1
Getting Started

1.1 Launching *Mathematica*

The first task you will face is finding where *Mathematica* resides in your computer's file system. If this is the first time you are using a computer in a classroom or lab, by all means ask your instructor for help. You are looking for "Spikey," an icon that looks something like this:

When you have located the icon, double click it with your mouse. In a moment an empty window will appear. This is your *Mathematica notebook*; it is the environment where you will carry out your work.

The remainder of this chapter is a quick tutorial that will enable you to get accustomed to the syntax and conventions of *Mathematica*, and demonstrate some of its many features.

1.2 The Basic Technique for Using *Mathematica*

A *Mathematica* notebook is an interactive environment. You type a command (such as $2+2$) and instruct *Mathematica* to execute it. *Mathematica* responds with the answer on the next line. You then type another command, and so on. Each command you type will appear on the screen in a **boldface** font. *Mathematica*'s output will appear in a plain font.

> **Entering Input**
>
> After typing a command, you *enter* it as follows:
>
> - On a machine running Windows: Hit the combination SHIFT-ENTER, or hit the ENTER key on the numeric keypad if you have one (usually in the lower right portion of the keyboard).
> - On a Mac: Hit the ENTER key (usually in the lower right portion of the keyboard), or hit the combination SHIFT-RET.

1.3 The First Computation

For your first computation, type

```
2 + 2
```

then hit the SHIFT-ENTER combination (Windows) or the ENTER key (Mac OS). There may be a brief pause while your first entry is processed. During this pause the notebook's title bar will contain the text "Running..."

```
In[1]:= 2 + 2
Out[1]= 4
```

The reason that this simple task takes a moment is that *Mathematica* doesn't start its engine, so to speak, until the first computation is entered. In fact, entering the first computation causes your computer to launch a second program called the MathKernel (or kernel for short). *Mathematica* really consists of these two programs, the Front End, where you type your commands and where output, graphics, and text are displayed, and the MathKernel, where calculations are executed. Every subsequent computation will be faster, for the kernel is now already up and running.

1.4 Commands for Basic Arithmetic

Mathematica works much like a calculator for basic arithmetic. Just use the +, –, *, and / keys on the keyboard for addition, subtraction, multiplication, and division. As an alternative to typing *, you can multiply two numbers by leaving a space between them (the × symbol will automatically be inserted when you leave a space between two numbers). You can raise a number to a power using the ^ key. Use the dot (i.e., the period) to type a decimal point. Here are a few examples:

```
In[1]:= 17 + 1
Out[1]= 18

In[2]:= 17 – 1
Out[2]= 16

In[3]:= 123 456 789 * 123 456 789
Out[3]= 15 241 578 750 190 521

In[4]:= 123 456 789 × 123 456 789
Out[4]= 15 241 578 750 190 521

In[5]:= 123 456 789 ^ 2
Out[5]= 15 241 578 750 190 521
```

```
In[6]:= 9.1 / 256.127
Out[6]= 0.0355292

In[7]:= 34 / 4
Out[7]= 17/2
```

This last line may seem strange at first. What you are witnessing is *Mathematica*'s propensity for providing exact answers. *Mathematica* treats decimal numbers as approximations, and will generally avoid them in the output if they are not present in the input. When *Mathematica* returns an expression with no decimals, you are assured that the answer is exact. Fractions are displayed in lowest terms.

1.5 Input and Output

You've surely noticed that *Mathematica* is keeping close tabs on your work. Each time you enter an expression, *Mathematica* gives it a name such as In[1]:=, In[2]:=, In[3]:=. The corresponding output comes with the labels Out[1]=, Out[2]=, Out[3]=, and so on. At this point, it is enough to observe that these labels will appear all by themselves each time you enter a command, and it's okay:

```
In[1]:= (1/2)^6
Out[1]= 1/64
```

You've surely noticed something else too (you'll need to be running a live session for this), those brackets along the right margin of your notebook window. Each input and output is written into a *cell*, whose scope is shown by the nearest bracket directly across from the respective input or output text. Cells containing input are called *input cells*. Cells containing output are called *output cells*. The brackets delimiting cells are called *cell brackets*. Each input–output pair is in turn grouped with a larger bracket immediately to the right of the cell brackets. These brackets may in turn be grouped together by a larger bracket, and so on. These extra brackets are called *grouping brackets*.

