Math 2441 – Introduction to Mathematical Computation
Project Report Format

Section 1. Background, Historical Development, Applications

In this section, you should provide the context for the mathematical algorithms used in the project – why was it developed, what are its applications, in what fields or professions is it used. Are there any extensions or simplifications of this mathematical concept that were not covered in this project? If you have found any online resources or simulators, you should reference that information here.

If the algorithms were named after certain individuals, you should provide a brief biographical statement about that individual, including lifespan, country of origin or nationality, and prominent location where the work was performed.

Section 2. Description of the Mathematical Concepts and Algorithms

You should define all variables and parameters and provide a discussion about valid values (i.e., the domain) for these variables. You should include all relevant formula used within this mathematical concept. The variables and constants in the formula should be defined. You should state important theorems, lemmas, corollaries and conjectures, relevant to this topic.

Section 3. Translation of Mathematical Algorithm into Pseudo-Code

Provide a step-by-step pseudo-code for each mathematical algorithm that you implemented within a computational tool. Use the format

```
Algorithm  <Name of Algorithm>
    Inputs  <List of Inputs>
    Outputs  <List of Outputs>
        1.
        2.
        3.
        …
```

<u>You should provide a brief discussion about the selection of the computational tool for implementing this algorithm</u>. For instance, exact symbolic arithmetic requires Mathematica. Computational efficiency and the use of repetition may necessitate MatLab or C++. Visual results such as graphs of curves cannot be obtained directly from C++, but can be obtained from other software. Nicely format output of all of the results can be performed well in Microsoft Excel. You should discuss these needs in the context of the choice of computational tool(s).

Section 4. Verification Results

In this section, you should provide convincing evidence that your computational implementation of the mathematical algorithm is correct. This verification is typically performed by selecting appropriate cases or examples that test the various capabilities of the algorithm and the computational implementation. You are free to select your own test cases, as long as the following criteria are met:

1. Test cases should have exact answers that can be determined outside of that computational tool (i.e., there is a formula that can provide the answer).
2. Test cases should demonstrate the veracity of each aspect of the computational tool, from extremely simple to complicated, as long as an exact answer can be obtained.

Please include the test cases, the exact answers and a statement that declares that the results from the computational tool agree with the exact answers. If you are not able to make such a statement, please explain why. In this explanation, please provide reasons for the lack of agreement.

Section 5. Validation Cases

Once the computational implementation has been verified to provide the same answers as mathematically defined formula, we have confidence that the results of the computational implementation are valid for other cases outside of those solved by pen-and-paper techniques. Provide the results for several validation cases and discuss why the results from the computational implementation are consistent with what you would expect for these cases.

Answer any additional questions that are provided in the programming lab description. These questions will focus on the overall observed behavior of the mathematical algorithms.