron bar) with time. Imagine that the bar is so thin that we can treat the bar as a one-dimensional object. The equation also expects that we are applying some heating to the bar at its ends (the boundary condition). The problem also has an initial condition; this describes the temperature of the bar at the initial time.

We can approximate the above (partial differential) equation as a finite difference equation as follows:

```
U(X,T+dt) - U(X,T)                     ( U(X-dx,T) - 2 U(X,T) + U(X+dx,T) )
------------------  = F(X,T) + k *  -----------------------------------------
        dt                                          dx * dx
```

If we rearrange the equation, we get an equation for U(X,T+dt). Put simply, the following equation means that if we know the present value of U, i.e., U at X and T, then we can compute the future value of U (i.e., U at X and T+dt) from the right hand side of the equation.

```
U(X,T+dt) = U(X,T)
  + cfl * ( U(X-dx,T) - 2 U(X,T) + U(X+dx,T) ) + dt * F(X,T)
```

Here "cfl" is the Courant-Friedrichs-Loewy coefficient:

```
cfl = k * dt / dx / dx
```

In order for accurate results to be computed by this explicit method, the CFL coefficient must be less than 0.5.

Start the simulation by specifying initial and boundary conditions. For the initial conditions, set the temperature to 50 C throughout the domain. As for *the boundary conditions, specify 90 C for the left side of the bar, and 70 C for the right side. Plot the results as a function of x and t using gnuplot or any other graphical software of your choice.*

2. Create a parallel version of the program using MPI and show that the results of this program is the same as the one created in 1 above. The simplest way is to plot both results together.