At this point, it's really enough just to know these brackets are there and to make the distinction between the innermost (or smallest, or leftmost) brackets which delimit individual cells and the others which are used for grouping. If you are curious about what good can possibly come of them, try positioning the tip of your cursor arrow anywhere on a grouping bracket and double click. You will *close the group* determined by that bracket. In the case of the bracket delimiting an input–output pair, this will have the effect of hiding the output completely (handy if the output runs over several pages). Double click again to open the group. This feature is useful when you have created a long, complex document and need a means of managing it. Alternately, you can double click on any

output cell bracket to *reverse-close* the group. This has the effect of hiding the input code and displaying only the output.

Since brackets are really only useful in a live *Mathematica* session, they will not, by default, show when you print a notebook. Further details about brackets and cells will be provided in Section 2.2 on page 27.

One last bit of terminology is in order. When you hit the SHIFT-ENTER combination (Windows), or the ENTER key (Mac OS) after typing an input cell, you are *entering the cell*. You'll be seeing this phrase quite a bit in the future.

1.6 The BasicMathInput Palette

There may already be a narrow, light gray window full of mathematical symbols along the side of your screen. If so, you are looking at one of *Mathematica*'s palettes, and chances are that it is the BasicMathInput palette:

The BasicMathInput palette

If you see no such window, go to the Palettes menu and select BasicMathInput to open it.

The BasicMathInput palette is indispensable. You will use it to help typeset your *Mathematica* input, creating expressions that cannot be produced in an ordinary one-dimensional typing environment. Palettes such as this provide you with a means of producing what the designers of *Mathematica* call *two-dimensional* input, which often matches traditional mathematical notation. For instance, use the $\square^{\square}$ button in the upper left corner of the palette to type an exponential expression such as 17^{19}. To do this, first type **17** into your *Mathematica* notebook, then highlight it with your mouse. Next, push the $\square^{\square}$ palette button with your mouse. The exponent structure shown on that button will be pasted into your notebook, with the 17 in the position of the black square on the palette button (the black square is called the *selection placeholder*). The text insertion point will move to the placeholder in the exponent position. Your input cell will look like this:

$$17^{\blacksquare}$$

You can now type the value of the exponent, in this case 19, into the placeholder, then enter the cell:

```
In[1]:= 17^19
Out[1]= 239 072 435 685 151 324 847 153
```

⚠ Another way to accomplish the same thing is this: First hit the palette button, then type 17 into the first placeholder. Next hit the TAB key to move to the second placeholder (in the exponent position). Now type 19 and enter the cell. This procedure is perhaps a bit more intuitive, but it can occasionally get you into trouble if you are not careful with grouping. For instance, if you want to enter $(1+x)^8$, and the first thing you do is push the $\square^{\square}$ button on the palette, then you must type **(1+x)** with parentheses, then TAB, then 8. By contrast, you could type **1+x** with or without parentheses and highlight the expression with your mouse, then hit the $\square^{\square}$ palette button, and then type 8. The parentheses are added automatically, if needed, when this procedure is followed.

