Contents

Chapter 1

Overview

1.1 Introduction

NZMATH[8] is a number theory oriented calculation system mainly developed by the Nakamula laboratory at Tokyo Metropolitan University. NZMATH system provides you mathematical, especially number-theoretic computational power. It is freely available and distributed under the BSD license. The most distinctive feature of NZMATH is that it is written entirely using a scripting language called Python.

If you want to learn how to start using NZMATH, see Installation (section 1.1.3) and Tutorial (section 1.1.4).

1.1.1 Philosophy – Advantages over Other Systems

In this section, we discuss philosophy of NZMATH, that is, the advantages of NZMATH compared to other similar systems.

1.1.1.1 Open Source Software

Many computational algebra systems, such as Maple[4], Mathematica[5], and Magma[3] are fare-paying systems. These non-free systems are not distributed with source codes. Then, users cannot modify such systems easily. It narrows these system's potentials for users not to take part in developing them. NZMATH, on the other hand, is an open-source software and the source codes are openly available. Furthermore, NZMATH is distributed under the BSD license. BSD license claims as-is and redistribution or commercial use are permitted provided that these packages retain the copyright notice. NZMATH users can develop it just as they like.

1.1.1.2 Speed of Development

We took over developing of SIMATH[10], which was developed under the leadership of Prof.Zimmer at Saarlandes University in Germany. However, it costs a lot of time and efforts to develop these system. Almost all systems including SIMATH are implemented in C or C++ for execution speed, but we have to take the time to work memory management, construction of an interactive interpreter, preparation for multiple precision package and so on. In this regard, we chose Python which is a modern programming language. Python provides automatic memory management, a sophisticated interpreter and many useful packages. We can concentrate on development of mathematical matters by using Python.

1.1.1.3 Bridging the Gap between Users And Developers

KANT/KASH[2] and PARI/GP[9] are similar systems to NZMATH. But programming languages for modifying these systems are different between users and developers. We think the gap makes evolution speed of these systems slow. On the other hand, NZMATH has been developed with Python for bridging this gap. Python grammar is easy to understand and users can read easily codes written by Python. And NZMATH, which is one of Python libraries, works on very wide platform including UNIX/Linux, Macintosh, Windows, and so forth. Users can modify the programs and feedback to developers with a light heart. So developers can absorb their thinking. Then NZMATH will progress to more flexible user-friendly system.

1.1.1.4 Link with Other Softwares

NZMATH distributed as a Python library enables us to link other Python packages with it. For example, NZMATH can be used with IPython[1], which is a comfortable interactive interpreter. And it can be linked with matplotlib[6], which is a powerful graphic software. Also mpmath[7], which is a module for floating-point operation, can improve efficiency of NZMATH. In fact, the module ecppecpp improves performance with mpmath. There are many softwares implemented in Python. Many of these packages are freely available. Users can use NZMATH with these packages and create an unthinkable powerful system.

1.1.2 Information

NZMATH has more than 25 modules. These modules cover a lot of territory including elementary number theoretic methods, combinatorial theoretic methods, solving equations, primality, factorization, multiplicative number theoretic functions, matrix, vector, polynomial, rational field, finite field, elliptic curve, and so on. NZMATH manual for users is at:

`http://tnt.math.se.tmu.ac.jp/nzmath/manual/`

If you are interested in NZMATH, please visit the official website below to obtain more information about it.

`http://tnt.math.se.tmu.ac.jp/nzmath/`

Note that NZMATH can be used even if users do not have any experience of writing programs in Python.

1.1.3 Installation

In this section, we explain how to install NZMATH. If you use Windows (Windows XP, Windows Vista, Windows 7 etc.) as an operating system (OS), then see 1.1.3.2 "Install for Windows Users".

1.1.3.1 Basic Installation

There are three steps for installation of NZMATH.

First, check whether Python is installed in the computer. Python 2.5 or a higher version is needed for NZMATH. If you do not have a copy of Python, please install it first. Python is available from `http://www.python.org/`.

Second, download a NZMATH package and expand it. It is distributed at official web site:

`http://tnt.math.se.tmu.ac.jp/nzmath/download`

or at sourceforge.net:

`http://sourceforge.net/project/showfiles.php?group_id=171032`

The package can be easily extracted, depending on the operating system. For systems with recent GNU tar, type a single command below:

```
% tar xf NZMATH-*.*.*.tar.gz
```

where, `%` is a command line prompt. With standard tar, type

```
% gzip -cd NZMATH-*.*.*.tar.gz | tar xf -
```

. Please read *.*.* as the version number of which you downloaded the package. For example, if the latest version is 1.0.0, then type the following command.

```
% tar xf NZMATH-1.0.0.tar.gz
```

Then, a subdirectory named NZMATH-*.*.* is created.

Finally, install NZMATH to the standard python path. Usually, this can be translated into writing files somewhere under `/usr/lib` or `/usr/local/lib`. So the appropriate write permission may be required at this step. Typically, type commands below:

```
% cd NZMATH-*.*.*
% su
# python setup.py install
```

1.1.3.2 Installation for Windows Users

We also distribute installation packages for specific platforms. Especially, we started distributing the installer for Windows in 2007.

