

Human Economic Science by Mathematica

Mathematica, its features and interfaces

Symbolic Programming, Use of Notebook and "math" command for
Tsubame computation and others

Hiroyuki AKAMA

Graduate School of Decision
Science and Technology

For the literacy of Macintosh

- We don't specially treat the basic parts of the computer literacy. Instead, I strongly recommend to refer to
 - <http://www.b.cc.titech.ac.jp/~kwakita/Matlab/>
 - <http://www.b.cc.titech.ac.jp/~kwakita/Matlab/?Calendar%2F2009-04-20%2FMacintosh>
 - <http://www.b.cc.titech.ac.jp/~kwakita/Matlab/?Calendar%2F2009-04-20%2FUNIX>
- Use your Tokyo Tech Portal account name and password to login.
- If you choose English at the language setting and relogin, Mathematica will be also switched to the English environment including the help browser in English.

5 minutes for the beginning

Do it yourself



Mathematica

- A highly ingenious, extremely precise and among others user-friendly programming language
- The most suitable for
- 1) complicated symbolic computation including vast amounts of terms,
- 2) analytical or numerical solutions of equations, differential equations or minimization calculation,
- 3) accurate and aesthetic visualization of graphs or figure.

Features of Mathematica

- Symbolic Programming
- General-purpose properties
 - Simple calculations
 - Complicated functions
- Various Interfaces
 - GUI: Notebook
 - Command line: “math” command

```
In[44]:= a = 3; b = 4;  
a + b  
ToString[a] + ToString[b]
```

```
Out[45]= 7
```

```
Out[46]= 3 + 4
```

```
In[36]:= Solve[{2 x - 7 y == 7, 4 x + 2 y == 2}]  
(*Solve a system of equations.*)
```

```
Out[36]= {{x ->  $\frac{7}{8}$ , y ->  $-\left(\frac{3}{4}\right)}$ }
```

```
In[39]:= OurSolve[a1_, b1_, c1_, a2_, b2_, c2_] :=  
Solve[{a1 * x + b1 * y == c1, a2 * x + b2 * y == c2}];
```

```
OurSolve[2, -7, 7, 4, 2, 2]
```

```
Out[40]= {{x ->  $\frac{7}{8}$ , y ->  $-\left(\frac{3}{4}\right)}$ }
```

Symbolic Programming

- We can manipulate intuitive representations as operators.

```
In[41]:= Symbol["ihes"];  
         Symbol["ihessc"];  
         Symbol["internationalHumanEconomicScience"];  
  
In[44]:= Names["ih*"]  
  
Out[44]= {ihes, ihessc}  
  
In[45]:= Names["iehs", SpellingCorrection -> True]  
  
Out[45]= {ihes}
```

- Mathematica allows us to create a symbol with a name.
- Mathematica commands are evaluated by **pressing Shift-Return**
- Pressing the single Return key is for starting a new line.

Symbolic Programming

- Mathematica allows us to put in the arguments of a command not only symbols or values but even the other commands to make a nested structure which instinctively duplicates our thought processes
- by multiple parenthesis
- or
- a succession of the postposing operator “//”.

```
In[87]:= a = Range[ 6]  
        b = Mean[ a]  
        c = N[ b]
```

```
Out[87]= {1, 2, 3, 4, 5, 6}
```

```
Out[88]=  $\frac{7}{2}$ 
```

```
Out[89]= 3.5
```

```
In[91]:= N[ Mean[ Range[ 6] ] ]  
        Range[ 6] // Mean // N
```

```
Out[91]= 3.5
```

```
Out[92]= 3.5
```

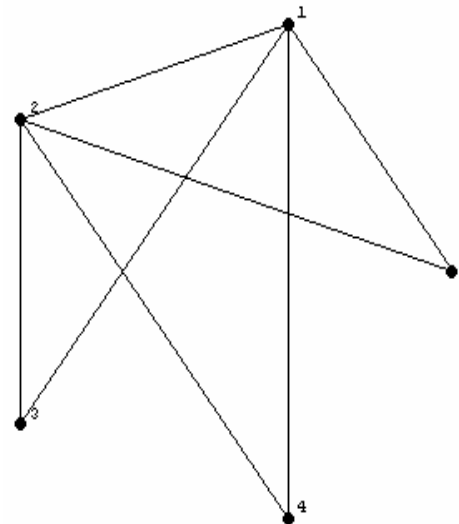
Symbolic Programming

- We can automatically draw a network by building a graph object from a sequence of number pairs.
- In this case, a list, a matrix, a combinatorial object and a picture are all treated seamlessly.

```
pairssample = {{1, 2}, {1, 3}, {1, 4}, {1, 5}, {2, 3}, {2, 4}, {2, 5}};  
adjsample = ToAdjacencyMatrix[FromUnorderedPairs[pairssample]];  
adjsample // MatrixForm
```

