CHEM 4331 Computer Laboratory 1: Mathematics of Physical Chemistry

"*Computer! Computer?... Hello, computer.*"
Lieutenant Commander Montgomery "Scotty" Scott
Star Trek IV: The Voyage Home (1986)

Contents:

1. Objectives and Notes on Completing the Exercises

The purpose of this first computer Physical Chemistry laboratory is to briefly review [and, perhaps, somewhat extend] mathematical techniques learned in the Pre-Calculus and Calculus courses, and apply those to solving relatively simple problems in already familiar basic Physics and Chemistry. In the process you will also be introduced to one of [if not the!] best computer algebra software *Mathematica* [1,2,3] designed by Dr. Stephen Wolfram [4] (PhD in particle physics from Caltech) and developed by Wolfram Research [5]. The main goal of this lab is to help you better understand the material presented in CHEM 4330, and, perhaps, demonstrate that doing Mathematics can be exciting and fun!

The laboratory requires you to complete in *Mathematica* ten (10) exercises included in this handout. Each exercise is worth 10 points. *Mathematica* notebooks for all exercises must be submitted to a D2L dropbox by the date and time indicated in your syllabus**.**

Each exercise contains a set of objectives listed as **(a)**, **(b)**, **(c)** etc.. A successful completion of each objective includes obtaining

1) correct numerical value(s) and/or formula/equation(s), and
2) correct units (where applicable).

Even if one of those is incorrect or missing (for example, units!), no credit shall be given for the objective. When in doubt, go back to the worked-out examples, and if it does not help, ask your lab instructor.

In each Mathematica notebook, you **must** clearly indicate/label objectives (answers and graphs) with symbols **(a)**, **(b)**, **(c)** … as instructed in the Exercises. For example:

(a) $v(t) = 4.4 + 0.6t^2 (m/s)$
(b) $x(t) = 1.4 + 4.4t + 0.2t^3$ (m)
….
(e) $a(2s) = 2.4\ (m/s^2)$
….

Unlabeled answers will be ignored, i.e. **no points shall be given for unlabeled answers**.

When working on the exercises, it is permitted to consult with your classmates. In fact, we encourage discussions that promote a creative learning environment. However, **copying each other's code is strictly prohibited!** You absolutely must write your own code following the guidelines and style of the worked-out examples provided in the handout. As such, you will have to come up with your own comments and names for the variables/parameters used in your calculations. Of course, make sure that those names are meaningful (see worked-out examples provided in this handout). This is your chance to be creative – make good use of it!

A couple of [hopefully, useful] suggestions. When working with *Mathematica* notebooks, save them often as it is not uncommon for Windows OS to crash. When done for the day, do not forget to either copy all your files from the computer to your USB flash drive, or email the files to your own email (home, college etc.) account. Alternatively, you can store files in the "cloud" – for more information contact the MTSU IT department (ITD). Note that computers in SCI 3095 will reboot every night, and *all* user-created files located on the desktop and disk D: (where applicable) will be permanently deleted. This, of course, applies to ALL CHEM 4331 labs (not just this one)

Good luck! … and *try* to have fun!

2. Introduction to Mathematica

Mathematica is a powerful computer algebra software for numerical and symbolic calculations. It is extensively used as a research tool in many scientific, engineering, mathematical and computing fields [3].

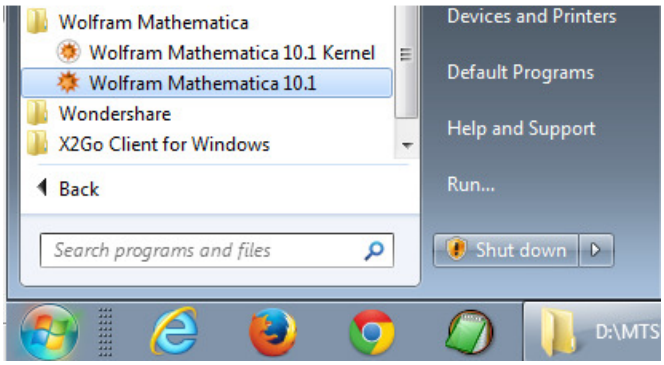
Mathematica is split into two parts, the kernel and the front end [3]. The kernel interprets expressions (*Mathematica* code) and returns result expressions [3]. The front end provides a GUI, which allows the creation and editing of Notebook documents containing program code with prettyprinting, formatted text together with results including typeset mathematics, graphics, GUI components, tables, and sounds [3]. All contents and formatting can be generated algorithmically or interactively edited [3]. Most standard word processing capabilities are supported [3]. It includes a spell-checker but does not spell check automatically as you type [3].

Documents can be structured using a hierarchy of cells, which allow for outlining and sectioning of a document and support automatic numbering index creation [3]. Documents can be presented in a slideshow environment for presentations [3]. Notebooks and their contents are represented as *Mathematica* expressions that can be created, modified or analyzed by *Mathematica* programs [3]. This allows conversion to other formats such as TeX or XML [3].

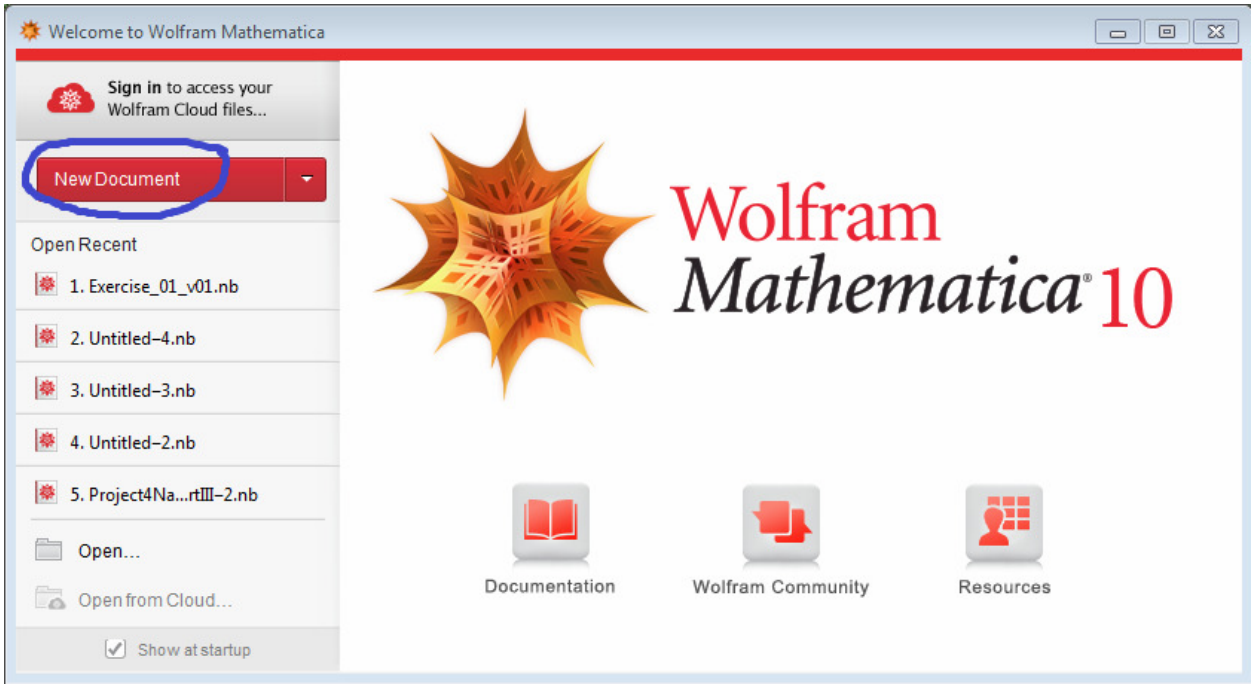
The front end includes development tools such as a debugger, input completion and automatic syntax coloring [3].

To start *Mathematica*, click on the *Mathematica* icon located on your desktop

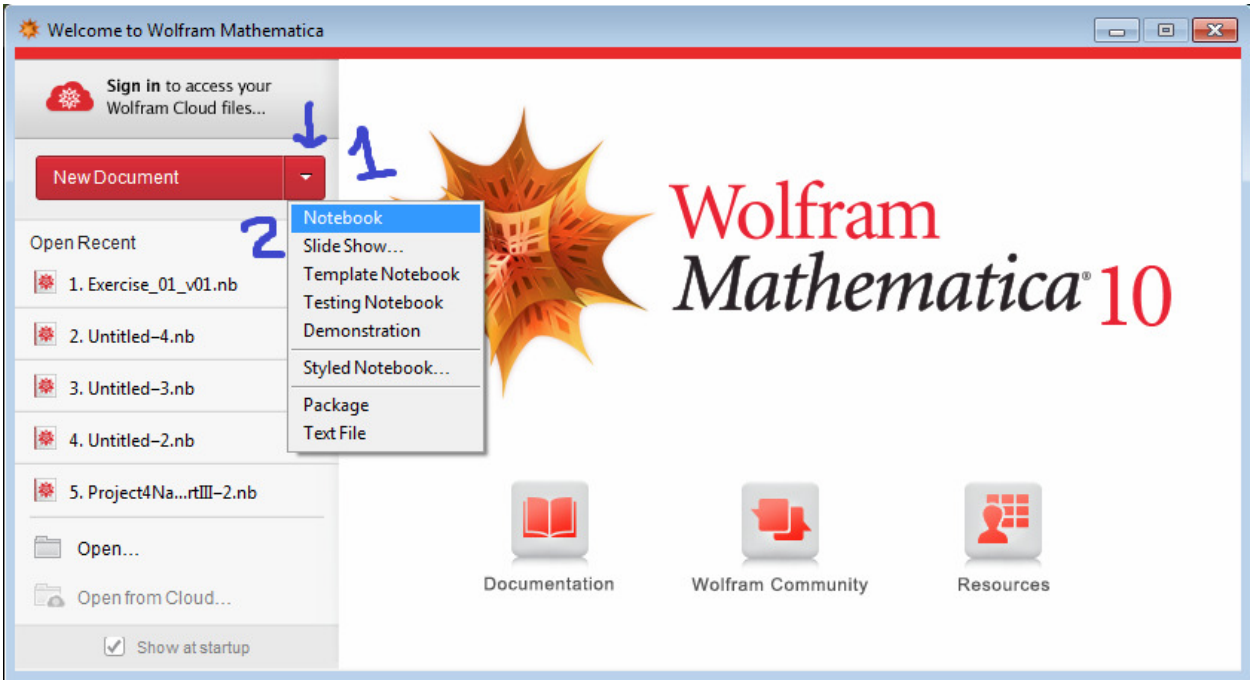
or go through the Windows "Start" menu: "Start" → "All Programs" → "Wolfram Mathematica" → "Wolfram Mathematica":



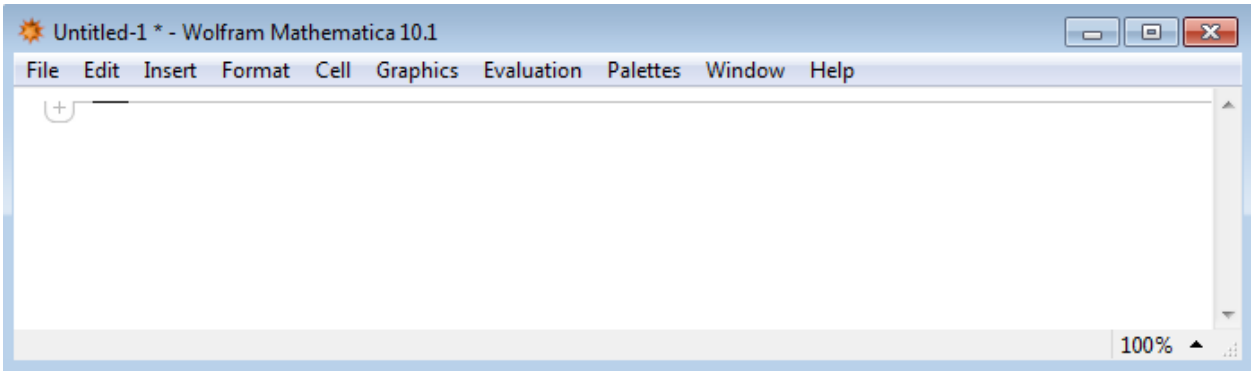
Note that in this tutorial I will be using *Mathematica* 10.1, while you may be using *Mathematica* v9 or v11. Not to worry, the interface and functionality of the two versions are almost the same! After the program is loaded, you will see a welcoming screen



Note that because I have already worked with *Mathematica*, I have several projects listed under the "Open Recent" menu. Now, left-click on the "New Document" button (or choose the "Notebook" option in the "New Document"



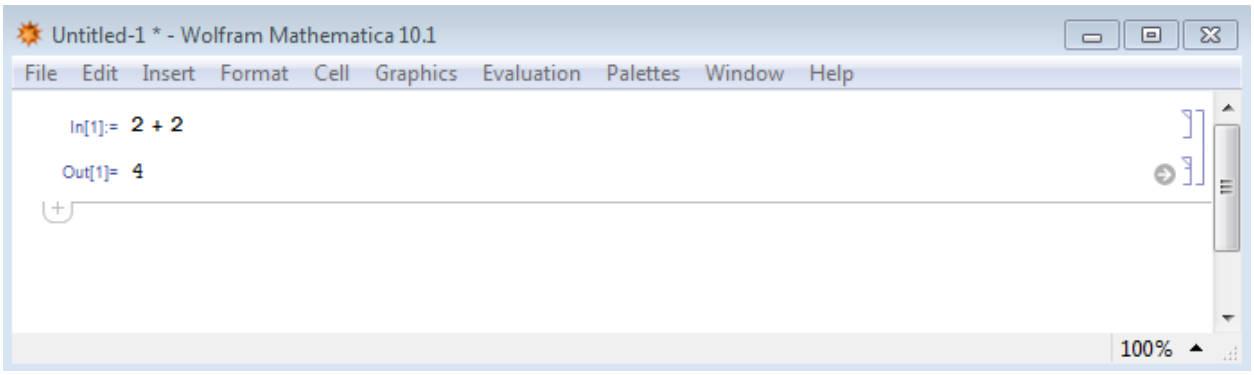
Whatever the case, a new *Mathematica* notebook will open.