If you don't understand what some of the palette buttons do, don't fret. Just stick with the ones that you know for now. For instance, you can take a cube root like this: type a number and highlight it with the mouse, then push the $\sqrt{\square}$ button on the BasicMathInput palette, then hit the TAB key, and finally type 3. Now enter the cell:

```
In[2]:= ∛50 653
Out[2]= 37
```

This is equivalent to raising 50653 to the power 1/3:

```
In[3]:= 50 653^(1/3)
Out[3]= 37
```

And of course we can easily check the answer to either calculation:

```
In[4]:= 37^3
Out[4]= 50 653
```

> **Entering Input**
>
> Speaking in general terms, the buttons on the top portion of the BasicMathInput palette (in fact all buttons containing a solid black placeholder ■ on this and any other palette) are used this way:
>
> - Type an expression into a *Mathematica* notebook.
> - Highlight all or part of the expression with your mouse (by dragging across the expression).
> - Push a palette button. The structure on the face of the button is pasted into your notebook, with the highlighted text appearing in the position of the solid black square.
> - If there are more placeholders in the structure, use the TAB key or forward arrow (or move the cursor with your mouse) to move from one to the next.

The buttons on the middle portion of the BasicMathInput palette have no placeholders. They are used simply to paste into your notebook characters that are not usually found on keyboards. To use them, simply position the cursor at the point in the notebook where you want the character to appear, then push a palette button.

For instance, the ≤ symbol can be used to test if one number is less than or equal to another:

```
In[5]:= Sqrt[50 653] ≤ 225
Out[5]= False

In[6]:= Sqrt[50 653] ≤ 226
Out[6]= True
```

The special symbol == is used to test if one quantity is equal to another. It has the same meaning as the equal sign in standard mathematical notation:

```
In[7]:= Sqrt[50 653] == 50 653^(1/2)
Out[7]= True
```

1.7 Decimal In, Decimal Out

Sometimes you don't want exact answers. Sometimes you want decimals. For instance how big is this number? It's hard to get a grasp of its magnitude when it's expressed as a fraction:

```
In[1]:= 17^19 / 19^17
Out[1]= 239 072 435 685 151 324 847 153 / 5 480 386 857 784 802 185 939
```

And what about this?

```
In[2]:= ∛59 875
Out[2]= 5 479^(1/3)
```

Mathematica tells us that the answer is 5 times the cube root of 479 (remember that a space indicates multiplication, and raising a number to the power 1/3 is the same as taking its cube root). The output is exact, but again it is difficult to grasp the magnitude of this number. How can we get a nice decimal approximation, like a calculator would produce?

If any one of the numbers you input is in decimal form, *Mathematica* regards it as approximate. It responds by providing an approximate answer, that is, a decimal answer. It is handy to remember this:

```
In[3]:= 17.0^19 / 19^17
Out[3]= 43.6233

In[4]:= ∛59 875.0
Out[4]= 39.1215
```

A quicker way to accomplish this is to type a decimal point after a number with nothing after it. That is, *Mathematica* regards "17.0" and "17." as the same quantity. This is important for understanding *Mathematica*'s output:

```
In[5]:= ∛59 875.
Out[5]= 39.1215

In[6]:= 30. / 2
Out[6]= 15.
```

Note the decimal point in the output. Since the input was only "approximate," so too is the output. Get in the habit of using exact or decimal numbers in your input according to the type of answer, exact or approximate, that you wish to obtain. Adding a decimal point to any single number in your

input will cause *Mathematica* to provide an approximate (i.e., decimal) output. A detailed discussion on approximate numbers can be found in Section 8.3 on page 392.

1.8 Use Parentheses to Group Terms

Use ordinary parentheses () to group terms. This is *very* important, especially with division, multiplication, and exponentiation. Being a computer program, *Mathematica* takes what you say quite literally; tasks are performed in a definite order, and you need to make sure that it is the order you intend. Get in the habit of making a mental check for appropriate parentheses before entering each command. Here are some examples. Can you see what *Mathematica* does in the absence of parentheses?

```
In[1]:= 3 * (4 + 1)
Out[1]= 15

In[2]:= 3 * 4 + 1
Out[2]= 13

In[3]:= (-3)^2
Out[3]= 9

In[4]:= -3^2
Out[4]= -9

In[5]:= (3 + 1) / 2
Out[5]= 2

In[6]:= 3 + 1 / 2
Out[6]= 7/2
```

