Mathematical Computing
Math 309.01

David Meredith
Department of Mathematics
San Francisco State University
San Francisco, CA 94132
meredith@sfsu.edu

Fall 2001—MWF 12-1—BUS 130
Contents

1 Introduction .................................. 2
  1.1 Instructor ................................ 2
  1.2 Web Site and E-mail .......................... 2
  1.3 Text .................................... 2
  1.4 Software ................................ 2
  1.5 Class procedures ........................... 3
    1.5.1 Attendance ............................. 3
    1.5.2 Schedule ................................ 3
    1.5.3 Homework .............................. 3
    1.5.4 Exams ................................ 4
    1.5.5 Grading ................................. 5
  1.6 Outline .................................. 6
Chapter 1  Introduction

Mathematical Computing (Math 309) will teach you to use computer-aided calculations to help solve math problems. You will learn what the computer can and cannot do, and you will learn how to devise a plan for solving a problem that includes steps to be performed by the computer. You will also learn how to make the computer perform these steps using Mathematica, but that is the easy part. The hard part is dividing a problem into computable steps.

You will learn two methods for computing, interactive and programming. The interactive method will be introduced first. With this method you do one calculation or graph at a time and immediately see the result. After the first seven weeks we will concentrate on programming, a method of writing a complicated prescription for a calculation and then having the computer follow it. Interactive computing is fun, but it limits you to a small number of steps. With programming you can execute a plan that requires thousands or even millions of steps, something that was impossible before the introduction of the stored-program computer.

1.1  Instructor

David Meredith  
TH 933  
Office Hours: MWF 11-12  
(415) 338-2199  
E-mail: meredith@sfsu.edu  
URL: http://online.sfsu.edu/~meredith

1.2  Web Site and E-mail

Class notes, handouts, syllabus changes, grade sheet, etc. can be found at the class web site, which you can access by going to my web site and clicking on Mathematical Computing. I will not distribute any more paper handouts to you. All information will be distributed via the class web site.

When you cannot get to my office hour or wait for the next class, you may submit questions to me by e-mail. If you cannot make it to class, you can submit homework by e-mail.

1.3  Text


This is not the easiest book about Mathematica, but it has the most useful information of any I could find.

1.4  Software

In this course you will learn to calculate and program with Mathematica. If you want to buy your own copy, the Bookstore will sell you the student version for $139. Plus
tax. The commercial version costs $1495, so the student version is a bargain. The
student version is the same as the commercial version (I think), except that the phrase
"printed by student version of Mathematica" appears at the bottom of each page you
print.

1.5 Class procedures

1.5.1 Attendance

I take attendance every day.

1.5.2 Schedule

<table>
<thead>
<tr>
<th>Topics</th>
<th>Due Date</th>
</tr>
</thead>
<tbody>
<tr>
<td>Notebooks, calculations and 2D graphs</td>
<td>9/14</td>
</tr>
<tr>
<td>3D graphs, lists and sums</td>
<td>9/28</td>
</tr>
<tr>
<td>Data analysis and calculus</td>
<td>10/12</td>
</tr>
<tr>
<td>Algebra and functions</td>
<td>10/26</td>
</tr>
<tr>
<td>Midterm</td>
<td>10/26</td>
</tr>
<tr>
<td>Declarative programming</td>
<td>11/9</td>
</tr>
<tr>
<td>Recursion and functional programming</td>
<td>11/23</td>
</tr>
<tr>
<td>Pattern matching and floating point numbers</td>
<td>12/14</td>
</tr>
<tr>
<td>Final (Wednesday, 10:45-1:15)</td>
<td>12/19</td>
</tr>
</tbody>
</table>

1.5.3 Homework

Every two weeks I will assign some Mathematica exercises and a small number of hard
problems. The assignment will be written in a Mathematica notebook that you will
download from the class web site. You will fill in the answers and return the notebook
to me either printed or electronically by e-mail.

The first part of the notebook will be the exercises. You should just fill in the
required calculations or graphs in a cell below each question.

The second part of the notebook will contain the problems. Your answer to each
problem must be a short formal report written in correct mathematical English. You
goal is to use text cells to explain what you are calculating and why. Actual calcula-
tions and graphs will be part of your answer, but only a small part. The explanatory
part should be much larger. The most important criteria for a good report is that
it is clear and without errors. It is better to write a grammatical, well-organized
report that correctly answers part of a question than to write a report that purports
to answer an entire question but is either fragmentary or contains mistakes. The test
of good writing is that a math student at another college should be able to read your
paper and understand what problem you were doing and how you answered it.

Papers are graded basically on how much they contain that is correct less how
much they contain that is not. There is no partial credit for mistakes. However, there
is lots of partial credit for correctly stating and answering part of a problem. If you
cannot do a problem as stated, then you might:

1. do a simpler version. If the problem asks about $x^n \sin x$, you might solve it for
   $x^3 \sin x$.
2. do an example. If the problem asks you to prove something about polynomials,
   you can show that the property holds for one polynomial.
3. discuss the problem. Prove what you can and explain why you are stuck.