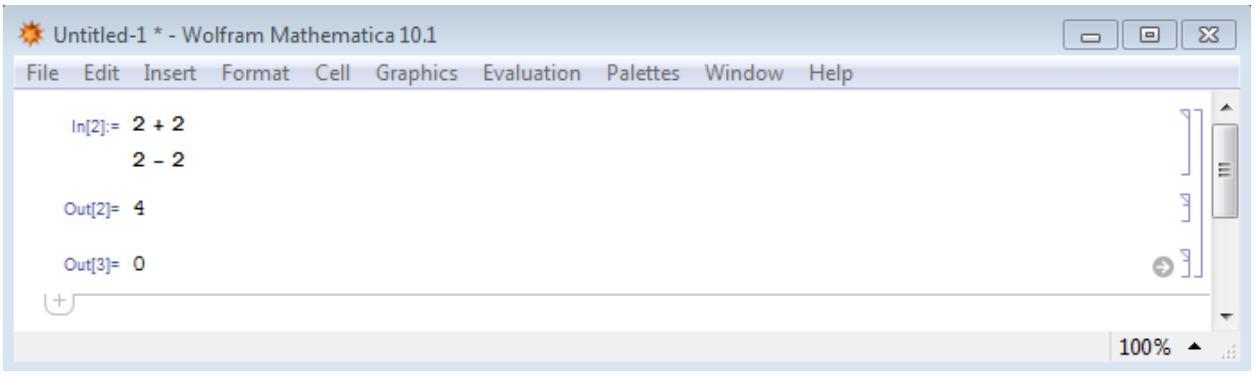
The *Mathematica* fully-interactive notebook interface looks almost like a text editor, and is quite intuitive. There is a menu bar, and an empty [for now] notebook canvas. Several short tutorials on how to start working with a notebook are available on the *Mathematica* website at:

https://reference.wolfram.com/language/tutorial/UsingANotebookInterface.html
https://reference.wolfram.com/language/guide/MenuItems.html
https://reference.wolfram.com/language/guide/NotebookBasics.html
https://reference.wolfram.com/language/tutorial/WorkingWithTheNotebookInterfaceOverview.html

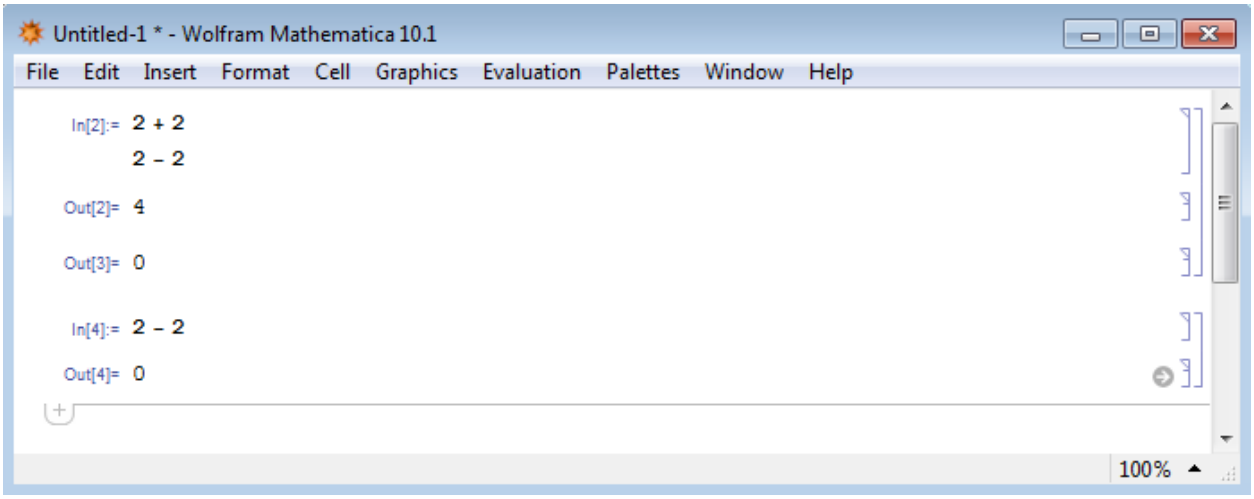
Let's do a couple of simple calculations. In your empty notebook, type in "2+2" (without the double quotes, of course) and hit the Shift + Enter key combination to perform a computation:



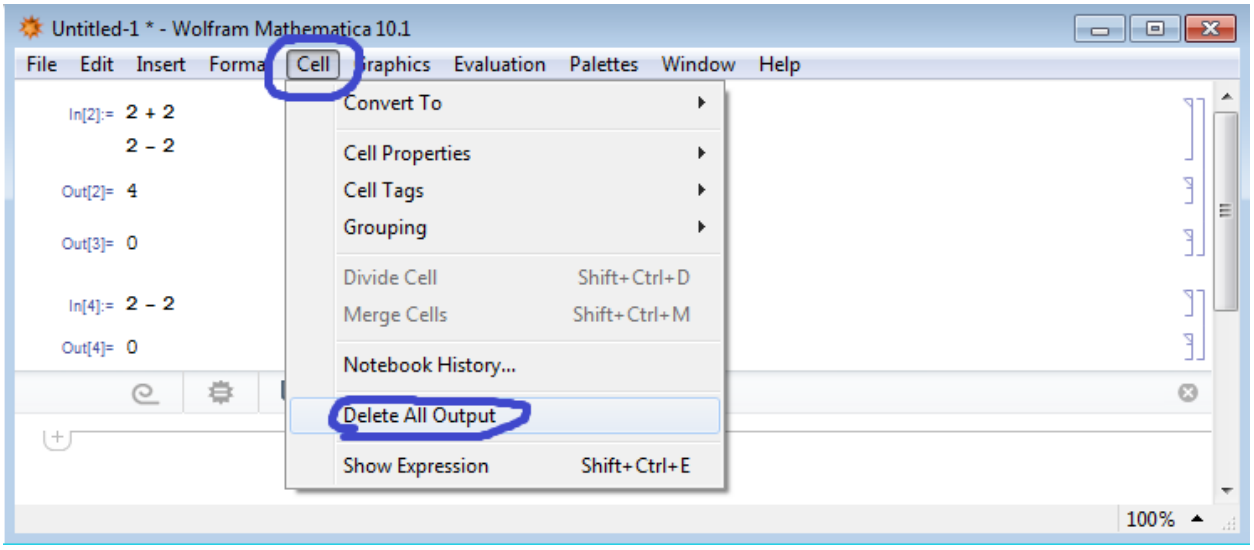
Note how *Mathematica* labels the input (In[*n*]) commands and the output (Out[*n*]), where *n* is a greater-than-zero integer number. Usually, for each input command, there is output. On the very right side of the notebook, you find cell (inner) and group (outer) brackets. Each input cell can contain one or more commands. At this time, there is only one input command in the first cell (i.e. "2+2" in cell In[1]). If you wish to add another command, for example "2-2", you can add it to the first input cell (In[1]) by positioning the mouse cursor inside that cell and typing the text. When done, hit Shift + Enter to get *Mathematica* to do a computation:



Alternatively, you can open a new cell by positioning the cursor below the last output cell, and type in the command (of course, do not forget the "magic" Shift + Enter key combination!):



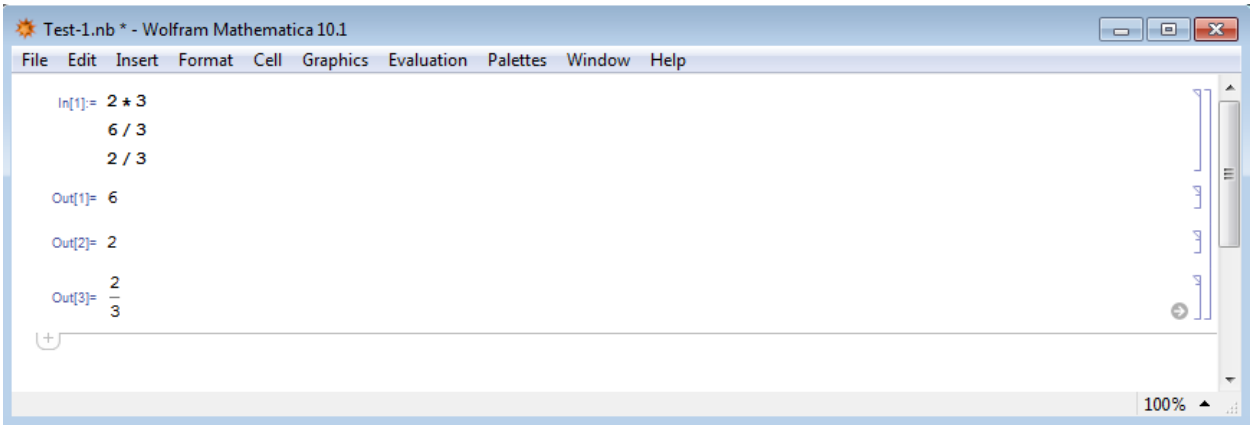
If at any time you wish to delete all your output cells, you can do so by using the "Delete All Output" option under the "Cell" menu:



If you wish to delete a particular cell (input or output), simply left-click the bracket of the cell you wish to get rid of, and hit the "Delete" button on your keyboard. Now, delete all output and the

bottom set of cells, and, using the "File" menu, save the notebook to the Windows desktop as "Test-1.nb". Now select all previously entered input and delete all cells to leave a blank notebook.

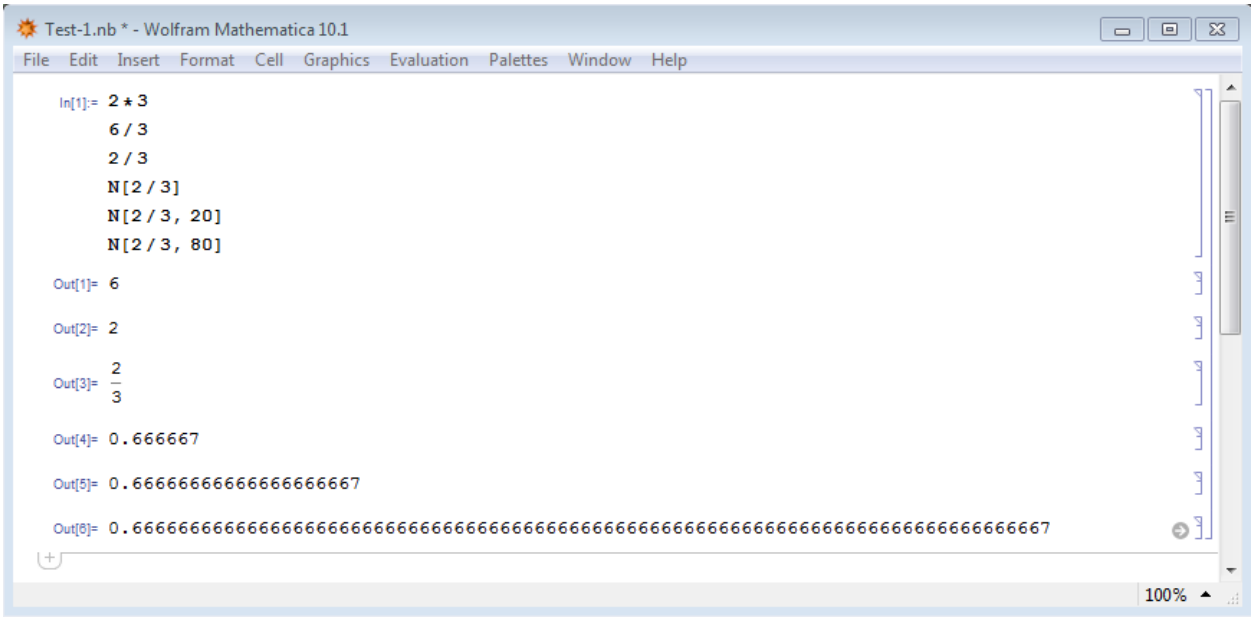
Mathematica obviously can do much more than a simple addition and subtraction. To multiply the two numbers, use the "*"symbol. To divide the two numbers, use the "/"symbol.