The last pair of examples above shows one benefit of using the BasicMathInput palette instead of typing from the keyboard. With the two-dimensional typesetting capability afforded by the palette there is no need for grouping parentheses, and no chance for ambiguity:

```
In[7]:= (3 + 1)/2
Out[7]= 2
```

In[8]:= $3 + \frac{1}{2}$

Out[8]= $\frac{7}{2}$

The lesson here is that the order in which *Mathematica* performs operations in the absence of parentheses may not be what you intend. When in doubt, add parentheses. Also note: you do not need to leave a space to multiply by an expression enclosed in parentheses:

In[9]:= **25 (2 + 2)**

Out[9]= 100

Note also that only round brackets can be used for the purpose of grouping terms. *Mathematica* reserves different meanings for square brackets and curly brackets, so never use them to group terms.

1.9 Three Well-Known Constants

Mathematica has several built-in constants. The three most commonly used are π, the ratio of the circumference to the diameter of a circle (approximately 3.14); e, the base of the natural logarithm (approximately 2.72); and i, the imaginary number whose square is -1. You can find each of these constants on the BasicMathInput palette.

In[1]:= π

Out[1]= π

In[2]:= π **+ 0.**

Out[2]= 3.14159

Again, note *Mathematica*'s propensity for exact answers. You will often use π to indicate the radian measure of an angle to be input into a trigonometric function. There are examples in the next section.

It is possible to enter each of these three constants directly from the keyboard, as well. You can type esc**p**esc for π, esc**ee**esc for e, and esc**ii**esc for i.

⚠ You can also type **Pi** for π, **E** for e, and **I** for i. The capitalizations are important. These do not look as nice, but it illustrates an important point: it is possible to type any *Mathematica* input using only the characters from an ordinary keyboard. That is, every formatted mathematical expression that can be input into *Mathematica* has an equivalent expression constructed using only characters from the keyboard. Indeed, versions 1 and 2 of *Mathematica* used only such expressions. These days, the keyboard, or **InputForm**, of an expression is used when you include a *Mathematica* input or output in an email message (say, to a friend or to your professor). If you copy a formatted expression such as $\pi^{1/3}$ from *Mathematica* and paste it into an

email or text editor, you'll find that it becomes Pi^(1/3) (or just π^(1/3) if the editor has the π symbol available). The point is that it is exceedingly simple to include formatted *Mathematica* expressions in plain text environments. Note that you can display any input cell in **InputForm** from within *Mathematica* by clicking on its cell bracket to select it, and going to the Cell menu and choosing ConvertTo ▷ InputForm.

```
In[3]:= Pi == π
Out[3]= True
```

1.10 Typing Commands in *Mathematica*

In addition to the basic arithmetic features discussed earlier, *Mathematica* also contains hundreds of *commands*. Commands provide a means for instructing *Mathematica* to perform all sorts of tasks, from computing the logarithm of a number, to simplifying an algebraic expression, to solving an equation, to plotting a function. *Mathematica*'s commands are more numerous, more flexible, and more powerful than those available in any hand–held calculator, and in many ways they are easier to use.

Commands are typically typed from the keyboard, and certain rules of syntax must be strictly obeyed. Commands take one or more *arguments*, and when entered transform their arguments into output. The typical syntax for a command is:

Command[*argument*] or **Command**[*argument1*, *argument2*]

Rules for Typing Commands

When typing commands into *Mathematica*, it is imperative that you remember a few rules. The three most important are:

- Every built–in command begins with a capital letter.Furthermore, if a command name is composed from more than one word (such as ArcSin or FactorInteger) then each word begins with a capital letter, and there will be no space between the words.
- The arguments of commands are enclosed in square brackets.
- If there is more than one argument, they are separated by commas.

When you begin typing a command, the individual characters will be blue. They will change to black as soon as they match the name of a built–in command. This syntax coloring mechanism is designed to help you spot typing errors. If you were to type **Arcsin** instead of **ArcSin**, for example, it would remain blue, indicating that it's not right.

Here are some examples of commonly used commands: