Mathematics on a UNIX workstation

Math is the King of Science

Skill Level: Intermediate

William B. Zimmerly (bill@zimmerly.com)
Freelance Writer and Knowledge Engineer
Author

31 Oct 2006

Updated 02 Feb 2007

Many tools are available for use on UNIX® that can satisfy your quest for mathematical enlightenment. Some are built in; most are just a download away. This article shows you the way to mathematical nirvana on a UNIX workstation.

When it comes to mathematics, there is no dearth of computer software packages. Math is a popular topic with programmer and analyst types, leading to a huge diversity of tools to choose from.

Mathematics is the *King of Science*. Commercial uses for mathematical workstations are vast: From basic engineering to designer drugs and from gene therapy to celestial navigation, mathematics rules the world. And there is no lack of computer programs to assist people in solving mathematical problems in their chosen field. The remainder of this article describes several commercial and open source systems that might prove valuable in your field of endeavor. All of the systems described have Web pages dedicated to them and are included in the Resources section of the article.

Numerical calculators

The most common mathematical tools found in UNIX workstations take the form of calculators and more advanced spreadsheet programs, such as GNU Calc. The rules of arithmetic are simple, and all these tools follow them faithfully. Almost every full-featured window management package features at least one calculator and sometimes several versions of calculators.

The GNU project also provides calculating tools for download. Among these are GNU Calc, GNU Plot, ATLAS, Choose, and more than 4,000 other math-oriented programs. See the Resources section for a link to the full list.

Symbolic mathematics

As you learned in the mid-level grades in school, math involves a lot more than simple operations on numbers. The symbolic representations of the placeholders of numbers -- variables like x, y, and z; Cartesian coordinates; the principles of factorization; and the world of integral and differential calculus -- are part of a rich and diverse field that demands an equally rich and diverse supply of computer resources. The most common designation for the field of programs designed for these tasks is computer algebraic systems (CAS). Both commercial and open source CAS programs are available, and some of the most commonly used are described below.

But being able to manipulate symbols is not enough in most cases. Graphs can tell us a lot more than the raw listing of coordinated point values ever could. Most systems available for computer algebra also feature a graphing mechanism that lets you visualize the data that is being plotted.

Educational resources

Many computer programs are available to assist you in solving numerical and symbolic mathematical problems, but there are also programs to help you learn about mathematics. The MetaMath family series of programs is an excellent starting point (see the Resources section). For the beginner or for those who do not normally consider themselves to be inclined to the field, mathematics and the programs described in this article can be entertaining as well. Just downloading and testing some of them might inspire you to become more of an expert at something that enriches your life more than any computer game ever could.

Commercial packages

The following is a brief overview of some of the most commonly used commercial CAS systems. If any of them interest you, more information can be found on them by browsing their Web sites, all of which are listed in the Resources section.

Derive. The Software Warehouse is one of the earliest pioneers of creating symbolic mathematical software designed to run on the first workstation computers. Formed in 1979, when the PC revolution was just underway and computer algebraic systems were available only on large mainframe computers using time-sharing terminals, the company designed a program called muMATH so that the users of PCs could go beyond simple calculators and handle symbolic mathematical problems on the limited hardware of small computers. The muMATH system has since been retired and replaced by the far more advanced Derive system. Derive is written in Lisp, a common programming language that is especially suited for the kind of rule-based processing necessary for transforming one abstract mathematical expression into another.

Fermat. This proprietary, shareware computer algebra system is named for one of the most famous mathematicians who ever lived: Pierre de Fermat. It runs on a variety of computer systems and is especially good at arithmetic involving arbitrary-length integers and fractions, graphics, and matrix and polynomial algebra. Fermat is fast. And as the company advertises, if you need to compute the characteristic polynomial of a 400 x 400 matrix over Q, you need Fermat.

Magma. Distributed under a cost-recovery license, Magma is a high-performance system designed to solve algebraic problems. It features functionality for group theory along with group databases, asymptotically fast algorithms for integer and polynomial arithmetic, and several cutting-edge libraries for advanced mathematics.

Maple. Developed and sold by Waterloo Maple Inc. (Maplesoft), the Maple mathematics software package was first created in 1981 by the Symbolic Computation Group at the University of Waterloo in Waterloo, Ontario, Canada. Maple is an impressive-looking package that is especially adept at three-dimensional graphing and at displaying mathematical formulae in textbook form. It combines an interface and a programming language into a dynamic tool for both numerical and symbolic problem solving. Many universities have standardized on Maple for teaching mathematical concepts; the company offers student and professional versions of the software.