Note the difference between the results of the two divisions. Division of 6 by 3 gives 2 which is an exact number. Why didn't *Mathematica* do the same for the 2/3 operation? This is NOT because *Mathematica* does not know how to divide 2 by 3, but rather because it knows that the result is 0.666666666……. Thus, rather than truncating the number at some arbitrary decimal, it keeps it as a fraction. If you want a numerical value, use the N[...] operator:

https://reference.wolfram.com/language/ref/N.html?q=N

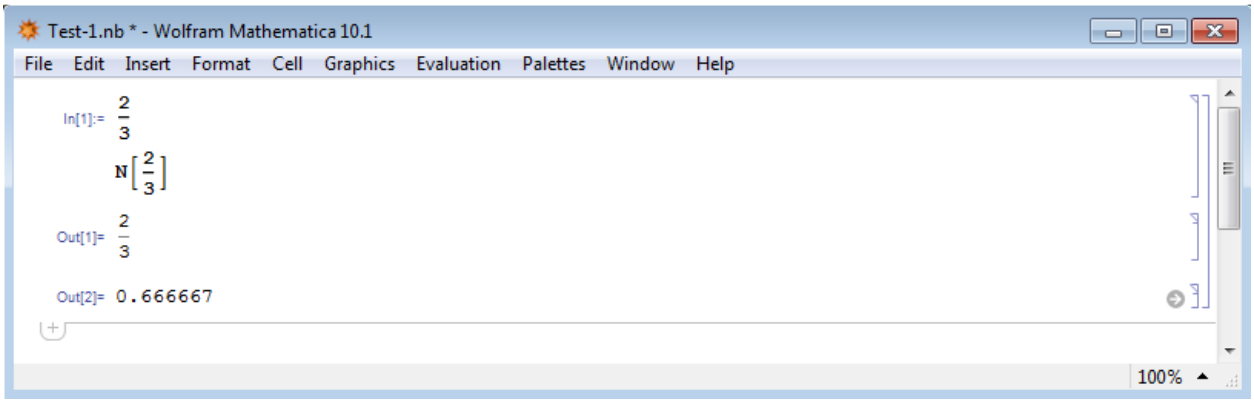
By the way, to make input easier, you can use the good old copy-and-paste option. First, select the text you wish to copy using the "left-click and drag" technique. Then, select "Copy" from the "Edit" menu (or simply hit "Ctrl+C"), and paste using either the "Paste" option from the "Edit" menu or the "Ctrl+V" key combination. Trust me, it makes working with *Mathematica* a lot easier and much more fun!



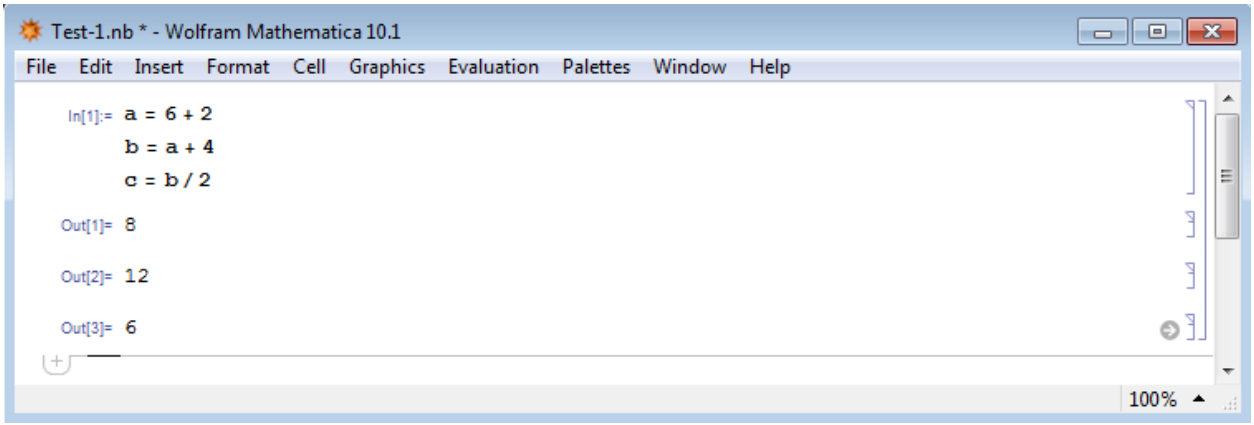
Note how the `N[2/3]` command evaluates the fraction to some default number of decimals. The `N[2/3,20]` command evaluates the same to 20 decimal points. `N[2/3,80]` gives the result to 80 decimals. How about that?!

When fractions are entered using the "/" symbol, the code is not very readable. The Wolfram System's notebook interface is a very powerful typesetting system that allows you to enter formulas as they are written in mathematical literature, using two-dimensional notation such as superscripts, subscripts, and so on. For example, try entering the "2/3" fraction using the Ctrl + / key combination:

https://reference.wolfram.com/language/tutorial/EnteringInputInNotebooks.html

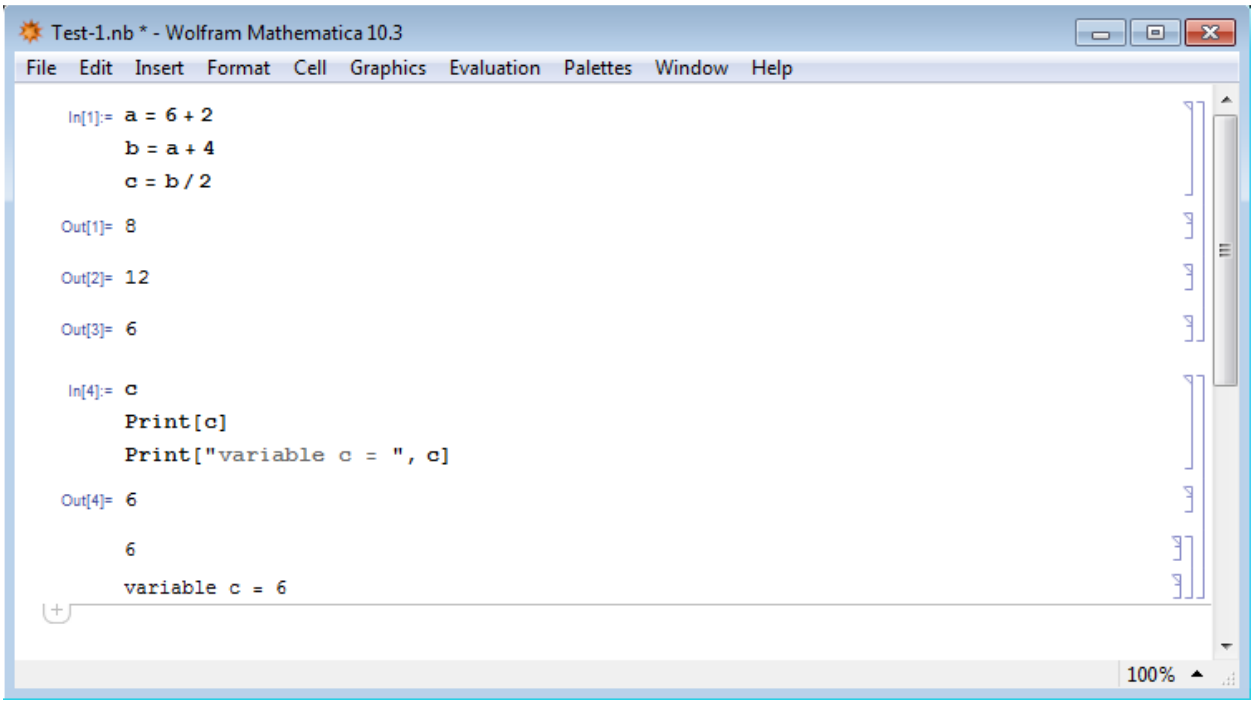


Looks much better, doesn't it? Another useful option in *Mathematica* is its ability to store the result of a mathematical operation (numerical or symbolic) in a named variable. For example,



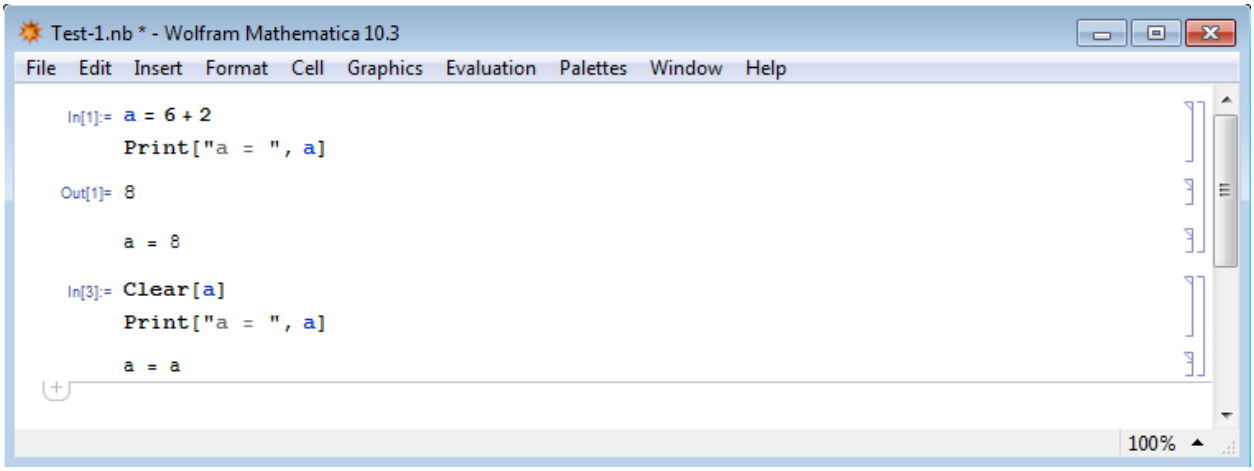
WARNING!!! Do not use subscripts, superscripts, and special symbols such as but not limited to ( ) [ ] & % $ # @ etc., in the names of the *Mathematica* variables because it can easily lead to program errors and wrong results!

To see what is stored in a given variable, simply type in its name, or use the Print[] command:



Note how in the last statement (Print["variable c = ", c]) we have supplemented the printout with text. This type of output can be very helpful when writing a code, and will be required when processing exercises.

If you wish to clear the value or definition of a given variable, use the Clear[] command:

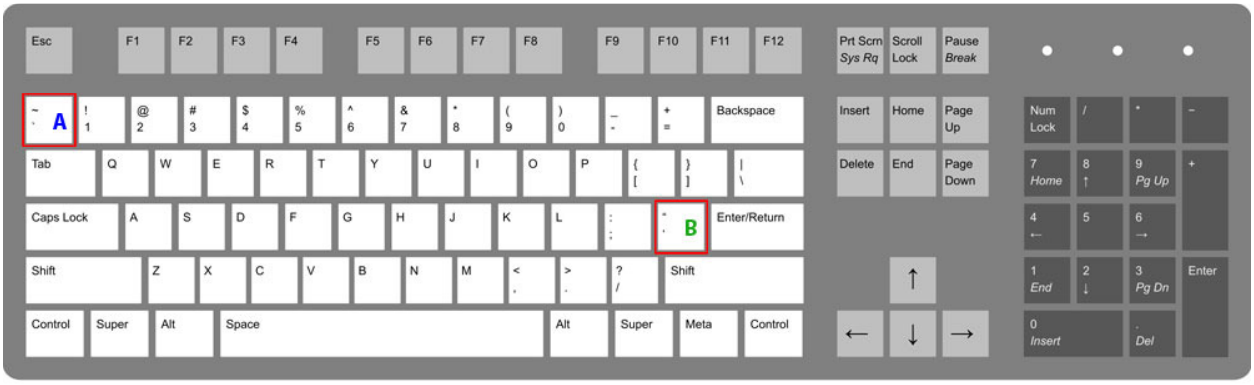


Clearing the declared values as they are no longer needed is definitely a good habit [6]. Beginning a new Mathematica session, like starting with a clean slate, for each problem is a good practice [6].