$$\begin{pmatrix} 0 & 1 & 1 & 1 & 1 \\ 1 & 0 & 1 & 1 & 1 \\ 1 & 1 & 0 & 0 & 0 \\ 1 & 1 & 0 & 0 & 0 \\ 1 & 1 & 0 & 0 & 0 \end{pmatrix}$$

```
ShowGraph[FromAdjacencyMatrix[adjsample], VertexLabel -> {1, 2, 3, 4, 5}]
```



- Graphics -

Use of Notebook

- Interactive
- Test each function separately to study its behavior
- Symbolic, numerical, and graphical computation system

```
In[56]:= a = Symbol["dst"];  
b = Symbol["gs"];
```

```
In[63]:= ?ToString
```

ToString[expr]は、OutputFormでの
exprの表示形に対応した文字列を与える。
ToString[expr, form]は、指定された形式
formでの出力に対応する文字列を与える。 [詳細](#)

```
In[64]:= ?<>
```

"s1" <> "s2" <> ... , StringJoin["s1", "s2", ...]
 , または StringJoin[{"s1", "s2", ... }]
 は、siを鎖状につなぎ合せて文字列を作成する。 [詳細](#)

```
In[65]:= ToString[a] <> ToString[b]
```

```
Out[65]= dstgs
```

We use " ? " to display basic information about the command.
" ? " objects gives information on the Mathematica object.

Use of Notebook

- The semicolon “;” at the end of an expression suppresses the display of the result.
- Mathematica is not an object-oriented language.
 - Two objects sharing an identical value are considered as same, contrary to Java, for instance.

```
In[66]:= s1 = Symbol["ihes"]  
         s2 = Symbol["ihes"];  
         s1 == s2
```

```
Out[66]= ihes
```

```
Out[68]= True
```

```
For object-oriented language,..  
Symbol s1 = new Symbol("ihes")  
Symbol s2 = new Symbol("ihes")  
s1==s2  
False
```

Use of Notebook

- Try the **N** command for floating-point approximation.
 - The **N** command for numerical approximation is necessary to get the approximate size of a value.
 - The second argument corresponds to the "digit precision", we can leave out by default.
 - Pi:circle ratio
- Try **N[Pi]** without the second argument.

```
N[Pi, 100]
```

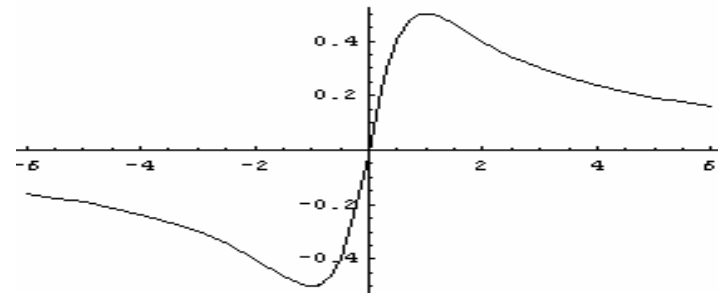
```
3.1415926535897932384626433832795028841971693993751058\ <  
20974944592307816406286208998628034825342117068
```

- Mathematica returns the output as an exact evaluation, so the command `Sqrt[2]` produce only its echo.
- Does “`TrueQ[Pi == 3.1415926535897]`” return true ?
- Does “`TrueQ[Pi == 3.14159265358979]`” return true?