The important thing is to write a well-organized report that is internally correct, even
if it does not completely answer the problem posed.
You do not have to take up the problem parts in the order I present them. You can organize your report any way you like, so long as it is clear. You can use a part that you cannot prove to prove another part, so long as you are clear about what you are doing.

Papers will be graded on a scale of 0-4

- 0: not submitted, no attempt to write complete sentences, or no correct parts.
- 1: many mathematical or grammatical errors
- 2: some parts done correctly, not too many mathematical or grammatical errors
- 3: most parts done correctly and few mistakes, writing is grammatically correct
- 4: almost all parts done correctly and no mistakes, writing is clear and well-organized

Late papers will not be accepted except by prior arrangement. If you cannot come to class, have a friend bring your work or send it by email. Write your work up as you go along. If something happens and you cannot finish, you will have a partially completed assignment to turn in.

I hope that you will work together on the problems. However, you must write your own answers. If there is significant copying between two papers, both will receive zeros. You should help each other do the problems, but you should not let anyone copy your answers.

1.5.3.1 Rules for Writing Problem Answers

1.5.3.1.1 Things to do

1. Write standard mathematical English, which means sentences organized into paragraphs. Start each paragraph at the left margin, but do not start each sentence at the left margin. Every word on your page must be part of a sentence.

2. Each part of your answer should begin with a statement of what you will prove in that part.

3. If you are only doing part of a problem or a special case, then your introductory statement should say exactly what you will do (not what you won’t do). If the problem asks you to prove something for $x^n \sin x$ and you do the problem for $x^3 \sin x$, then your statement should start: “Let $f(x) = x^3 \sin x$. Then ...”.

4. Finish each part with a conclusion.

5. Use standard mathematical notation in the text sections. Mathematica will properly format fractions, exponents, etc.

1.5.3.1.2 Things to Avoid

1. Never turn in a paper that includes unfinished problems, fragments, scratchwork, or other clutter. The reader will try to read everything on your page, so make sure that everything on your page is clear and complete. If Mathematica gives you some output that you don’t want, delete it from your notebook.

2. Do not use abbreviations or other informal grammatical elements. Prepare a manuscript that you would be proud to see published.

1.5.4 Exams

Exam problems will be similar to the easier homework problems.

1.5.4.1 Midterms

There will be one midterm on . You may use one $8\frac{1}{2} \times 11$” page of notes (both sides) for each test.
1.5.4.2 Final

The final will be in the regular lecture room on Wednesday, Dec. 19, at 10:45. You may use two 8.5" × 11" page of notes (both sides) for the final.

1.5.5 Grading

<table>
<thead>
<tr>
<th>Grading System</th>
<th>Percentage</th>
</tr>
</thead>
<tbody>
<tr>
<td>Attendance</td>
<td>10%</td>
</tr>
<tr>
<td>Problems</td>
<td>60%</td>
</tr>
<tr>
<td>Midterm</td>
<td>10%</td>
</tr>
<tr>
<td>Final Exam</td>
<td>20%</td>
</tr>
</tbody>
</table>

Final grades will be assigned according to a scale no harsher than: $A \geq 85\%$, $B \geq 70\%$, $C \geq 60\%$. 
1.6 Outline

1. Notebooks, Basic Calculations and Graphs
   (a) Starting and stopping.
   (b) Opening and saving files.
   (c) Different types of cells in a notebook
   (d) Numbers: integers, fractions and reals
      i. Complex numbers
   (e) Basic arithmetic calculations
   (f) Variables—use at least two letters. Use descriptive names.
   (g) Standard functions.
   (h) Help file.
   (i) Entering mathematical formulas as part of text

2. 2D graphs. Packages.
   (a) Plotting one or more functions.
   (b) Common options
   (c) Polar graphs and packages. What to do when you forget to load a package.
   (d) Parametric graphs.
   (e) Combining graphs. Hiding graphs.

3. 3D graphs and options
   (a) Parametric 3D graphs
   (b) Moving graphs

4. Sums and lists.
   (a) Creating
   (b) Combining
   (c) Applying functions element by element and globally
   (d) Extending
   (e) Extracting
   (f) ListPlot
   (g) Matrices and vectors.

   (a) reading data files
   (b) averages and standard deviations
   (c) histograms, pie charts
   (d) linear regression

6. Derivatives and Integrals. Limits. Solving Equations

7. Algebra

8. functions including Map and Apply

9. Declarative Programming

10. Declarative programming
11. Declarative Programming and recursion
12. functional programming
13. functional programming
14. Pattern matching
15. Real Number Precision.