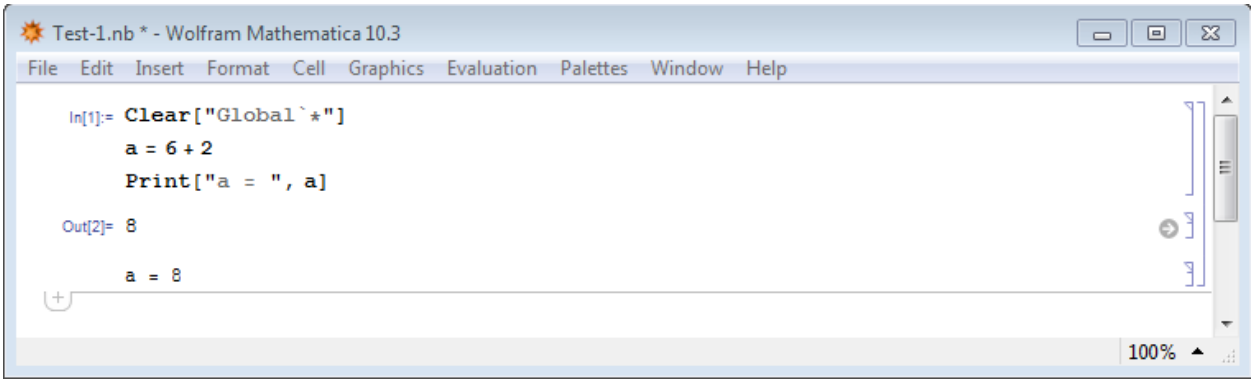
An alternative, almost as good as starting with a clean slate, is to execute the command Clear["Global` * "] which clears the values for all symbols in the Global` context, except those that are protected [6]. Note that symbol " ` " is not an apostrophe, it is the so-called "grave accent":

https://en.wikipedia.org/wiki/Grave_accent

which can be located in your keyboard as shown below [7]:

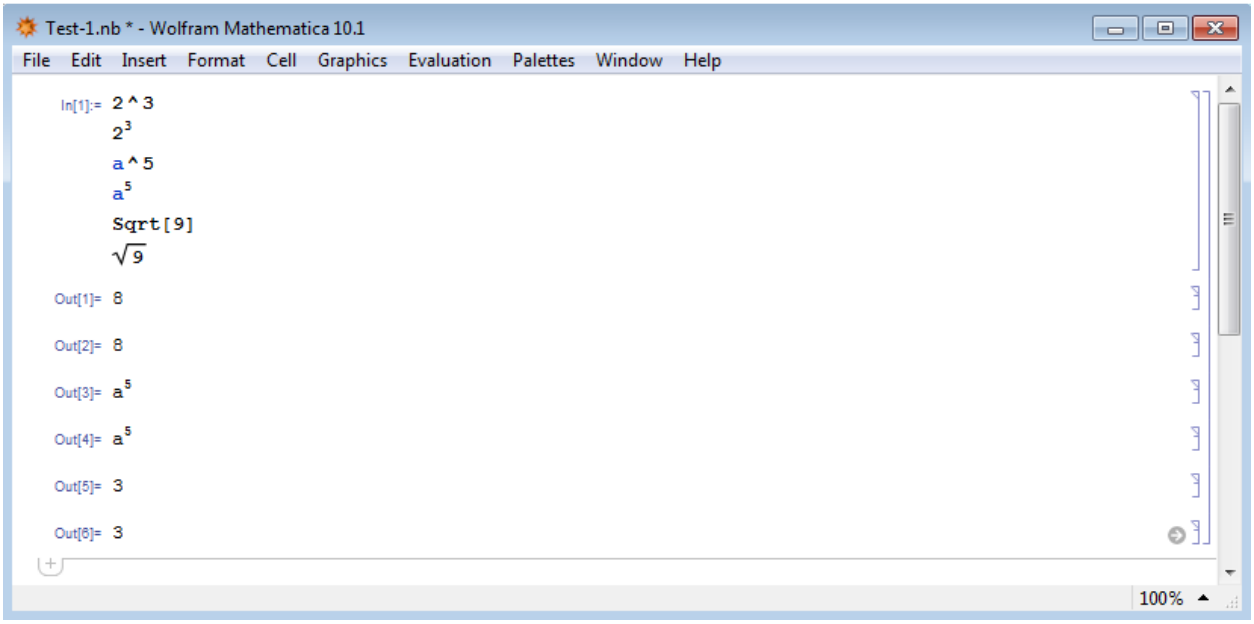


Here is an example that shows how to use the Clear["Global` * "] statement:



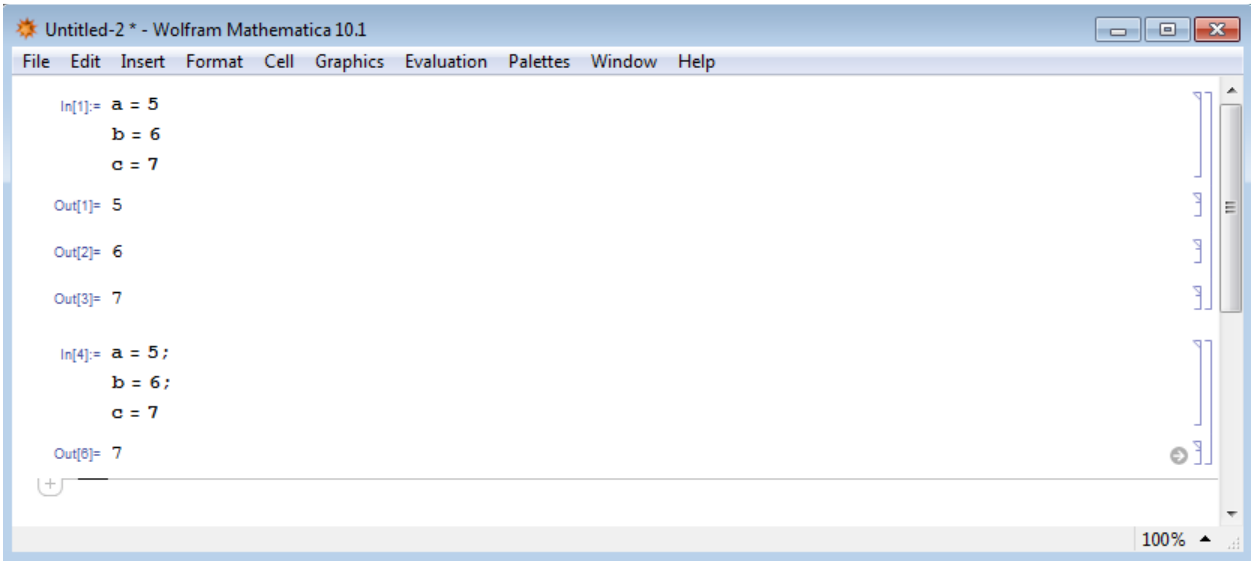
If you wish to execute the entire notebook automatically (one cell after another), use the "Evaluate Notebook" option in the "Evaluation" menu. Keep in mind, that *Mathematica* keeps definitions and assignments of all variables in memory as it executes notebook(s). Sometimes, you want to start over, and erase all those assignments. In this case, use option "Quit Kernel" → "Local" in the same "Evaluation" menu, and re-evaluate the cells of interest, or use the "Evaluate Notebook" option to evaluate *all* cells in your notebook. To clear the output only, use the "Cell" → "Delete All Output" option.

Now, let's try some other mathematical operations that you may find useful. To raise a number or a variable (including a symbolic variable) to a power, you can either use the "^" symbol or Ctrl + ^ . To take square root, you can use the Sqrt[] function, or Ctrl + 2.

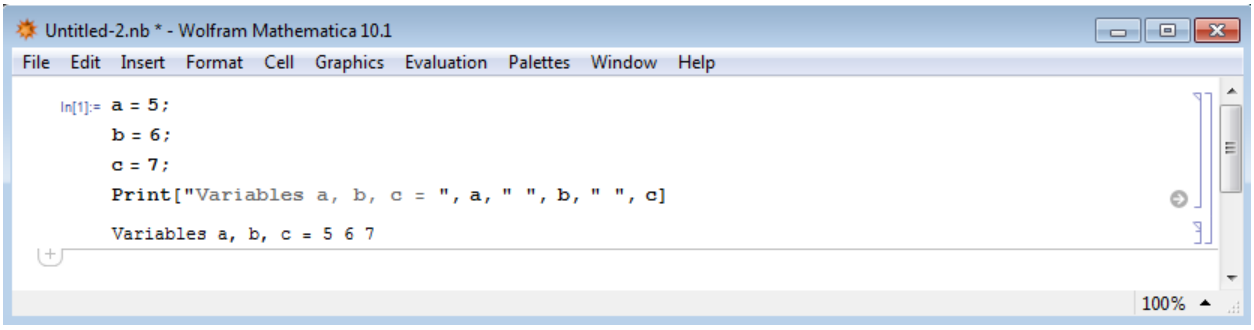


As you can see, the readability of the notebook improves dramatically when using the *Mathematica* typesetting system. Also note that a variable need not have a numerical value assigned for *Mathematica* to apply a certain operation to it. This comes in very handy when doing symbolic computations (we will get to that in just a moment).

I hope that you have noted that *Mathematica* always prints out variables and parameters as they are defined. If you do not want *Mathematica* to do that, terminate a command with the semicolon symbol (";"). For example, compare the two seemingly identical sequences:



In this lab, we require your *Mathematica* notebooks to have user-friendly printouts of parameters using the Print[] option. In the following, I show how to do that:

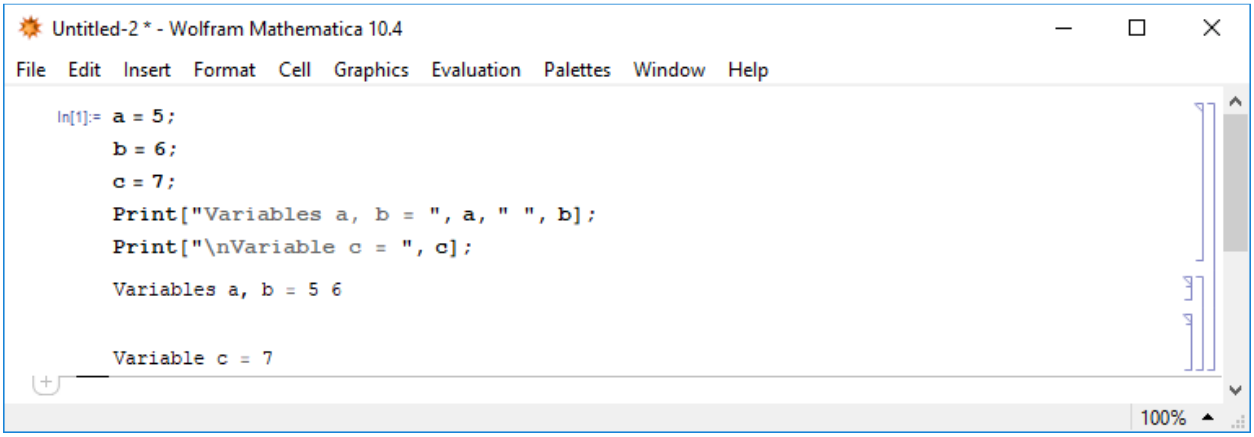


Note how the values of parameters a, b, and c are *not* printed out as they are defined because each line is terminated with the semicolon. Indeed, rather than printing them out one by one without annotation, it is much better to print out their values in a separate Print[] statement with accompanying [self-explanatory] text.