Use of Notebook

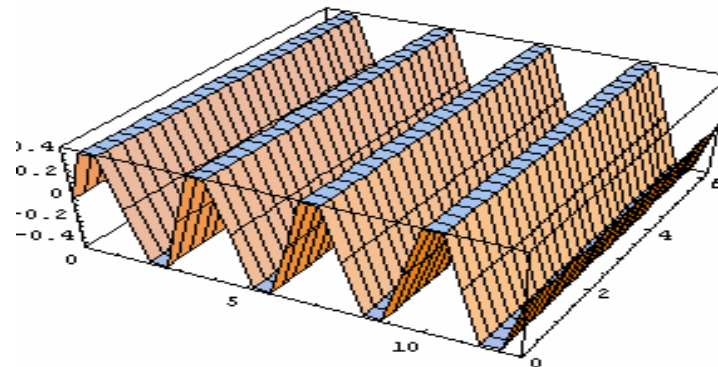
- With the “Plot” and the other derived commands, you can depict beautiful and precise pictures of graph on the notebook.

```
Plot[x/(x^2 + 1), {x, -6, 6}]  
(*Graph on the interval [-6,6]*)
```



-Graphics-

```
Plot3D[Sin[x] Cos[x], {x, 0, 4 Pi}, {y, 0, 2 Pi}]  
(*Graph a function of two variables in three  
dimensions.*)
```



-SurfaceGraphics-

Use of Notebook

Liabile to be wrong (1)

- The rational number is not calculated and treated as it is. If you want to treat as a real number, you must approximate it by using N.
- You must always pay attention to the current value of a variable to avoid superficial errors. Use the function “Clear” to clear values or definitions of a symbol.

```
In[42]=  $\frac{1}{2} + \frac{1}{3}$ 
Out[42]=  $\frac{5}{6}$ 

In[43]=  $\frac{1}{2} + \frac{1}{3}$ 
Out[43]=  $\frac{5}{6}$ 

In[44]= N[1/2 + 1/3]
Out[44]= 0.833333

In[45]= r = % // N
(*Same as the precedent command.*)
Out[45]= 0.833333

In[46]= MyFunction[x1_, x2_] := If[x1 > r, Print["r1 is bigger than r2"],
Print["r1 is smaller or equal to r"]];

In[47]= MyFunction[0.84, 0.85]
r1 is bigger than r2

(*Funny! Is 0.84=r1 bigger than to 0.85=r2?*)
(*The reason is that the value of the variable r=
5/6 remains in the function and caused a careless misprint.*)
(*You must then use the function "Clear" to avoid such a
careless miss.*)

In[48]=
r
Clear[r];
r

Out[48]= 0.833333

Out[50]= r
```

Palette

- The combinations of the ready-made graphical icons prepared in the palette (select the menu of “BasicMathInput ”) and the insertion of values into the blank squares of them lead you to easily construct the most complicated mathematical expressions.

