Math/CS 466/666

466/666 NUMERICAL METHODS I (3+0) 3 credits

```
Instructor  Course Section                      Time           Room
---------------------------------------------------------------------
Eric Olson  Math 466/666 Numerical Methods I   MWF noon-12:50pm  DMS106
                                                due to COVID the first
                                                week is on Zoom
```

Click here ↗ to view the online syllabus for this class.

The username/password to access the class handouts is unr/unr.

Course Summary:

Date	Details	Due
Mon Aug 29, 2022	📅 MATH 466.666.1001 Combined (MATH) Sections	12pm to 1pm
Wed Aug 31, 2022	📅 MATH 466.666.1001 Combined (MATH) Sections	12pm to 1pm
Fri Sep 2, 2022	📅 MATH 466.666.1001 Combined (MATH) Sections	12pm to 1pm
Mon Sep 5, 2022	📅 MATH 466.666.1001 Combined (MATH) Sections	12pm to 1pm
Wed Sep 7, 2022	📅 MATH 466.666.1001 Combined (MATH) Sections	12pm to 1pm
Fri Sep 9, 2022	📅 MATH 466.666.1001 Combined (MATH) Sections	12pm to 1pm

[handwritten: first week online]

[handwritten: line stream from classroom...]

Course References:

Hosking, Joe, Joyce and Turner, *First Steps in Numerical Analysis*, 2nd Edition, 1996, Arnold. *[handwritten: almost too easy...]*

Justin Solomon, *Numerical Algorithms: Methods for Computer Vision, Machine Learning and Graphics,* CRC Press, 2015.

Endre Suli, David F. Mayers, *An Introduction to Numerical Analysis*, 1st Edition, Cambridge University Press, 2003.

Anthony Ralston and Philip Rabinowitz, A First Course in Numerical Analysis, Second Edition, Dover, 1978. *[handwritten: Good level... not too easy...]*

Richard Hamming, *Numerical Methods for Scientists and Engineers, Second Edition*, Dover, 1986. *[handwritten: a little difficult]*

[handwritten: first 2 or 3 weeks here]

[handwritten: online... very modern computer science perspective... very nice intuition...]

[handwritten: never used as a text... quirky...]

[handwritten: Classic texts that haven't been improved on]
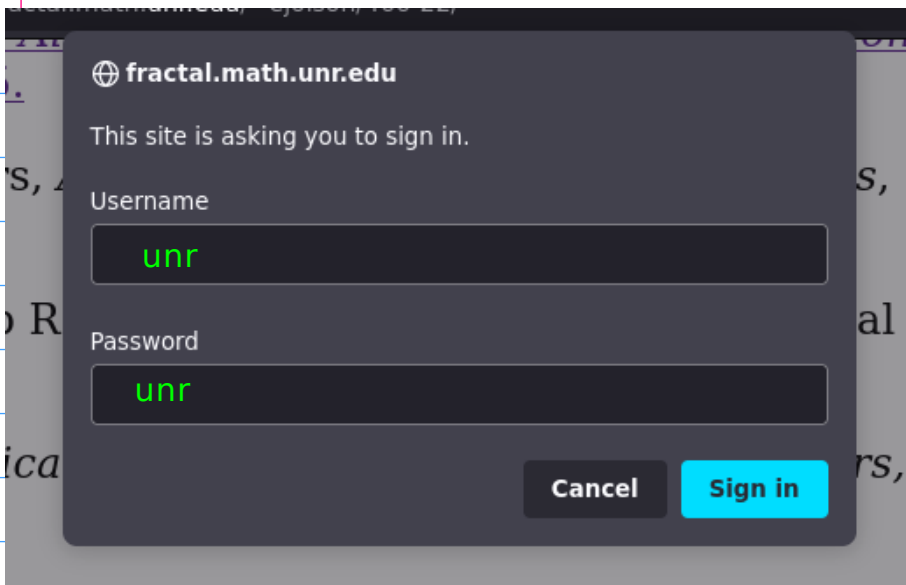
[handwritten: Electronic version through UNR Library.]

[handwritten: Click here...]

Class Handouts

Course materials specific for this section of Math 466 are available by clicking on this link. Details for how to access these files may be found on our course ==page in WebAssign.==

STEP 1

ERRORS 1
Sources of error

Step 2

anything could go wrong...   *rounding error*

4   Mistakes

In the language of Numerical Analysis, a ==mistake (or blunder) is not an error!==
A mistake is due to fallibility (usually human, not machine). Mistakes may be

In numercal method errors are expected and the point is to estimate and know their size.

Science is not only having an answer but knowing how correct that answer is...

Use mathematics to understand the errors in the answers produced by the computer...

GIGO ---> garbage in ... garbage out...

error in .... error out...   but how much?

er;

4 Error generation

Often (for example, in a computer) an operation $\otimes$ is also approximated, by an operation $\otimes^*$, say. Consequently, $x \otimes y$ is represented by $x^* \otimes^* y^*$. Indeed, one has

$$|x \otimes y - x^* \otimes^* y^*| = |(x \otimes y - x^* \otimes y^*) + (x^* \otimes y^* - x^* \otimes^* y^*)|$$
$$\leq |x \otimes y - x^* \otimes y^*| + |x^* \otimes y^* - x^* \otimes^* y^*|$$

so that the accumulated error does not exceed the sum of the propagated and generated errors. Examples may be found in Step 4.

The approximation to addition and multiplication used on a computer are not associate like they should be mathematically... only approximately associative...

For this week let's try to do Steps 1 through 4.

Homework from these steps will be assigned soon...check the website...

Please read the checkpoint questions after each section, because they help reflect on what was just discussed.

Information about Software

read the later ...

- The *Julia 1.6 Lan*guage, official documentation and software download.
- Thomas Breloff, *Plots--Powerful Convenience for Visualization in Julia*. *← ignore for now ...*
- Jupyter Lab, *A Comprehensive List of Links for the Juptyer Project*.

The idea behind Julia is to make a language as easy to use a Matlab or Python but a high performance as C, Fortran.

Julfa is also free! and better ...

Recommend install Julia at home so you don't have to come into the lab to do your homework...

Grading

Computer Labs   50 points
    Theoretical Midterm   75 points   ← first exam pencil and paper...
    Practical Midterm   75 points   ← use the computer... (easier but frightening)
Homework   100 points   50 points
    Final   100 points
----
                                 250 points total

+ written work from the steps I just talked about...   ⎤ -50
+ Computer labs... maybe about 7 or 8.   ⎦ -50
        done in class...

Recall ...

                                    start reading...
                                  &
→ For this week let's try to do Steps 1 through 4.

Homework from these steps will be assigned soon...check the website...

Please read the checkpoint questions after each section, because they
help reflect on what was just discussed.