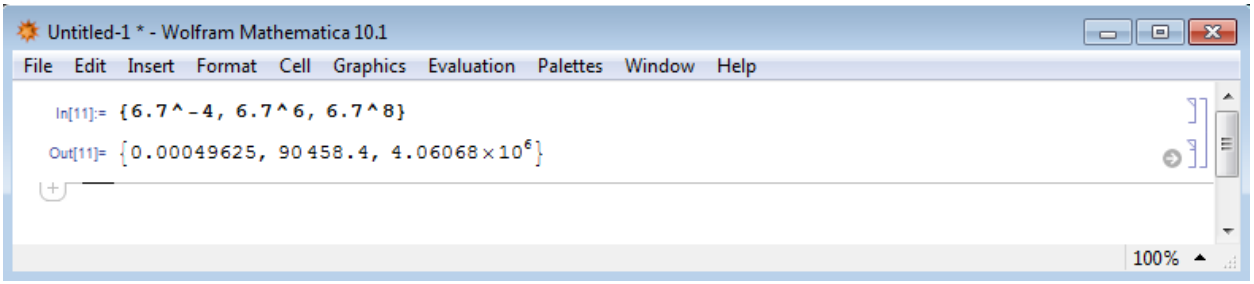
It is often desirable to separate two printout statements by a blank line. While one can use the Print[" "] option to print out a blank line, a more elegant solution is to include the newline character "\n" in the second Print[] statement.

http://reference.wolfram.com/language/tutorial/NewlinesAndTabsInStrings.html

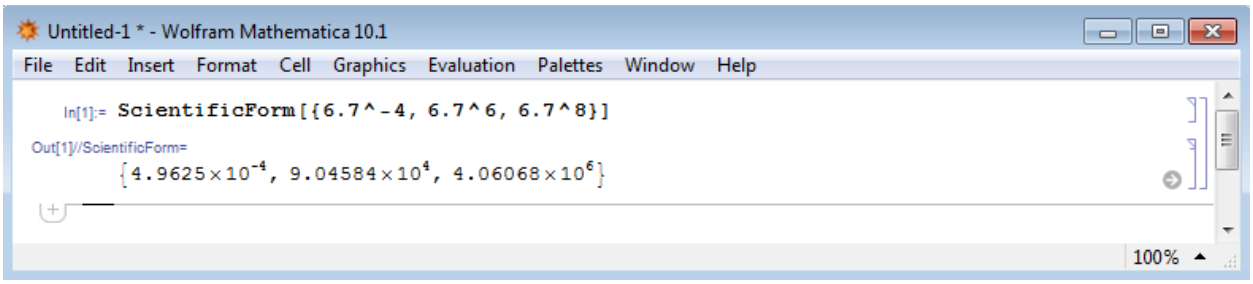
For example,



By default, *Mathematica* prints out all numbers in some default output format. Only very large and very small numbers are given in scientific notation:



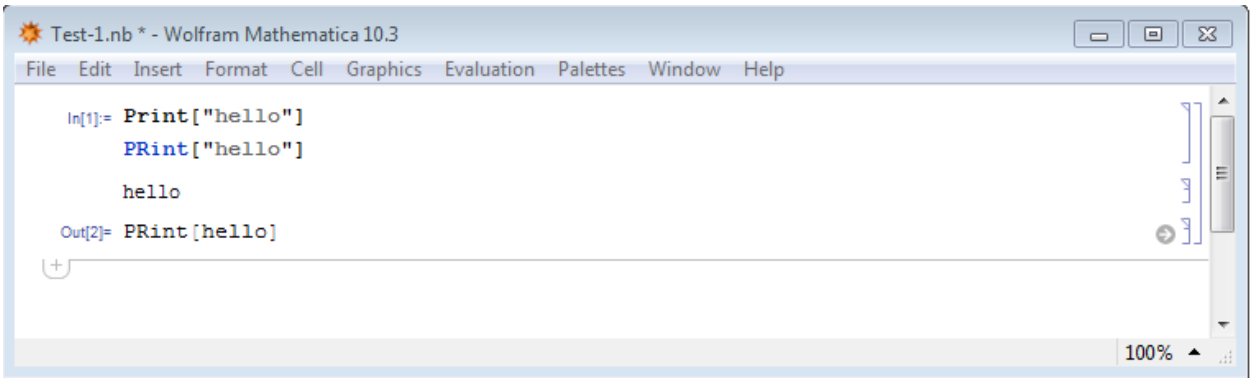
To force *Mathematica* print out numbers in scientific notation, use the ScientificForm[] option:



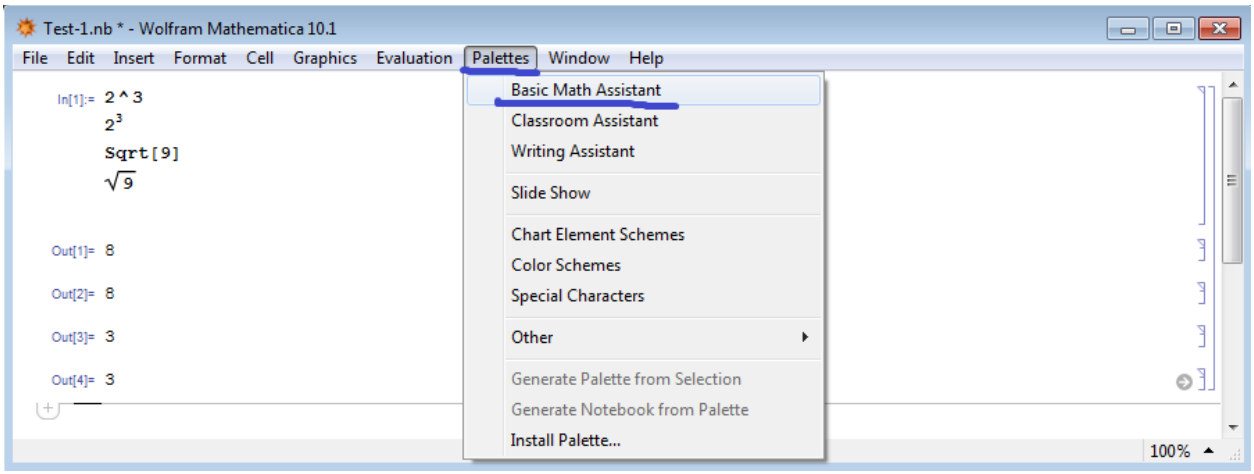
There are many other number output formats available. For more information, refer to

http://reference.wolfram.com/language/tutorial/OutputFormatsForNumbers.html

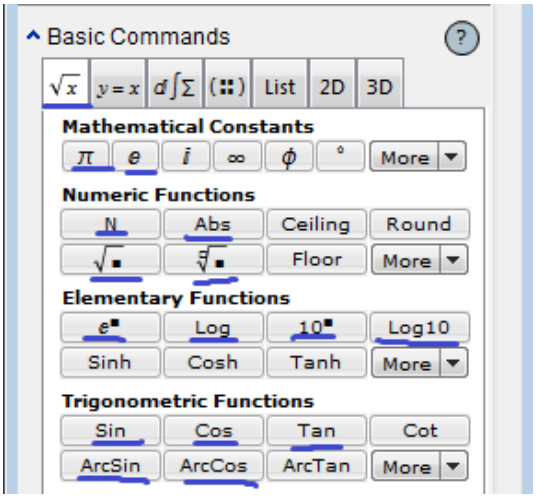
Note how all Mathematica built-in functions (Print[], Clear[], N[], Sqrt[] etc. are spelled with the first character in upper case, while all characters that follow are in lower case. **If you misspell a name of the function, the function will not work! At the same time, the name of the unrecognized function will be highlighted in blue:**



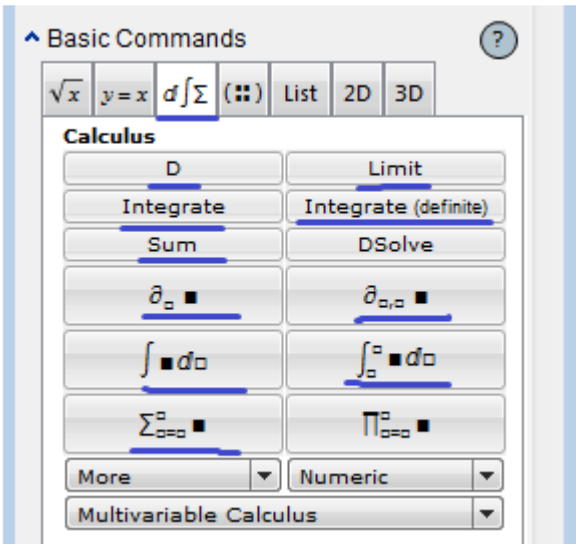
Finally, if you do not wish to remember all the special key combinations for symbols of mathematical operations, you can use the "Palettes" option. Left-click on the "Palettes" menu and choose the "Basic Math Assistant" option (actually, it is *not* very basic at all!):



You will immediately get a new "Basic Math Assistant" palette, which has two options" "Basic" and "Advanced". Try switching between those to see the difference. Note that in addition to a number of algebraic and trigonometric functions,



it has many Calculus-related options, such as derivatives, limits, sums, and integrals:



Finally, before you can start working on the exercises, let's look at a couple of very useful functions.

Functions Simplify[] and FullSimplify[] :

http://reference.wolfram.com/language/ref/Simplify.html

http://reference.wolfram.com/language/ref/FullSimplify.html

request simplification of the specified equation(s). These functions are extremely powerful because *Mathematica* knows how to handle a lot of mathematical identifies.

For example, here is a simple familiar simplification:

```
In[1]:= Simplify[a^2 - 2*a*b + b^2]
Out[1]= (a - b)^2
```

Perhaps, something a little more complicated involving a trigonometric function:

```
In[2]:= Simplify[-4*Sin[x]^3 + 3*Sin[x]]
Out[2]= Sin[3 x]
```

In the following example, we

1) define function *f* that depends on variables *x*, *y*, and *z*, which we usually write as $f(x, y, z)$,
2) differentiate *f*(*x*,*y*,*z*) with respect to (w.r.t.) *x*., and
3) simplify the resulting expression:

```
In[3]:= f = x / Sqrt[x^2 + y^2 + z^2]
Out[3]= x / Sqrt[x^2 + y^2 + z^2]

In[4]:= dfdx = D[f, x]
Out[4]= -x^2 / (x^2 + y^2 + z^2)^(3/2) + 1 / Sqrt[x^2 + y^2 + z^2]

In[5]:= Simplify[dfdx]
Out[5]= (y^2 + z^2) / (x^2 + y^2 + z^2)^(3/2)
```

Note that taking a derivative of function $f(x, y, z)$ w.r.t. variable *x* is very easy in *Mathematica*. Alternatively, we could have written:

```
In[1]:= dfdx = D[x / Sqrt[x^2 + y^2 + z^2], x]
Out[1]= -x^2 / (x^2 + y^2 + z^2)^(3/2) + 1 / Sqrt[x^2 + y^2 + z^2]
```

However, from our point of view the first way is more efficient because one may want to do something else with function $f(x, y, z)$, i.e. re-use the definition of function *f*. Perhaps, you know that when a function of several variables is differentiated w.r.t. one of the variables, the result is called a *partial* derivative of the function with respect to the variable. For example, when differentiating function $f(x, y, z)$ w.r.t. parameter *x*, we get partial derivative $\frac{\partial f(x, y, z)}{\partial x}$. Because

function $f(x, y, z)$ depends on three parameters, there will be three partial derivatives. The remaining two partial derivatives are $\frac{\partial f(x, y, z)}{\partial y}$ and $\frac{\partial f(x, y, z)}{\partial z}$. The *total derivative* (aka *differential change* or *total differential*) of function $f(x, y, z)$, denoted usually as $\mathrm{d}f(x, y, z)$, is the sum of all of its partial derivatives:

$$\mathrm{d}f(x, y, z) = \frac{\partial f(x, y, z)}{\partial x}\mathrm{d}x + \frac{\partial f(x, y, z)}{\partial y}\mathrm{d}y + \frac{\partial f(x, y, z)}{\partial z}\mathrm{d}z,$$

where each of $\mathrm{d}x$, $\mathrm{d}y$, and $\mathrm{d}z$ is called simply a *differential*.

Ok, now that you know how to do some basic [and not very basic] things in *Mathematica*, let's see how you can use it to perform familiar scientific computations.

3. Solutions of Algebraic Equations

Before diving into Calculus (which is what Physical Chemistry is all about), let's learn how to solve equations with *Mathematica*. In this section we limit ourselves to solving [relatively simple] algebraic equations: *quadratic*, *cubic*, and *quartic*. For example, we wish to solve a standard *quadratic* equation in the form

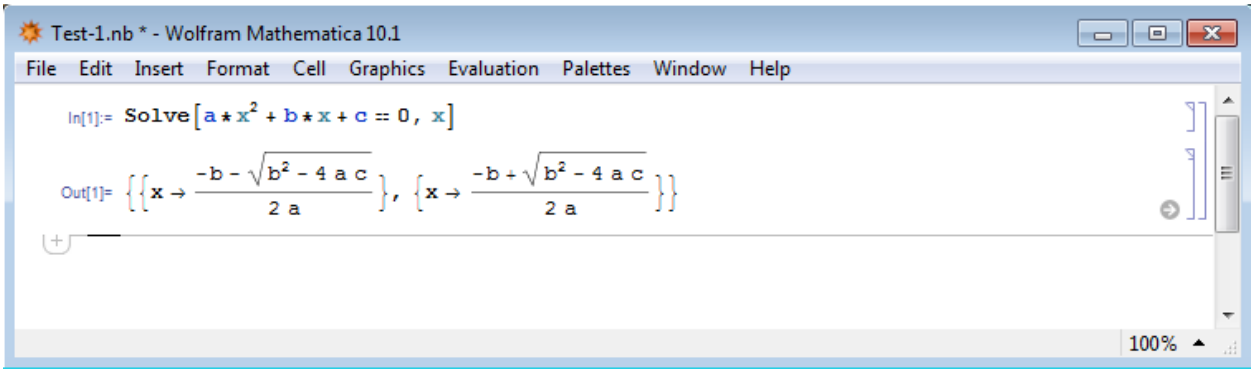
$$ax^2 + bx + c = 0 \tag{1}$$

Of course, we know that there are two solutions for x given by

$$x = \frac{-b \pm \sqrt{b^2 - 4ac}}{2a} \tag{2}$$

Let's see if *Mathematica* can do that. In this case, we can use the Solve[] option:

http://reference.wolfram.com/language/ref/Solve.html



Yes, it can! Note how I have used the "==" symbol to define the equality telling *Mathematica* that the right had side of the equation (rhs) is equal to the left hand side (lhs).