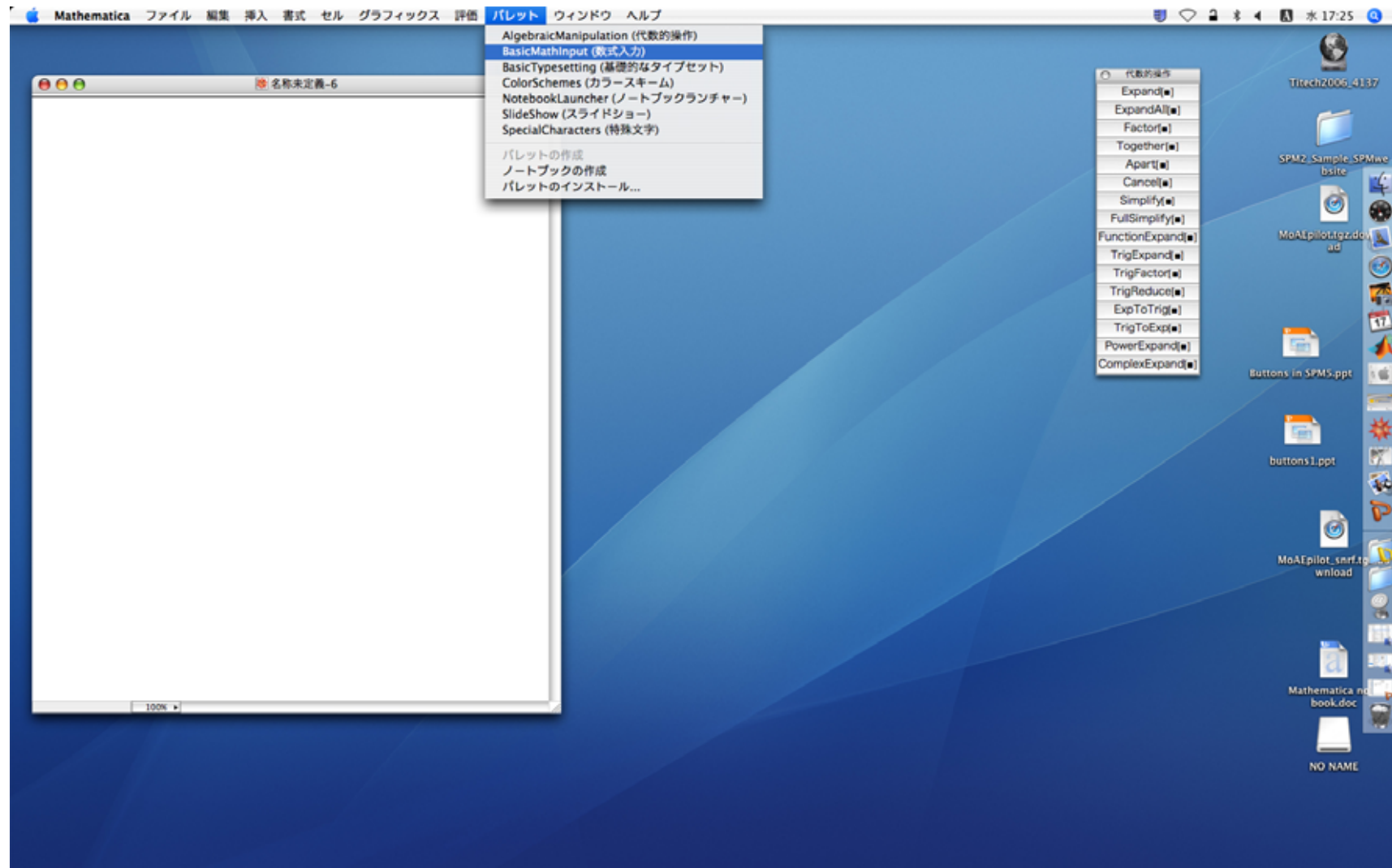
$$\int_{-1}^1 \frac{\sqrt{x}}{x} dx$$

$$2 - 2 I$$

