Math 712 Midterm Project
Due: Monday Nov. 3

Instructions: *1. Everything must be returned in report form and must be type-written.*
2. You must submit your code.
3. Late solutions are NOT accepted.

Problem:
Consider solving the simple heat equation

$$\frac{\partial u}{\partial t} = \frac{\partial^2 u}{\partial x^2}$$

using the so-call "theta method":

$$\frac{v_j^{n+1} - v_j^n}{\Delta t} = \frac{\theta}{\Delta x^2}(v_{j+1}^{n+1} - 2v_j^{n+1} + v_{j-1}^{n+1}) + \frac{(1-\theta)}{\Delta x^2}(v_{j+1}^n - 2v_j^n + v_{j-1}^n).$$

Part (a) Analysis:
(1) Derive the necessary and sufficient condition for stability (dependent on θ);
(2) Find the truncation error of the scheme, discuss the highest order of accuracy that can be achieved by varying θ for a fixed Courant number $\sigma = \frac{\Delta t}{\Delta x^2}$; also discuss the special case where $\sigma = 1/\sqrt{20}$;

Part (b) Implementation:
Implement the above scheme to solve the initial-boundary value problem for the heat equation with two different boundary conditions:
(i) general mixed boundary conditions:

$$\alpha_1 u(a,t) + \beta_1 \frac{\partial u(a,t)}{\partial x} = f_1(t), \qquad \alpha_2 u(b,t) + \beta_2 \frac{\partial u(b,t)}{\partial x} = f_2(t);$$

(ii) periodic boundary conditions:

$$u(a,t) = u(b,t),$$

where a, b are the end points.

(1) Write a subroutine to solve a general tridiagonal linear system using the so-called Thomas algorithm;

(2) For some boundary conditions, the resulting linear system may not be strictly tridiagonal. Instead, the matrix is a tridiagonal matrix plus a low rank perturbation. The famous and very useful Sherman-Morrison-Woodbury formula states that:

$$(A + UCV)^{-1} = A^{-1} - A^{-1}U(C^{-1} + VA^{-1}U)^{-1}VA^{-1}.$$

where A is $n \times n$, U is $n \times k$, C is $k \times k$, and V is $k \times n$. Combine the above formula and the Thomas algorithm to implement a linear algorithm to solve the resulting linear system for the initial-boundary value problem.

Part (c) Testing:
Construct an initial-boundary value problem whose exact analytical solution is known. Use the particular problem to test your code.
(1) Justify all claims you made in part (a);
(2) Verify that your algorithm is indeed a linear algorithm with respect to the number of unknowns.
You may present your results using either tables or graphs with clear captions. Do **not** simply plot out the solutions without any explanations, this does **not** serve for the purpose of justification or verification!