Actually, according to the *Mathematica* documentation,

"[b]*uilt into the Wolfram Language is the world's largest collection of both numerical and symbolic equation solving capabilities—with many original algorithms, all automatically accessed through a small number of exceptionally powerful functions. The Wolfram Language's symbolic architecture allows both equations and their solutions to be conveniently given in symbolic form, and immediately integrated into computations and visualizations."*

http://reference.wolfram.com/language/guide/EquationSolving.html

Example 3.1. Let's consider the following exercise taken from your Physical Chemistry textbook [8]. At a certain temperature, for reaction

$$\mathrm{N_2(g) + 3H_2(g) \longleftrightarrow 2NH_3(g)} \tag{3}$$

the equilibrium constant K (K_p) is equal to 977. For this particular example, the equilibrium is given by

$$K = K_\mathrm{p} = \frac{p_{\mathrm{NH_3}}^2}{p_{\mathrm{N_2}} p_{\mathrm{H_2}}^3} \tag{4}$$

where p_J is the partial pressure of substance J divided by standard pressure of 1 bar, i.e.

$$p_\mathrm{J} \leftarrow p_\mathrm{J} / p^\mathrm{o} \tag{5}$$

where $p^\mathrm{o} = 1$ bar. Note that the equilibrium constant K is a dimensionless (unitless) quantity. In general, the equilibrium constant K equals the value of the reaction quotient Q at equilibrium. For a general chemical reaction of the form [8]:

$$a\ \mathrm{A} + b\ \mathrm{B} \leftrightarrow c\ \mathrm{C} + d\ \mathrm{D} \tag{6}$$

the equilibrium constant is given by [8]

$$K = Q_\mathrm{equilibrium} = \left(\frac{\alpha_\mathrm{C}^c \alpha_\mathrm{D}^d}{\alpha_\mathrm{A}^a \alpha_\mathrm{B}^b} \right)_\mathrm{equilibrium} \tag{7}$$

where α_J is the activity of species J [8]. For perfect gases, $\alpha_\mathrm{J} = p_\mathrm{J} / p^\mathrm{o}$, the partial pressure of J relative to the standard pressure $p^\mathrm{o} = 1$ bar [8]. For solutes in very dilute solutions, $\alpha_\mathrm{J} = [\mathrm{J}] / c^\mathrm{o}$, the molar concentration (molarity) of J relative to the standard value $c^\mathrm{o} = 1$ mol/L (M) [8]. If one wishes to discuss a gas-phase reaction in terms of molar concentrations (the amount of gas molecules in moles divided by the volume of the container, $[\mathrm{J}] = n_\mathrm{J} / V$), rather than partial pressures, the equilibrium constant K_p is replaced with K_c [8]:

$$K_c = \frac{[\mathrm{C}]^c [\mathrm{D}]^d}{[\mathrm{A}]^a [\mathrm{B}]^b} \tag{8}$$

where each molar concentration is raised to a power equal to the stoichiometric coefficient of the species in the chemical equation [8]. As usual, we have replaced $[\mathrm{J}]/c^\mathrm{o}$ by [J] itself, which represents the numerical value of the molar concentration of the gas J [8]. For example, for the chemical reaction shown in equation (3), [8]

$$K_c = \frac{[\mathrm{NH_3}]^2}{[\mathrm{N_2}][\mathrm{H_2}]^3} \tag{9}$$

Now, back to our example. Suppose we mix in 1.00 bar of N_2(g) with 3.00 bar of H_2(g). We need to find the equilibrium partial pressures of all three gases: **(a)** $p_{\mathrm{N_2}}$, **(b)** $p_{\mathrm{H_2}}$, and **(c)** $p_{\mathrm{NH_3}}$, and **(d)** check the final results using the equilibrium constant.

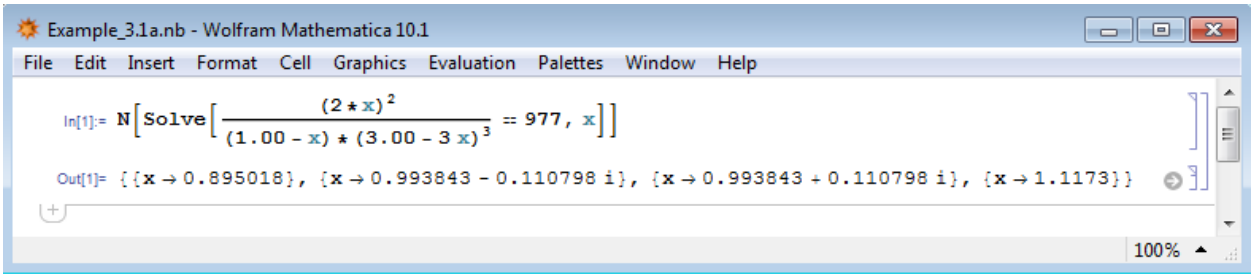
After setting up the ICE table [8],

Species	N_2	H_2	NH_3
Initial partial pressure/bar	1.00	3.00	0
Change/bar	$-x$	$-3x$	$+2x$
Equilibrium partial pressure/bar	$1.00 - x$	$3.00 - 3x$	$2x$

we get:

$$K = \frac{(2x)^2}{(1.00-x)(3.00-3x)^3} \tag{10}$$

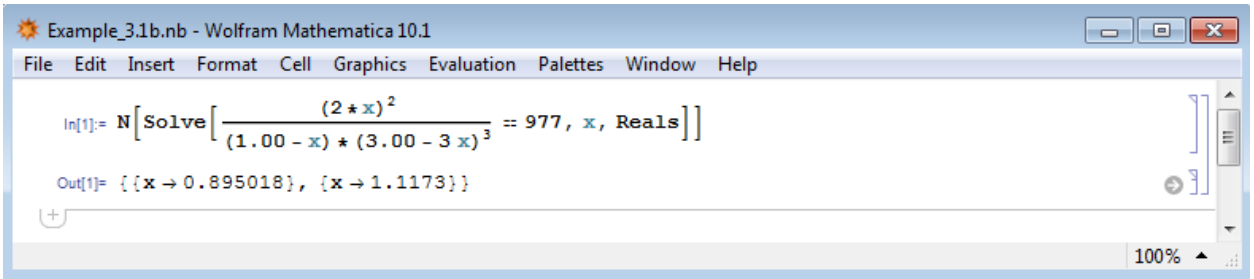
Note that this is a *quartic* equation (i.e. the highest power of x is 4)! *Mathematica* can [relatively] easily solve this equation:



Note how I used the N[] function to get *Mathematica* to output numerical values. Because this is a *quartic* equation, we get four solutions. Which one is correct? Well, mathematically all of them are correct! However, according to Physical Chemistry, partial pressure (or any other physically observable quantity) can not be a complex number (i.e. number which includes the imaginary component with $i = \sqrt{-1}$). Thus, solutions two and three are not physical, and should be discarded! In fact, in Thermodynamics and Kinetics we are usually interested in *real* solutions only

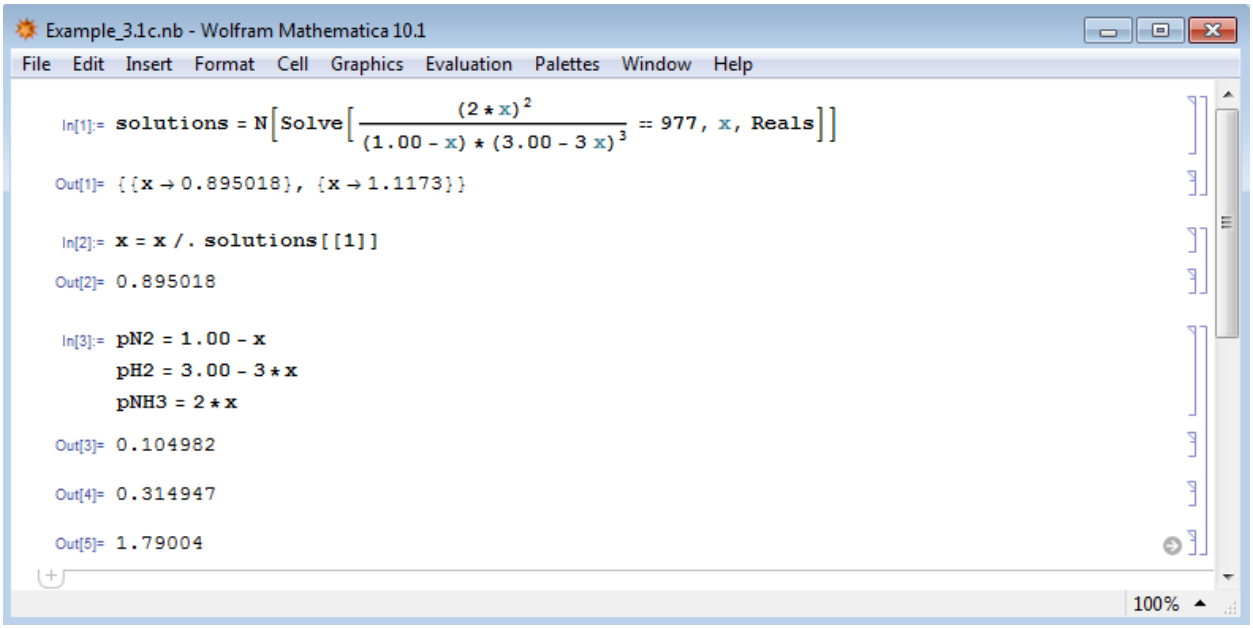
https://en.wikipedia.org/wiki/Real_number

If x is *real*, it must belong to a *domain of real numbers*. In the language of Mathematics, it can be written as $x \in \mathbb{R}$, where the symbol "$\in$" means "*is an element of*" and the symbol "$\mathbb{R}$" denotes the set of ALL real numbers. How about that! Anyway, if we are interested only in *real* solutions of a given equation, the Solve[] command can be modified as follows:



This leaves us with only two solutions. That is nice! But which one (or both) is (are) physically meaningful? Recall that in our ICE table, x denotes the change in partial pressure. The equilibrium partial pressures of $N_2(g)$ and $H_2(g)$ are (1.00-x) bar and (3.00-3x) bar. Solution x = 1.1173 bar

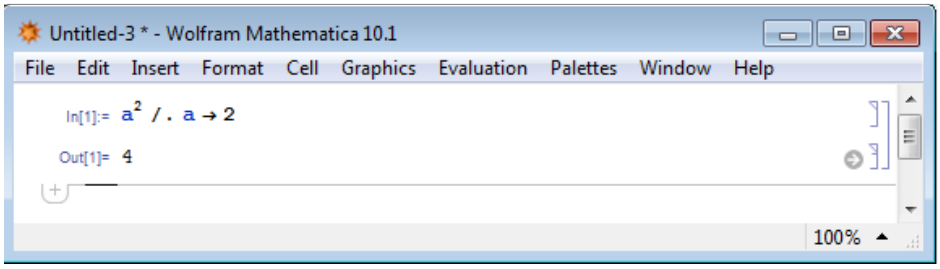
would give us negative equilibrium pressures of $N_2(g)$ and $H_2(g)$, and thus is not physically meaningful. This leaves us with only one solution: $x = 0.895018$ bar. Now we can calculate the final partial pressures of all three gases. Here is our *Mathematica* code that does that:



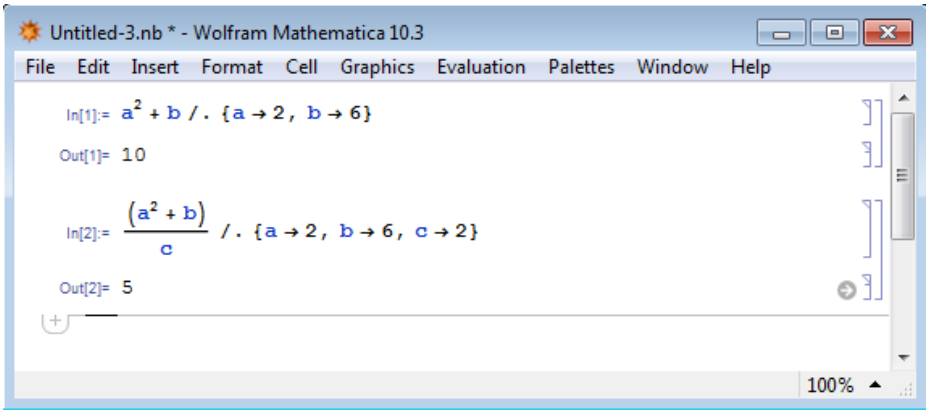
Let me explain what I am doing in each cell:

In[1]: Solve equation, get numerical solutions, and store all solutions in a variable (to be more precise, a list of variables) called "solutions"

In[2]: Extract the first solution (command "solutions[[1]]"), and put it in variable "x" using command "/.". For example, command $a^2 /. a \rightarrow 2$ means: evaluate "a^2" while replacing "a" with 2 (symbol → can be typed in Mathematica as "->"). Naturally, the answer is 4.



To do the same with several variables is not much more difficult:



A detailed information on the "/." command can be obtained here:

http://reference.wolfram.com/language/ref/ReplaceAll.html

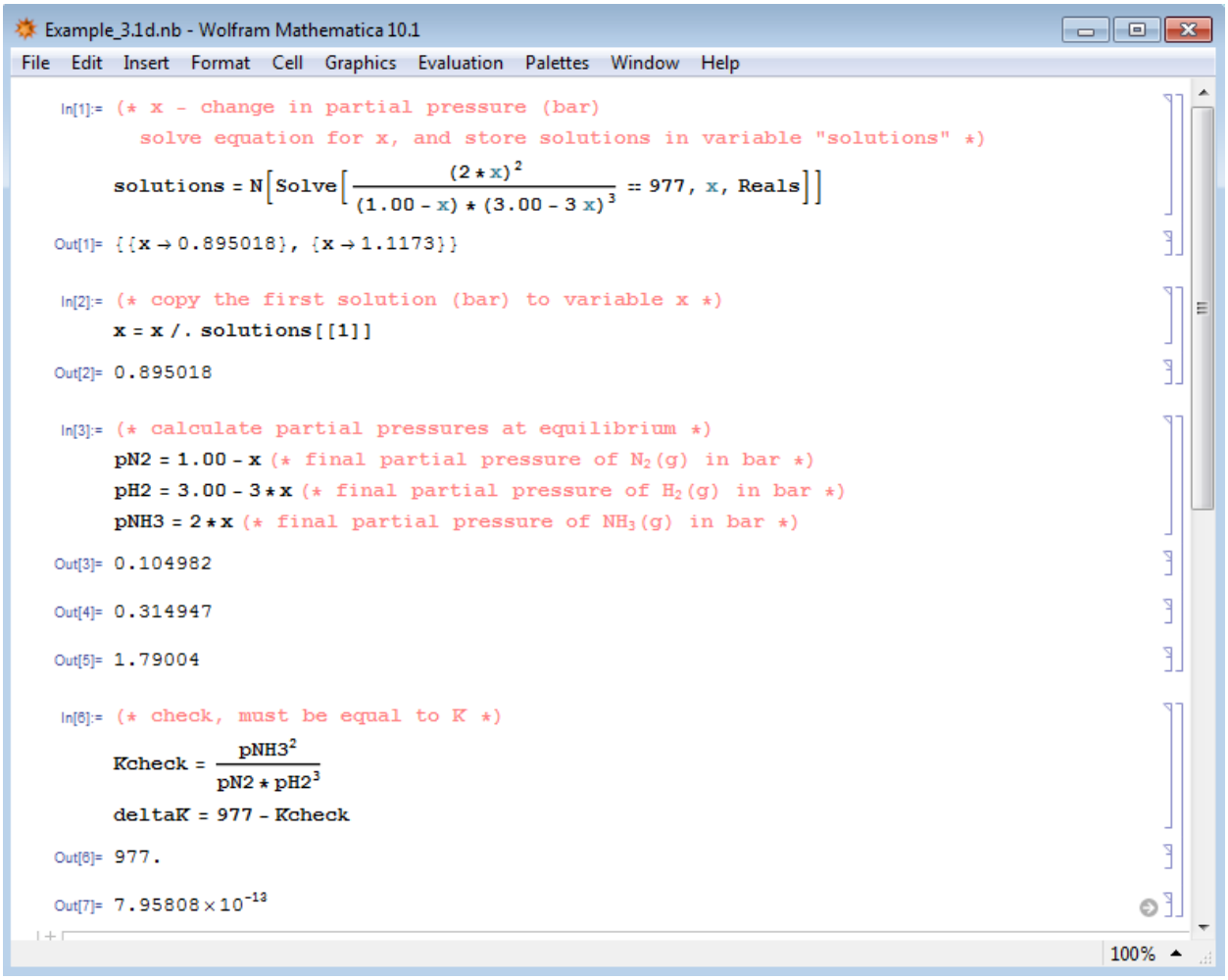
In[3]: Calculate the final partial pressures of all three gases and store them in variables pN2, pH2, and pNH3. Note how the names of variables are self-explanatory. **This is VERY important!**

Note on using Units in *Mathematica*: Each scientific calculation must include units. In general, *Mathematica* can use units:

https://reference.wolfram.com/language/guide/Units.html

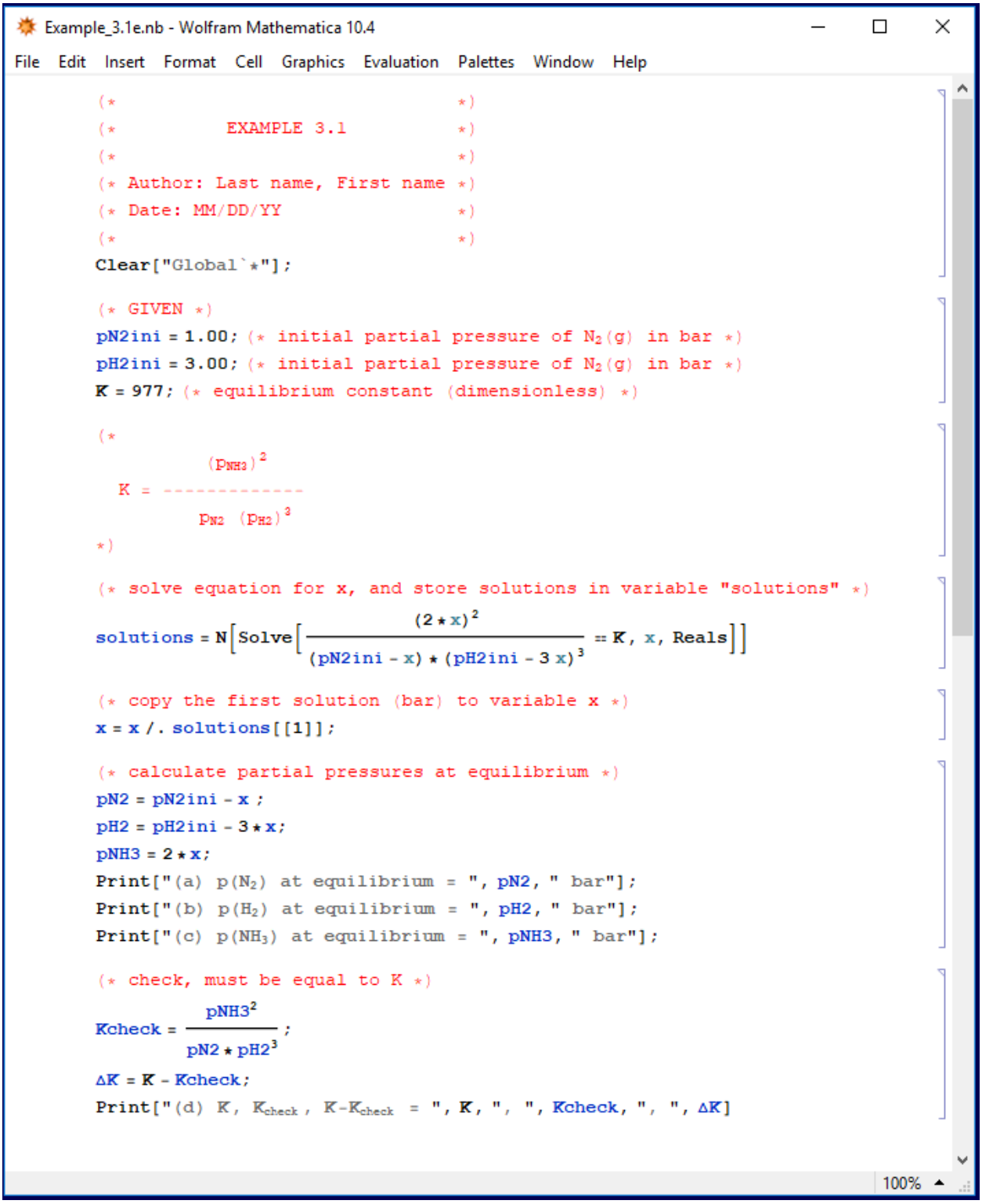
However, the interface is not very user-friendly. Thus, *for now*, it is acceptable to ignore units when doing calculations in *Mathematica*.

That said, in every single *Mathematica* notebook you absolutely must include comments that describe your calculation steps and the units of variables used. A comment in *Mathematica* is any text entered between symbols sequences "(*" and "*)"



Note how I have also added a check: the numerical value of the reaction quotient at equilibrium must be equal to the equilibrium constant (K = 977). The difference between the equilibrium constant that was calculated using the equilibrium partial pressures of the three gases and the given value is below 10^{-12}, which means that the final equilibrium partial pressures must be correct!

Now we tweak the processing of variables (to make the code more transparent and portable, and less cluttered), and add the project number, author name, date, the Clear["Global`*"] command, and user-friendly output of calculated properties. See for yourself:



Note how we suppressed the default *Mathematica* printout by terminating each line (except for the Solve[] statement) with a semicolon. This eliminates unnecessary printout from the final version of the code (we suggest you do so only in the final version of the code, after you have made sure that it produces correct results, and when you are ready to upload it to D2L).

Finally, once you are satisfied with your code, **go through the following checklist before submitting your *Mathematica* notebook to D2L:**

1) Make sure that the header of the Mathematica notebook includes the exercise number, your name, and the creation date. .

2) Make sure that all required answers are printed out using the Print[] statement.

3) Verify that each Print[] statement includes the appropriate label, i.e.
   Print["(a) …… "];
   Print["(b) …… "];
   Print["(c) …… "]; etc.

4) Suppress output from lines that do not contain printout of answers.

5) Remove all output cells (go to the "Cell" option of the notebook menu and select "Delete All Output")

6) Merge all input cells, which can be done using the following two steps:
   a. go to the "Edit" option of the notebook menu and select "Select All" (or simply use the "Ctrl+A" key combination)
   b. go to the "Cell" option of the notebook menu and select "Merge Cells" (or simply use the "Shift + Ctrl + M"

7) Save the notebook, and upload it to D2L.

Here is the final version of the Example 3.1 *Mathematica* notebook:

Example_3.1f.nb - Wolfram Mathematica 10.4

File Edit Insert Format Cell Graphics Evaluation Palettes Window Help

```
(*                                  *)
(*          EXAMPLE 3.1             *)
(*                                  *)
(* Author: Last name, First name    *)
(* Date: MM/DD/YY                   *)
(*                                  *)
Clear["Global`*"];

(* GIVEN *)
pN2ini = 1.00; (* initial partial pressure of N2(g) in bar *)
pH2ini = 3.00; (* initial partial pressure of N2(g) in bar *)
K = 977; (* equilibrium constant (dimensionless) *)

(*
          (pNH3)^2
  K = -------------
       pN2 (pH2)^3
*)
(* solve equation for x, and store solutions in variable "solutions" *)
solutions = N[Solve[(2*x)^2/((pN2ini - x)*(pH2ini - 3 x)^3) == K, x, Reals]]

(* copy the first solution (bar) to variable x *)
x = x /. solutions[[1]];

(* calculate partial pressures at equilibrium *)
pN2 = pN2ini - x ;
pH2 = pH2ini - 3*x;
pNH3 = 2*x;
Print["(a) p(N2) at equilibrium = ", pN2, " bar"];
Print["(b) p(H2) at equilibrium = ", pH2, " bar"];
Print["(c) p(NH3) at equilibrium = ", pNH3, " bar"];

(* check, must be equal to K *)
Kcheck = pNH3^2/(pN2*pH2^3);
ΔK = K - Kcheck;
Print["(d) K, Kcheck, K-Kcheck = ", K, ", ", Kcheck, ", ", ΔK]
```

100%

Note that your *Mathematica* notebook for this example should look very similar!

Perhaps now is a good time to demonstrate how to use the *Mathematica* Solve[] function

http://reference.wolfram.com/language/ref/Solve.html

to solve *two equations* for *two unknowns* simultaneously:

```
In[1]:= sols = Solve[a * x + y == 7 && b * x - y == 1, {x, y}]
```

$$\text{Out[1]}= \left\{\left\{x \to \frac{8}{a+b},\ y \to -\frac{a-7\,b}{a+b}\right\}\right\}$$

where *x* and *y* are the unknowns, and *a* and *b* are constants. Note how we used the "&&" symbol to specify the logical AND function. Suppose, we wish to save the solutions to some variables. Here is how we save the solution for *x* to a variable called *xs*:

```
In[2]:= xs = x /. sols[[1, 1]]
```

$$\text{Out[2]}= \frac{8}{a+b}$$

, and the same for *y*:

```
In[3]:= ys = y /. sols[[1, 2]]
```

$$\text{Out[3]}= -\frac{a-7\,b}{a+b}$$

If numerical values for *a* and *b* are known, *xs* and *ys* can be easily evaluated. For example,

```
In[4]:= xs /. {a → 2, b → 3}
```

$$\text{Out[4]}= \frac{8}{5}$$

```
In[5]:= ys /. {a → 2, b → 3}
```

$$\text{Out[5]}= \frac{19}{5}$$

Not bad!