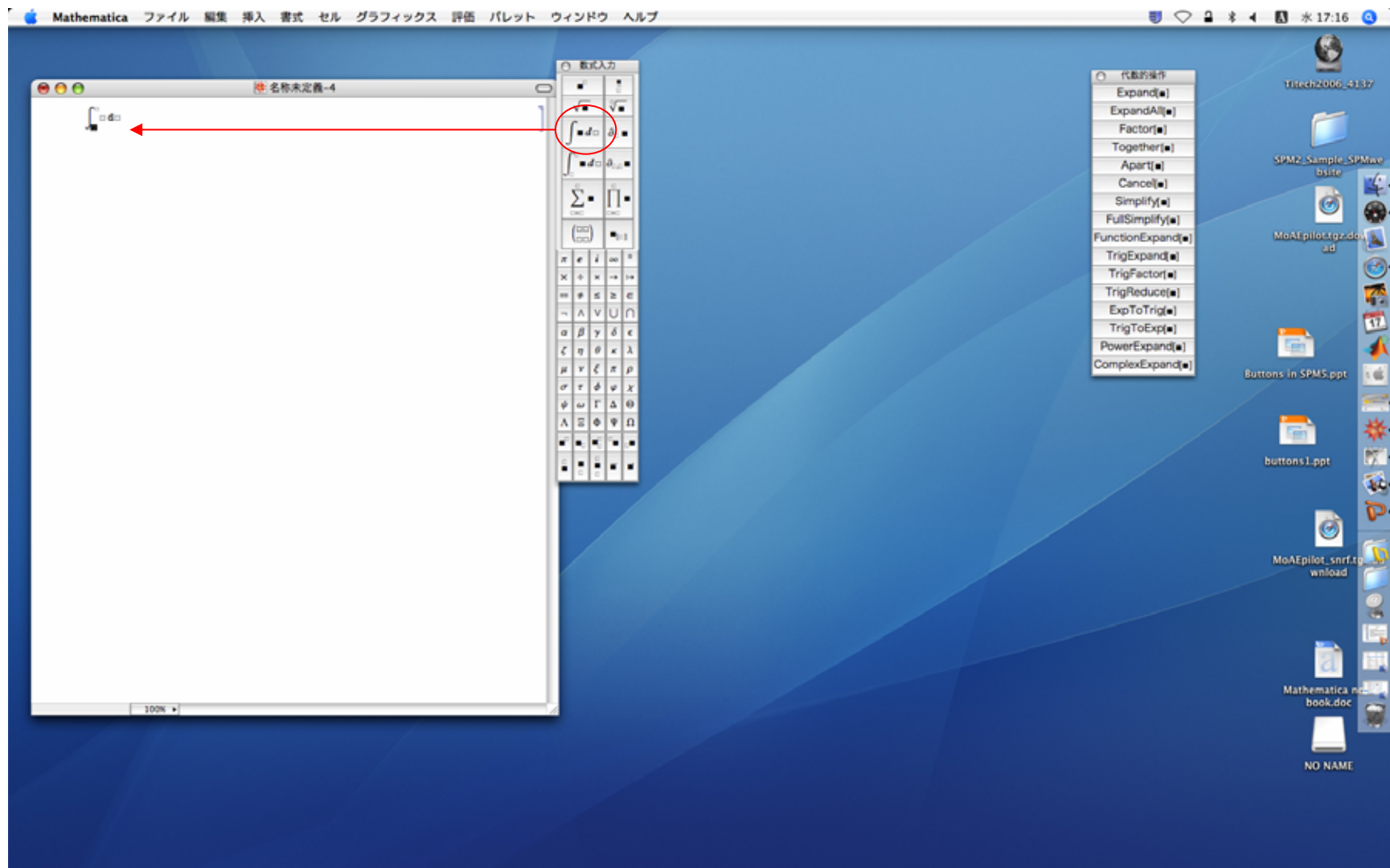
$$\sum_{i=1}^3 i (i + 1)$$

$$20$$

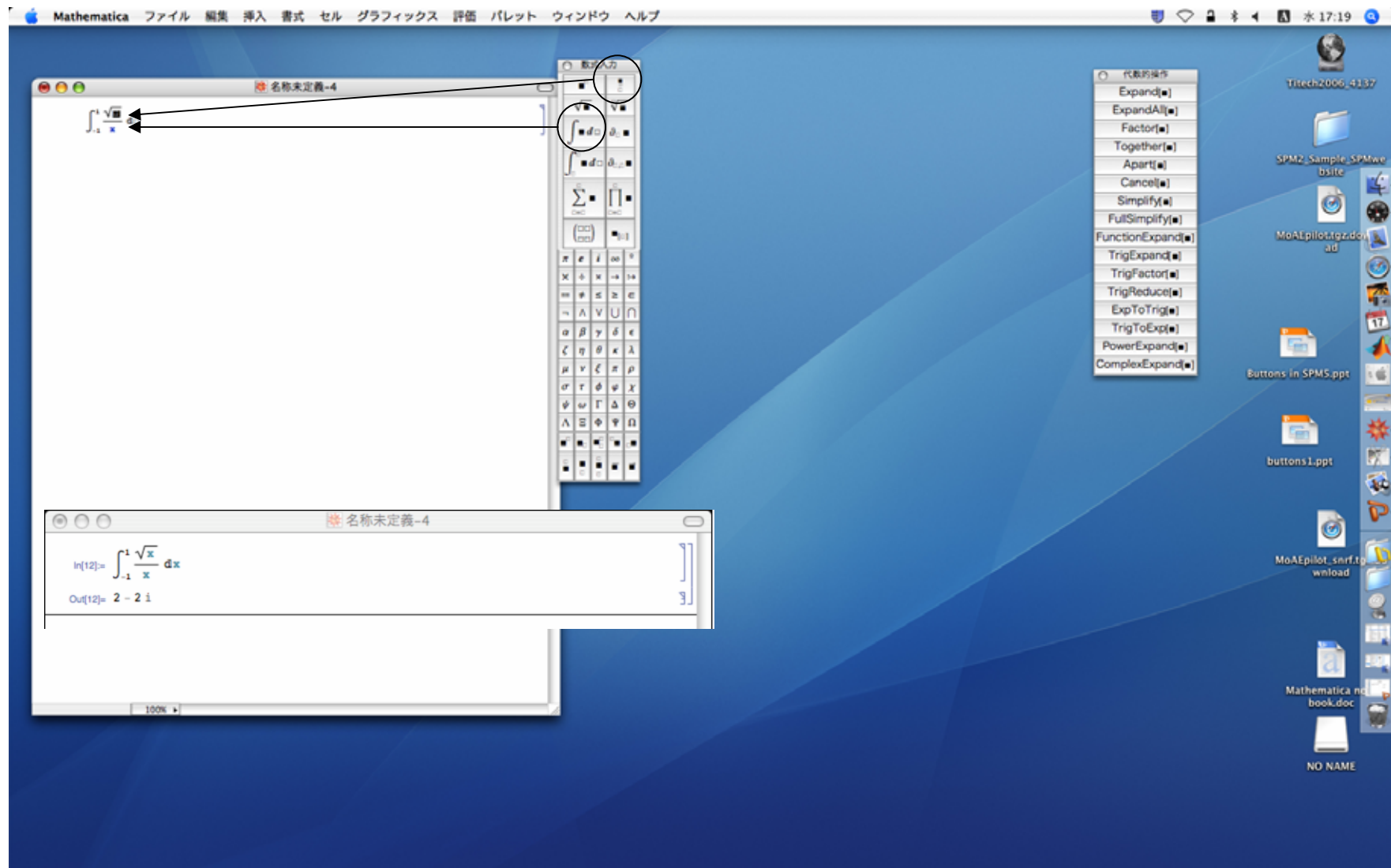
Palette(1)