MathCAD. PTC, an engineering services company, acquired the rights to MathCAD in 2006. MathCAD, similar to Maple in its simplicity, has an advanced interface that lets engineers enter and display a diversity of formulae along with graphics and text on the screen. MathCAD combines a large library of functional capability including calculus, Laplace transforms, Bessel functions, statistics, and financial functions.

Mathematica. Stephen Wolfram's company, Wolfram Research, developed Mathematica in the late 1980s with a heavy emphasis on the programming aspects of the tool. It offers a flexible combination of both procedural and functional programming and introduces a nondeterministic method for the automatic rewriting of formulae. It's a powerful software package with a large and diverse library of functionality, and it offers a unique multiple-paradigm way for solving problems.

Reduce. Reduce is another general-purpose computer algebra system that was produced in a collaborative way by many scientists. This system was created in the 1960s by Anthony Hearn. Reduce is now a favorite of many in the scientific community. It's distributed for a cost-recovery fee and usually includes the source code with the distribution.

Open source packages and free software

This section explores the world of open source. What follows is a brief overview of some of the most commonly used CAS systems whose source code is licensed under one of the commonly used open source licenses. If any of them interest you, you can find more information by browsing their Web sites, all of which are listed in the Resources section.

Axiom. Useful for exploring diverse mathematical algorithms, the open source computer algebra system known as Axiom is released under a modified Berkeley Software Distribution (BSD) license. It features a strongly typed hierarchy of mathematical objects and commonly used programming data structures for grouping and controlling the objects. Written in the custom A# programming language, Axiom is a worthy tool for those inclined to explore algorithmic designs.

CoCoA. Computations in Commutative Algebra (CoCoA) is another free computer algebra system for working with very large integers, rational numbers, and polynomials. It provides a useful library of mathematical functionality for custom C++ programs.

Dcas. The field of computer algebra offers a rich diversity of methodologies for almost every aspect of mathematical research. There are many ways of representing the objects of mathematics in programs, and Martin Johansen's Dcas system demonstrates just how different these methods can be. Dcas features a method of using identities as rules to manipulate algebraic expressions. Dcas is well worth a look; you might find it to be an ideal way to work on problems in your field.

DoCon. The functional programming language known as Haskell features a program for symbolic mathematics known as DoCon. Per the documentation on the company's Web site (see the Resources section) DoCon implements linear algebra, polynomial gcd, factorization, Grobner bases and supports *constructions on domains* -- Fraction, Polynomial, Residue Ring, and so forth. It's open source and just might be the right tool for an application that you're looking to write.

Eigenmath. Written in C by George Weigt, the Eigenmath program is a simple, easy-to-use computer algebra system. With the source code available, it makes a useful tool for students just beginning to exploring computer algebra systems.

GiNaC. GiNaC, unlike most other computer algebra systems, doesn't provide a graphical user interface (GUI) for the input of expressions, preferring that the user input expressions in native C++ (the implementation language). It accomplishes algebraic syntax by using the native C++ object-oriented technique of operator overloading. It also has one of the stranger names in a field dominated by strange names!

Jscience. A powerful Java™-based library of physics and mathematical functions, the Jscience package is intended to help build a synergy across the domains of science by presenting a single architecture for all development. It's a lofty goal, to say the least, but I wouldn't bet against the company achieving that goal.

Macaulay. The Macaulay computer algebra system is useful for polynomial computations with emphasis on Grobner basis calculations. It was designed for solving problems with a simple syntax and has been described as an *algebraic machine language*.

Mathomatic. This program has no programming capability built in and is intended to be used as a simple symbolic math calculator. It compiles on any system with a C compiler, the standard C libraries, and the UNIX `make` utility.

Maxima. Like most computer algebra systems, Maxima is written in Lisp. Based on the commercial package Macsyma, Maxima includes a complete ALGOL-like programming language and is most useful for teaching programming concepts in computer algebra. It offers arbitrary-precision arithmetic so that integers and rational numbers are limited in size only by the system's available memory.

