MAT 305: Mathematical Computing

Introduction to Mathematical Computing

John Perry

University of Southern Mississippi

Spring 2013

Outline

Outline

Description

- **Online:** Introduction to a computer algebra system using calculus-based projects. Students will solve mathematical problems in the MAPLE environment which require an understanding of calculus concepts.

- **Syllabus:** Introduction to a computer algebra system using calculus-based projects. Students will solve mathematical problems in the Sage environment which require an understanding of calculus concepts.

Problem solving or programming?

- This class is about mathematics
  - Problem solving
  - (new!) Introduce ideas of higher mathematics: matrices, modular arithmetic
- Some problems best attacked with a computer
  - Long
  - Experimentation
  - Repetitive/tedious
- Computers require instructions, called **programs**
- We study *some* programming, but class not about programming

Sage?

- Software for Algebra and Geometry Exploration
- Computer Algebra System“started” by William Stein

- Access to other CASs
  - Calculus: Maxima, SymPy, …
  - Linear Algebra: M4RI, Linbox, PARI, …
  - Commutative Algebra: SINGULAR, Macaulay, …
  - Group theory: GAP, …
  - etc.

Why Sage?

- "Free" software
  - "Free as in beer": no cost to you
    - Downloading free
    - Installing free
    - Copying free
    - Bug fixes free
    - Future versions free
  - "Free as in speech":
    - Open-source software
    - No secret algorithms
    - Can study implementation
    - Can correct, improve, contribute

Analogy

Free mathematics

Theorem

There are infinitely many primes.

Proof.

- Consider finite list of primes, $q_1, q_2, \ldots, q_n$.
- Let $p = q_1 q_2 \cdots q_n + 1$.
- **Fact:** since $p \neq 1$, divisible by at least one prime
- By Division Theorem, p not divisible by any q_i (remainder 1, not 0).
- p divisible by unlisted prime q_{n+1}!
- $\therefore$ no finite list, lists all primes.

Analogy

Secret mathematics

Theorem

There are infinitely many primes.

Proof.

"I have discovered a truly marvelous proof of this, which this margin is too narrow to contain."† □

†Real quote, different theorem.

Analogy

Proprietary mathematics

Theorem

There are infinitely many primes.

Proof.

Trade Secret. □

But I prefer Maple!

- Fine, buy your own copy
  - Student discount available
  - I will tell you the Maple equivalents for everything we do in Sage
  - You can submit homework as Maple worksheet
- Be warned:
  - Future versions not free
  - Bug fixes not free
  - I used to use Maple and switched to Sage
  - Recent versions disappointed me
  - After you graduate, pay full price
  - Not always backwards compatible (neither is Sage, but Sage is free)

Outline

Why program?

- Programming bridges gap between humans, computers
  - Computers don't understand human languages
    - Humans intuitive, poetic; computers literal, mechanical
    - Computers only understand **on** or **off**
  - (Most) humans don't understand a computer's native language
    - Mathematics literal and precise, but (most) humans don't understand it, either!
    - Even the humans that do, prefer not to talk to the computer in that language
- Control over computer
- Deeper understanding of computer technology

Kinds of computer languages

- Compiled
  - C/C++
  - FORTRAN
  - Go
- Interpreted or scripting
  - BASIC
  - Python
  - Perl
- Mixed (“bytecode”)
  - C#(.NET)
  - Java

Paradigms of computer languages

- Imperative: BASIC, FORTRAN
- Modular: Modula-2
- Object-oriented: Smalltalk
- Functional: Haskell, ML
- Modern (C++, Python) languages blend

Python

- "Sage" primarily Python
- Python also interface between Sage and user
- Not all *components* of Sage in Python:
  - Maxima: LISP
  - Singular: C/C++
- Python also interface between Sage and user

Advantages of Python

- Modern
  - Facilities for object-oriented, functional programming
- Wide distribution, usage
  - Many employers use it
- Flexible
  - Many good packages enhance it
- Compile for efficiency: Pyrex or Cython

Python ≠ Sage

- Some Python commands don't work in worksheet mode
  - `input()`
- Sage commands do not work in plain Python

Outline

Summary

- This class about mathematical problem solving
- Uses programming via Sage
- No "secret methods" in Sage: can
  - inspect inner workings
  - modify it, fix it, break it…
- Interface is sound programming language in wide use