MSc in CSTE
Computational Methods & C++ Assignment

Irene Moulitsas & Peter Sherar
Cranfield University

October 6, 2017

Hand in date: 05/12/17 (FT), 19/12/17 (PT), 09:30am

1 Introduction

In this assignment you are asked to examine the application of numerical schemes for the solution of partial differential equations as discussed in the Computational Methods lectures, using C++ Object Oriented and other programming practices discussed in the C++ lectures. In order to do this, we will consider the following problem.

A wall 1 ft. thick and infinite in other directions (see Figure 1) has an initial uniform temperature T_{in} of 100°F. The surface temperatures T_{sur} at the two sides are suddenly increased and maintained at 300°F. The wall is composed of nickel steel (40% Ni) with a diffusivity of $D = 0.1\ ft^2/hr$. Please compute the temperature distribution within the wall as a function of time.

The governing equation to be solved is the unsteady one-space dimensional heat conduction equation, which in Cartesian coordinates is:

$$\frac{\partial T}{\partial t} = D\frac{\partial^2 T}{\partial x^2}$$

2 Computational Methods Tasks

1. Write a C++ program which solves the above problem on a uniform grid with the prescribed initial and boundary conditions using the following methods:

    - DuFort-Frankel
    - Richardson (Central time, central space, explicit)
    - Laasonen Simple Implicit (Forward time, central space, implicit)
    - Crank-Nicholson (Trapezoidal)

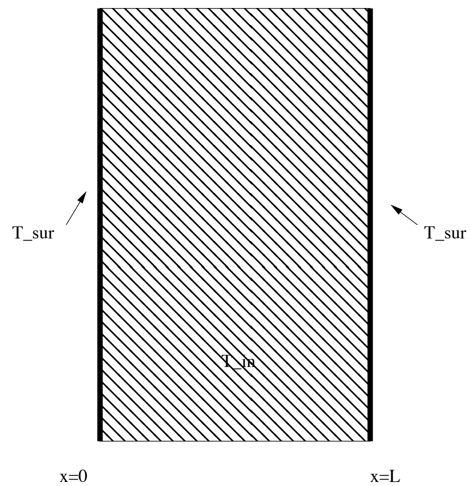

Figure 1: Nomeclature and Problem Domain

For each method you may assume that $\Delta x = 0.05$ and $\Delta t = 0.01$

2. The analytical solution of this problem, subject to the imposed initial and boundary conditions, is

$$T = T_{sur} + 2(T_{in} - T_{sur}) \sum_{m=1}^{m=\infty} e^{-D(m\pi/L)^2 t} \, \frac{1-(-1)^m}{m\pi} \, \sin\left(\frac{m\pi x}{L}\right)$$

   Use the analytical solution to compare/validate the results of the above numerical methods. Comparisons should be both *qualitative* and *quantitative*.

   In all cases the solution is to be printed and plotted for all x locations every $0.1 hrs$ time intervals from 0.0 to $0.5 hrs$.

3. Investigate the effect of step size on the accuracy of the solution and required computation time of an implicit method, using the Laasonen simple implicit method, with the following time steps sizes:

   - $\Delta t = 0.01$
   - $\Delta t = 0.025$
   - $\Delta t = 0.05$
   - $\Delta t = 0.1$

   For all cases assume $\Delta x = 0.05$.

4. Explain the behaviour of the solutions of the above numerical methods in terms of the expected properties of the methods involved. In particular for the Richardson method, study the accuracy and stability properties in depth and include your detailed mathematical calculations in the Appendix of your report.

3 Object Oriented Programming Tasks

Think about the design of your solution before jumping into coding.

- What classes (abstractions) will you use?
- What will be the separation of responsibilities into the chosen classes i.e what is each class going to be responsible for doing?
- What will be the data/methods encapsulation for each class?
- What are the relationships between the chosen classes: aggregation ('has a' or 'contains'), inheritance ('is a kind of'), association ('uses' - one class may use objects of another class in the implementation of its methods)?

Some desirables

- Try to adhere to the SOLID principles for effective design (see BB pages for C++).
- Try to use the Standard Library components where appropriate (containers, iterators, algorithms, numerics). This is well designed, thoroughly tested, robust and efficient code.
- Try to use exceptions to deal with exceptional conditions and catch them at an appropriate higher level.
- Try not to overcomplicate things. Keep things simple but effective.

Documentation is required (see Vector/Matrix classes used in the module for examples of this).

Doxygen documentation for the classes and concise in line comments in the functions/methods to explain what 'chunks of code are doing.

Finally, remember there is no right or wrong answer when it comes to the design but there are better and worse designs. In terms of software, you are aiming for a clean and effective solution to the problem.

4 Report

Write a report to present and discuss your findings. The report should be no less than 2,500 words and must not exceed 5,000 words. The report can contain any number of figures. All figures and tables in the report should be numbered and discussed. The report should include a description of the design of your solution explaining your choices and incorporating a UML class diagram. The source code with the doxygen documentation files should be included as an Appendix to the report.

The report should be submitted electronically via the **TurnItInUK** link by 9:30am on the 5th December (full-time students) or the 19th December (part-time students).

The source code and documentation files must also be submitted via the **Blackboard** link by the prescribed deadline, for the assignment submission to be considered complete.

The source program will need to compile on the IT lab machines using Visual Studio 2015 or 2013 without any other external dependencies/libraries/source codes of third parties.

5 Marking

The assignment will be assessed based on the following marking scheme:

- 20% Introduction, methodology, conclusions
- 40% Source code, documentation
- 30% Analysis of the results and derivation of theoretical properties
- 10% Report structure, presentation, references

6 References

K.A. Hoffmann and S.T. Chiang, 'Computational Fluid Dynamics', Fourth Edition, Vol. I, Engineering Education System Books, pp. 486, 2000.
S. Scott Collis, 'An Introduction to Numerical Analysis for Computational Fluid Dynamics', Technical Report SAND2005-2745, Sandia National Laboratories, 2005.
Links to the following two documents can be found on the C++ BB page under external links:
Pras Pathmanathan, 'Numerical Methods and Object-oriented Design'.
Dr O Gloth, 'Object Oriented Techniques and Numerics'.