PARI/GP. PARI is a fast-running C library of symbolic functions for factorization, algebraic number theory, elliptic curves, matrices, and transcendental functions. GP is the interactive shell that provides access to the PARI functions. By using the gp2c compiler, fast-running programs can be created for the problem domains.

SAGE. Software for Algebra and Geometry Experimentation (SAGE) is written in Python and uses the interactive Python shell as its user interface. SAGE is unique in that it can function as an integrator of other diverse computer algebra systems, allowing the user to leverage the individual strengths of the various packages.

SINGULAR. For commutative algebra, algebraic geometry, and singularity theory, the SINGULAR computer algebra system provides a large variety of algorithms in the package kernel as well as shared libraries. It also includes extensive documentation. SINGULAR is a worthwhile system to try out, especially if you're interested in singularity theory.

Yacas. Yet Another Computer Algebra System (Yacas) has a beautiful user interface and all the other strengths of open source software. Input to the system can be in ASCII or OpenMath; the program also has a batch mode too.

Conclusion

Mathematics is the King of Science, and there is no lack of tools to assist you in solving the mathematical problems you face in your particular field. Whether you're an actuary working on pricing insurance policies or a celestial navigator determining the optimal path for flying a spacecraft to Pluto, programs are available to help with your quests for answers.

Resources

Learn

- AIX and UNIX articles: Check out other articles written by William Zimmerly.
- Check out the following math resources:
  - FSF/UNESCO Free Software Directory
  - Axiom home page
  - CoCoA home page
  - the Dcas home page
  - Derive Page
  - DoCon home page
  - Eigenmath homepage
  - Fermat home page
  - Jscience home page
  - Macaulay home page
  - Magma home page
  - Maplesoft
  - PTC
  - Wolfram Research
  - Mathomatic home page
  - Maxima home page
  - MetaMath home page
  - PARI/GP home page
  - Reduce home page
  - SAGE home page
  - SINGULAR Web site
  - Yacas home page
- Search the AIX® and UNIX library by topic:
  - System administration
  - Application development
  - Performance

    - Porting
    - Security
    - Tips
    - Tools and utilities
    - Java™ technology
    - Linux®
    - Open source

- AIX and UNIX: The AIX and UNIX developerWorks zone provides a wealth of information relating to all aspects of AIX systems administration and expanding your UNIX skills.
- New to AIX and UNIX: Visit the New to AIX and UNIX page to learn more about AIX and UNIX.
- AIX 5L™ Wiki: A collaborative environment for technical information related to AIX.
- Safari bookstore: Visit this e-reference library to find specific technical resources.
- developerWorks technical events and webcasts: Stay current with developerWorks technical events and webcasts.
- Podcasts: Tune in and catch up with IBM technical experts.

Get products and technologies

- IBM trial software: Build your next development project with software for download directly from developerWorks.

Discuss

- Participate in the developerWorks blogs and get involved in the developerWorks community.
- Participate in the AIX and UNIX forums:
    - AIX 5L -- technical forum
    - AIX for Developers Forum
    - Cluster Systems Management
    - IBM Support Assistant
    - Performance Tools -- technical
    - Virtualization -- technical
    - More AIX and UNIX forums

About the author

William B. Zimmerly
Bill Zimmerly is a knowledge engineer, a low-level systems programmer with expertise in various versions of UNIX and Microsoft® Windows®, and a free thinker who worships at the altar of Logic. Bill is also known as an unreasonable person. Unreasonable as in, "*Reasonable people adapt themselves to the world. Unreasonable people attempt to adapt the world to themselves. All progress, therefore, depends on unreasonable people*" (George Bernard Shaw). Creating new technologies and writing about them are his passions. He resides in rural Hillsboro, Missouri, where the air is fresh, the views are inspiring, and good wineries are all around. There's nothing quite like writing an article on UNIX shell scripting while sipping on a crystal-clear glass of Stone Hill Blush. You can contact him at bill@zimmerly.com.

Trademarks

IBM, AIX, and AIX 5L are registered trademarks of International Business Machines Corporation in the United States, other countries, or both.
Java and all Java-based trademarks are trademarks of Sun Microsystems, Inc. in the United States, other countries, or both.
Microsoft, Windows, Windows NT, and the Windows logo are trademarks of Microsoft Corporation in the United States, other countries, or both.
UNIX is a registered trademark of The Open Group in the United States and other countries.