Q 1) [10 points] The program heatExp.py solves the partial differential equation $\frac{\partial u}{\partial t} = \frac{\partial^2 u}{\partial x^2}$ by the explicit method, subhect to the boundary conditions $u\left(0,t\right) = u\left(L,t\right) = 0$ and the initial condition

$$u\left(x,0\right) = \begin{cases} 2h\frac{x}{L} & x < \frac{L}{2} \\ 2h\left(1 - \frac{x}{L}\right) & x \geq \frac{L}{2} \end{cases}$$

where we have used $h = L = 1$. The code is nearly complete, except for one crucial line.

i) Correct the code and run it. [4 points]

ii) Change the boundary condition to $u\left(0,t\right) = 0$ and $u\left(L,t\right) = A\sin^2\left(\omega t\right)$. Use the values $A = 1$,and $\omega = 25$. Take a screenshot of the output at any instant of your choice. [3 points]

iii) Experiment with the value of Δt to try and pinpoint the minimum value where instability sets in (to see what instability looks like, just use the larger value of Δt that is commented out in the given program). [3points]

Q 2) [10 points] The code heatImp.py uses the implicit algorithm to solve the same problem. Note that the set of simultaneous equations is solved here simply using numpy's inverse function to invert a matrix. Pay attention to how a non-zero boundary condition is handled here. Modify the code in heatImp.py to solve this problem using the Crank-Nicolson algorithm. Run it for the same initial value in Q 1 , and for both sets of boundary conditions in 1i and 1ii.