Palette(2)



Palette(3)



Various ways of accessing Mathematica

- Various ways available at Tokyo Tech to use Mathematica
- 1)Your desktop (free of charge thanks to the site license at Tokyo Tech)
 - Notebook to test a program by trial and error
 - The “Palette” for easily inputting mathematical expressions is very useful.
- 2)GSIC Educational System
 - Now on iMac, mainly used in this course
- 3)Tsubame
 - Grid Cluster, where the math command for a large scaled computation is available.
 - The “math” command is applied to a script in which a series of procedures is written in order.
 - But “Palette” and the symbols other than ASCII characters cannot be used in the text file submitted in background process.

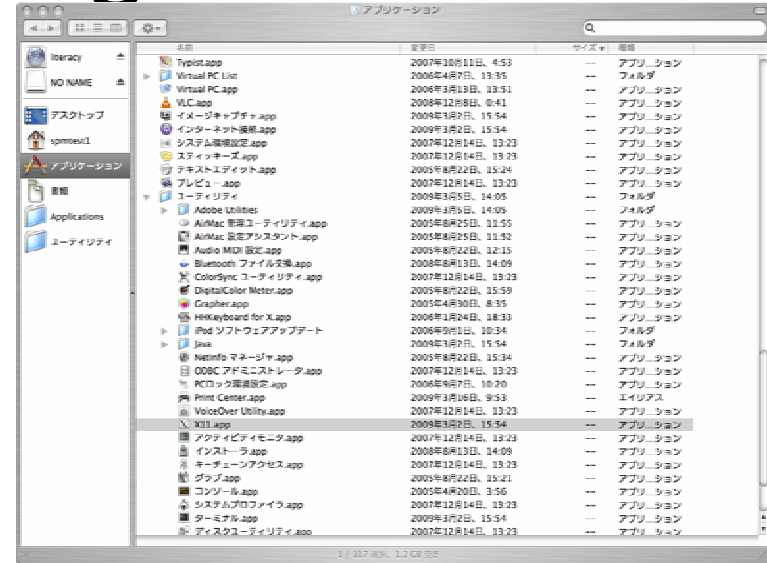
Mathematica on Tsubame

- Needs Unix literacy
- You must apply for another account with different login name and password. The initial password must be changed by each user with the “passwd” command.
 - <http://www.gsic.titech.ac.jp/~ccwww/>
 - http://www.gsic.titech.ac.jp/~ccwww/faq_e.html
- Remote login is possible from the educational Computer System by using ssh.
- There are two ways of use: in foreground with “Mathematica” command as well as in background with “math” command