Please download the installer (NZMATH-*.*.*.win32Install.exe) from

`http://tnt.math.se.tmu.ac.jp/nzmath/download`

or at sourceforge.net:

`http://sourceforge.net/project/showfiles.php?group_id=171032`

Here, we explain a way of installing NZMATH with the installer. First please open the installer. If you use Windows Vista or higher version, UAC (User Account Control) may ask if you run the program. click "Allow". Then the setup window will open. Following the steps in the setup wizard, you can install NZMATH with only three clicks.

1.1.4 Tutorial

In this section, we describe how to use NZMATH.

1.1.4.1 Sample Session

Start your Python interpreter. That is, open your command interpreter such as Terminal for MacOS or bash/csh for linux, type the strings “python” and press the key Enter.

Examples

```
% python
Python 2.6.1 (r261:67515, Jan 14 2009, 10:59:13)
[GCC 4.1.2 20071124 (Red Hat 4.1.2-42)] on linux2
Type "help", "copyright", "credits" or "license" for more information.
>>>
```

For windows users, it normally means opening IDLE (Python GUI), which is a Python software.

Examples

```
Python 2.6.1 (r261:67517, Dec  4 2008, 16:51:00) [MSC v.1500 32 bit (Intel)] on win32
Type "copyright", "credits" or "license()" for more information.

    **********************************************************************
    Personal firewall software may warn about the connection IDLE
    makes to its subprocess using this computer's internal loopback
    interface.  This connection is not visible on any external
```

```
    interface and no data is sent to or received from the Internet.
    ****************************************************************

IDLE 2.6.1
>>>
```

Here, '>>>' is a Python prompt, which means that the system waits you to input commands.

Then, type:

Examples

```
>>> from nzmath import *
>>>
```

This command enables you to use all NZMATH features. If you use only a specific module (the term "module" is explained later), for example, prime, type as the following:

Examples

```
>>> from nzmath import prime
>>>
```

You are ready to use NZMATH. For example, type the string "prime.nextPrime(1000)", then you obtain '1009" as the smallest prime among numbers greater than 1000.

Examples

```
>>> prime.nextPrime(1000)
1009
>>>
```

"prime" is a name of a module, which is a NZMATH file including Python codes. "nextPrime" is a name of a function, which outputs values after the system executes some processes for inputs. NZMATH has various functions for mathematical or algorithmic computations. See **??** Functions.

Also, we can create some mathematical objects. For example, you may use the module "matrix". If you want to define the matrix

$$\begin{pmatrix} 1 & 2 \\ 5 & 6 \end{pmatrix}$$

and compute the square, then type as the following:

Examples

```
>>> A = matrix.Matrix(2, 2, [1, 2]+[5, 6])
>>> print A
1 2
5 6
>>> print A ** 2
11 14
35 46
>>>
```

"Matrix" is a name of a class, which is a template of mathematical objects. See ?? Classes for using NZMATH classes.

The command "print" enables us to represent outputs with good-looking forms. The data structure such as "[a, b, c, $\cdots$]" is called list. Also, we use various Python data structures like tuple "(a, b, c, $\cdots$)", dictionary "$\{x_1 : y_1, x_2 : y_2, x_3 : y_3, \cdots\}$" etc. Note that we do not explain Python's syntax in detail because it is not absolutely necessary to use NZMATH. However, we recommend that you learn Python for developing your potential. Python grammar are easy to study. For information on how to use Python, see `http://docs.python.org` or many other documents about Python.

1.1.5 Note on the Document

† Some beginnings of lines or blocks such as sections or sentences may be marked †. This means these lines or blocks is for advanced users. For example, the class *FiniteFieldElement* (See **FinitePrimeFieldElement**) is one of abstract classes in NZMATH, which can be inherited to new classes similar to the finite field.

[$\cdots$] For example, we may sometimes write as *function(a,b[,c,d])*. It means the argument "c, d" or only "d" can be discarded. Such functions use "default argument values", which is one of the feature of Python.
(See `http://docs.python.org/tutorial/controlflow.html#default-argument-values`)

Warning: Python also have the feature "keyword arguments". We have tried to keep the feature in NZMATH too. However, some functions cannot be used with this feature because these functions are written expecting that arguments are given in order.

Bibliography

[1] IPython. http://ipython.scipy.org/.

[2] KANT/KASH. http://www.math.tu-berlin.de/~kant/kash.html.

[3] Magma. http://magma.maths.usyd.edu.au/magma/.

[4] Maple. http://www.maplesoft.com/.

[5] Mathematica. http://www.wolfram.com/products/mathematica/.

[6] matplotlib. http://matplotlib.sourceforge.net/.

[7] mpmath. http://code.google.com/p/mpmath/.

[8] NZMATH. http://tnt.math.se.tmu.ac.jp/nzmath/.

[9] PARI/GP. http://pari.math.u-bordeaux.fr/.

[10] SIMATH. http://tnt.math.se.tmu.ac.jp/simath/.