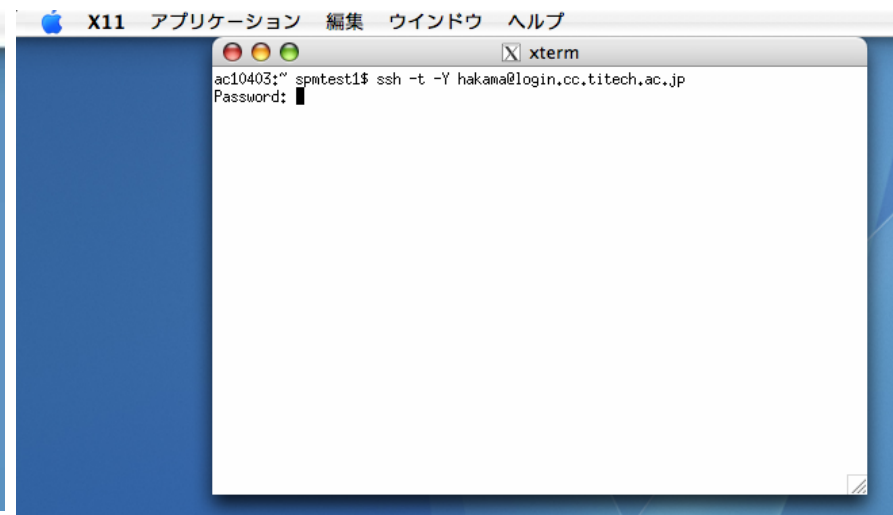
Remote login

1

1. Choose “X11.aps” from “Applications”
2. Select “terminal” from the pull down menu of “Application” on the tool bar of “X11.aps”.
3. Execute the following command
“ssh -t -Y <username>@login.cc.titech.ac.jp
(Type your Tsubame username.)
4. Type your Tsubame password and enter.



3



The “Mathematica” Command

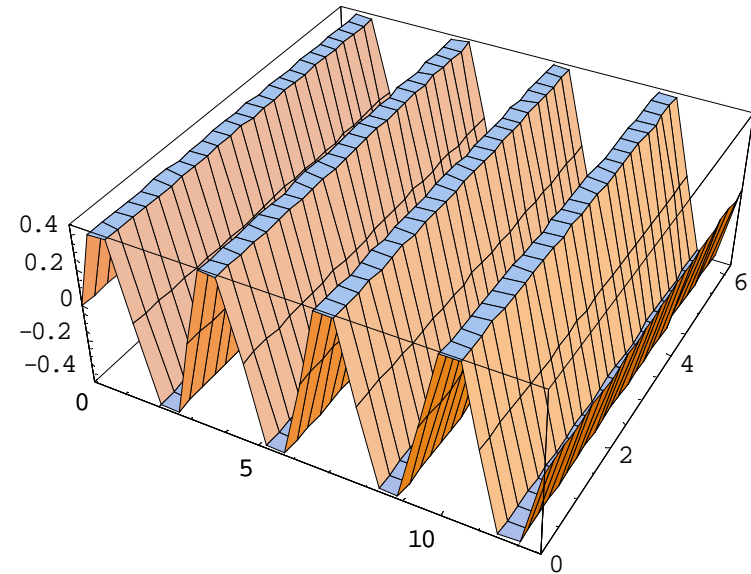
- The “Mathematica” as a command allows us to use GUI environment in foreground.
- You have only to type **n1ge -fore Mathematica** and press the enter key.

```
hakama@tgg075022:/home4/usr8/hakama
aci0403:~$ ssh -t -Y hakama@login.cc.titech.ac.jp
Password:
Warning: No xauth data; using fake authentication data for X11 forwarding.
Last login: Thu Apr  9 17:46:25 2009 from aci0403.b.cc.titech.ac.jp
quota check result.
FileSystem      MaxSize(GB)    Used(GB)
-----
/home           300.000        25.232
/home           0.000          0.186
Forwarding to NICE Interactive Queue.....
hakama@tgg075022:/home4/usr8/hakama> n1ge -fore Mathematica[]
```



The “math” command

- Useful for using Mathematica on Tsubame
 - %**cat plot3d.txt**
 - `pict = Plot3D[Sin[x]Cos[x], {x, 0, 4Pi}, {y, 0, 2Pi}]`
 - (*Graph a function of two variables in three dimensions.*)
 - `Export["3d.gif", pict, "GIF", ImageResolution -> Automatic, ImageSize -> {850,850}]`
 - %**n1ge math <plot3d.txt>log-plot.3d.txt**



A picture file is generated even without GUI.
It can be imported with the Import command.
`%Import["3d.gif", "GIF"]`

N1GE is adopted in the TSUBAME batch system as a job management system (a queue system). To know the details, use “`n1ge -help`”

The “math” command

- Foreground or background
- The command **Mathematica** can only be executed in foreground because it is a GUI interface command.
- The **math** can either be executed in interactive mode in foreground or be executed in program mode in background.

Basic Grammar(1)

- **Parenthesis** () are used for grouping expressions just as in the ordinary mathematics
 - $x+(1-y)/2$
- **Braces** {} define list on which records are separated by commas.
 - {x,y,z}
 - {1,2,3}
- **Square brackets** [] can be used only for a function to put arguments or their values in position.
 - Sin[x^2]
 - N[Pi]
- **Attentions:** double square brackets [[]] means, then equal to the function Part[
 - In[1]:A={3,4,5,1,9};
 - In[2]:A[[4]]
 - Out[1]:1
 - In[3]:Part[A,4]
 - Out[2]:1

Basic Grammar (2)

- The symbol of a single equality sign = means the assignment of a value in a variable.
- The symbol of double equal signs == means that the value of the right side is equal to that of the left side.

```
In[1]:= {x, y, z} = {1, 2, 3};
```

```
x
```

```
Out[2]= 1
```

s of
n

```
In[3]:= y
```

```
Out[3]= 2
```

```
In[4]:= z
```

```
Out[4]= 3
```

```
In[5]:= {x, y, z} == {1, 2, 3}
```

```
Out[5]= True
```

Basic Grammar(3)

- The symbol of `:=` is used for defining the function you make.
- It is highly recommended to name your function all in small letters, because there are a bunch of built-in functions whose names begin with a capital letter and that cannot be available for your own implementation.
- Each of the arguments in the definition of a function must end with an underbar `_`.

```
In[6]:= myfunction[arg1_, arg2_] := If[arg1 > arg2, arg1^2, -arg2];
```

```
In[7]:= myfunction[6, 2]
```

```
Out[7]:= 36
```

```
In[8]:= myfunction[2, 6]
```

```
Out[8]:= -6
```

If[]

The syntax is

If[predicate, true , false]

First the predicate (logical expression) is tested.

If it evaluates to true, the value returned is equal to the value of the expression true.

If it is evaluated to false, the value returned is the value of the false.

Basic Grammar (4)

- Comment mark: (* *)
- The system of Mathematica discards what is inside this mark. You will put some words of explanation for the better understanding of your program.

```
In[21]:= matrix1 =  $\begin{pmatrix} 1 & 2 \\ 3 & 4 \end{pmatrix}$ ;
```

```
matrix1 = {{1, 2}, {3, 4}}
```

```
(*A matrix can be given by nested lists or a graphical  
expression made with the palette. *)
```

```
Out[22]= True
```

Books to be recommended

- Roman E. Maeder, Computer Science with Mathematica, Theory and Practice for Science, mathematics, and Engineering, Cambridge University Press, 2000 (Elementary book)
- Roman E. Maeder, The Mathematical Programmer II, Academic Press, 1996 (Highly advanced)
- Colin Rose, Murray D. Smith, Mathematical Statistics with Mathematica, Springer Texts in Statistics, 2002
- Martha L. Abell, James P. Braselton, John A. Rafter, Statistics with Mathematica, Academic Press, 1999

Exercise(1)

- 1. Execute the command `N[E]`. What does this mean?
- 2. Even though `N[E]` gives 2.71828 as an output, `N[E]==2.71828` returns false. How come?
- 3. Double quotation marks denote word strings as
 - `word1 = "Tokyo";word2 = "Institute";word3 = "of";word4="Technology";`
- 4. What do you have to do for combining these 4 variables to output "Tokyo Institute of Technology"?
- 5. Line up all the integers from 1 through 1000 into a list. And calculate the mean value of the elements of this list.

Exercise(2)

- 5.Examine the function “Table”. And explain what will be given by executing the commands
 - `Table[Random[Integer],{i, 1, 10}]`
 - `Table[Random[Real],{i, 1, 10}]`
- 6.Rewrite the function “Range” by using “Table.”
- 7.Solve this simultaneous equation by Mathematica.
 $2x+y=3, 3x-y=-4$
 - What do we must do to get the solution of this equation by real numbers (not by fractional numbers)?
- 8.Solve this equation of second degree by using Mathematica. $2x^2+3x-2=0$
- 9.Calculate this integration by using the palette.

$$\int_1^2 (x^2 + 3x - 4) dx$$

Exercise(3)

- 10. Make by yourself a series of mathematical formulas using the palette and calculate them on your notebook.
- And save the contents you have made in your home directory (and your removable media) in multiple file formats.