Now that we have shown you how to perform this kind of calculations, it is your turn to do some.

Exercise 3.1. At a certain temperature, the equilibrium constant K_p for the gas-phase reaction

$$H_2(g) + I_2(g) \leftrightarrow 2HI(g)$$

is 40. Suppose, a gas mixture of H_2 and I_2 is prepared that has initial partial pressures (before reaction) of 1.8 bar for H_2 and 3.2 bar for I_2. Using the final notebook from Example 3.1 as a template, create a *Mathematica* notebook to calculate the equilibrium partial pressures (in bar) of all three gases: **(a)** H_2, **(b)** I_2, **(c)** HI, and **(d)** perform the check for K (K_p). Save the notebook as "Exercise_3.1.nb". **Warning! All calculation steps must be performed in your *Mathematica* notebook, and all items (a-d) must be clearly labelled in the Print[] functions, or the score for this exercise will be zero!**

Answers: **(a)** 0.17 bar **(b)** 1.6 bar **(c)** 3.3 bar **(d)** …

Exercise 3.2. At a certain temperature, the equilibrium constant K_p for the gas-phase reaction

$$CO(g) + Br_2(g) \leftrightarrow COBr_2(g)$$

is 0.0250. Suppose a 9:4 (in terms of partial pressures) reaction mixture of CO and Br_2 is heated in a closed vessel at a total pressure of 19.5 bar. Using the final notebook from Example 3.1 as a template, create a *Mathematica* notebook to calculate the equilibrium partial pressures (in bar) of all three gases: **(a)** CO, **(b)** Br_2, **(c)** $COBr_2$, **(d)** perform the check for K_p, and **(e)** calculate the percentage of $Br_2(g)$ that will be converted to $COBr_2(g)$ under these conditions. Save the notebook as "Exercise_3.2.nb". **Warning! All calculation steps must be performed in your *Mathematica* notebook, and all items (a-e) must be clearly labelled in the Print[] functions, or the score for this exercise will be zero!** *Hint:* https://en.wikipedia.org/wiki/Partial_pressure

Answers: **(a)** 12.1 bar **(b)** 4.61 bar **(c)** 1.39 bar **(d)** … **(e)** …

Exercise 3.3. It is sometimes convenient to express concentrations of species in a gas phase reaction in mol L^{-1}. For example, consider the gas-phase reaction

$$4\ HCl(g) + O_2(g) \leftrightarrow 2\ H_2O(g) + 2\ Cl_2(g)$$

that has reached equilibrium at the following concentrations of substances: $[H_2O(g)] = [Cl_2(g)] = 0.14$ mol L^{-1}, $[HCl(g)] = 0.20$ mol L^{-1}, and $[O_2(g)] = 0.32$ mol L^{-1}. Using the final notebook from Example 3.1 as a template, create a *Mathematica* notebook to calculate **(a)** the equilibrium constant K_c of the reaction. Considering that the initial reaction mixture did not contain any of the products, calculate the initial (before reaction) concentrations of **(b)** $O_2(g)$ (mol L^{-1}) and **(c)** HCl(g) (mol L^{-1}). Save the notebook as "Exercise_3.3.nb". **Warning! All calculation steps must be performed in your *Mathematica* notebook, and all items (a-c) must be clearly labelled in the Print[] functions, or the score for this exercise will be zero!**

Answers: **(a)** 0.75 **(b)** 0.39 mol L^{-1} **(c)** 0.48 mol L^{-1}

4. Differential Calculus

In this part of the lab, we are going to review some of the most fundamental physical concepts, and at the same time refresh your knowledge of Differential Calculus.

Displacement [9]**.** Pretty much everything around [and inside] us is in motion. Motion involves the displacement of an object from one place in space and time to another. Describing motion requires some convenient coordinate system and a specified origin. A frame of reference is a choice of coordinate axes that defines the starting point for measuring any quantity, an essential first step in solving virtually any problem in mechanics. The *displacement* Δx of an object is defined as its change in position and is given by

$$\Delta x = x_f - x_i \tag{11}$$

where x_i is the coordinate of the initial position of the object and x_f is the coordinate of the objects' final position (here, indices i and f stand for initial and final, respectively). The SI unit of displacement is a meter (m).

Because displacement has both a magnitude (size) and a direction, it's a vector quantity. In general, a vector quantity is characterized by having both a magnitude and a direction. By contrast, a scalar quantity has magnitude, but no direction. Scalar quantities such as mass and temperature are completely specified by a numeric value with appropriate units; no direction is involved.

Velocity and speed [9]. In everyday usage the terms *speed* and *velocity* are interchangeable. In physics, however, there's a clear distinction between them: speed is a scalar quantity, having only magnitude, whereas velocity is a vector quantity, having both magnitude and direction. However, the two quantities share the same SI unit: meter per second (m/s).

The *average speed* of an object over a given time interval is the length of the path it travels divided by the total elapsed time:

$$\text{Average speed} = \frac{\text{path length}}{\text{elapsed time}} \tag{12}$$

Unlike average speed, *average velocity* is a vector quantity, having both a magnitude and a direction. The average velocity $\bar{v}$ during a time interval Δt is the displacement, Δx, divided by the elapsed time, Δt:

$$\bar{v} = \frac{\Delta x}{\Delta t} = \frac{x_f - x_i}{t_f - t_i} \tag{13}$$

While the average speed is always non-negative, the average velocity of an object in one dimension can be either zero, or positive or negative, depending on the sign of the displacement. (The time interval Δt is always positive.)

The *instantaneous velocity v* is the limit of the average velocity as the time interval Δt becomes infinitesimally small:

$$v = \lim_{\Delta t \to 0} \frac{\Delta x}{\Delta t} \tag{14}$$

The notation $\lim_{\Delta t \to 0}$ means that the ratio $\Delta x / \Delta t$ is repeatedly evaluated for smaller and smaller time intervals Δt. Notice that the displacement Δx approaches zero as Δt approaches zero, so the ratio looks like 0/0 [10]. While this ratio may appear to be difficult to evaluate, it does have a specific value. As Δx and Δt become smaller and smaller, the ratio $\Delta x / \Delta t$ approaches a value equal to the slope of the line *tangent* to the *x*-versus-*t* curve, which is defined as the *instantaneous velocity* [Fig. 1].

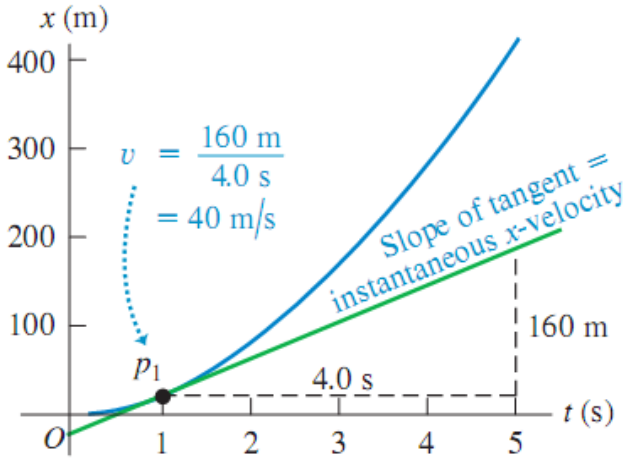


Figure 1. The instantaneous *x*-velocity *v* at any given point equals the slope of the tangent to the *x*-*t* curve at that point [11].

In the Calculus notation, this limit is called the *derivative* of *x* with respect to (w.r.t.) *t*, written as dx/dt [10]

$$v = \lim_{\Delta t \to 0} \frac{\Delta x}{\Delta t} = \frac{dx}{dt} \tag{15}$$

The *instantaneous speed* of an object, which is a scalar quantity, is defined as the magnitude of the instantaneous velocity. Like average speed, instantaneous speed (which we will usually call, simply, "speed") has no direction associated with it and hence carries no algebraic sign.

Acceleration, *a* [9]. The changing of an object's velocity with time is called *acceleration*. The average acceleration $\bar{a}$ during the time interval Δt is the change in velocity Δv divided by Δt:

$$\bar{a} = \frac{\Delta v}{\Delta t} = \frac{v_f - v_i}{t_f - t_i} \tag{16}$$

Acceleration is a vector quantity having dimensions of length divided by the time squared. In the SI system of units, acceleration has dimensions of meters per second per second ((m/s)/s, which is usually written as m s^{-2}). Acceleration is a vector quantity. For the case of motion in a straight line, the direction of the velocity of an object and the direction of its acceleration are related as follows: When the object's velocity and acceleration are in the same direction, the speed of the

object increases with time. When the object's velocity and acceleration are in opposite directions, the speed of the object decreases with time (this type of acceleration is often called *deceleration*).

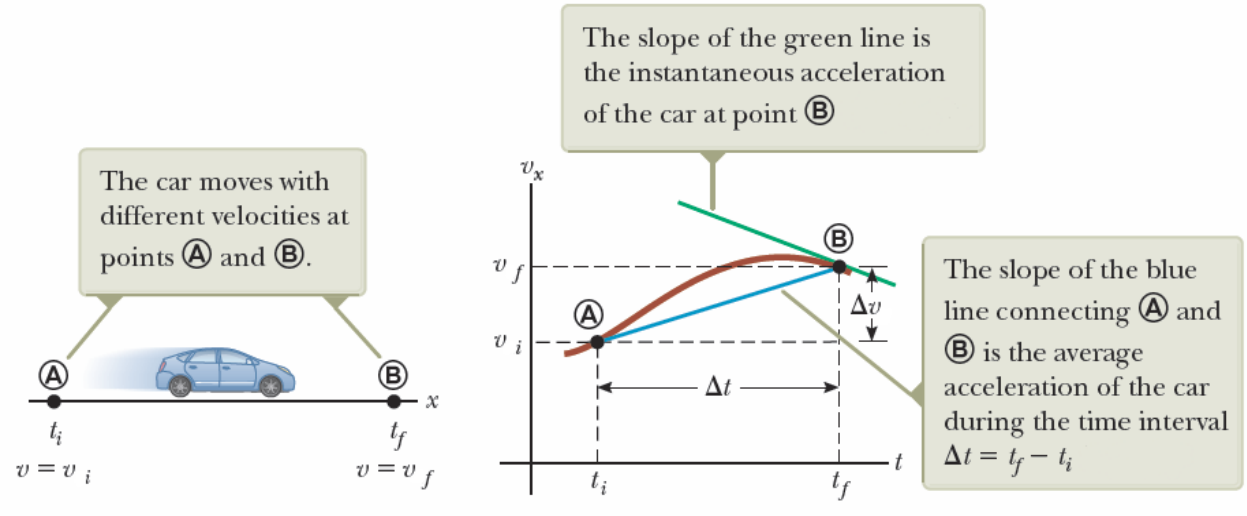


Figure 2. **(a)** A car, modeled as a particle, moving along the x axis from A to B, has velocity v_i at $t = t_i$ and velocity v_f at $t = t_f$. **(b)** Velocity–time graph (red-brown) for the particle moving in a straight line. [10]

The value of the average acceleration often differs in different time intervals, so it is useful to define the instantaneous acceleration, which is analogous to the instantaneous velocity discussed above. The *instantaneous acceleration* a is the limit of the average acceleration as the time interval Δt goes to zero:

$$a = \lim_{\Delta t \to 0} \frac{\Delta v}{\Delta t} \tag{17}$$

Here again, the notation $\lim_{\Delta t \to 0}$ means that the ratio $\Delta v / \Delta t$ is evaluated for smaller and smaller values of Δt. The closer Δt gets to zero, the closer the ratio gets to a fixed number, which is the instantaneous acceleration.

In the language of Calculus, instantaneous acceleration equals the derivative of the velocity with respect to time:

$$a = \lim_{\Delta t \to 0} \frac{\Delta v}{\Delta t} = \frac{dv}{dt} \tag{18}$$

which, by definition, is the slope of the velocity–time graph [10]. The slope of the green line in Figure 2b is equal to the instantaneous acceleration at point B. Notice that Figure 2b is a velocity–time graph, not a position–time graph like Figure 1. Therefore, we see that just as the velocity of a moving particle is the slope at a point on the particle's $\boldsymbol{x}$–t graph, the acceleration of a particle is the slope at a point on the particle's $\boldsymbol{v}$–t graph. One can interpret the derivative of the velocity with respect to time as the time rate of change of velocity. If a is positive, the acceleration is in the positive x direction; if a is negative, the acceleration is in the negative x direction (often called *deceleration*) [10].

Because $v = dx/dt$, the acceleration can also be written as

$$a = \frac{dv}{dt} = \frac{d}{dt}(v) = \frac{d}{dt}\left(\frac{dx}{dt}\right) = \frac{d^2x}{dt^2} \tag{19}$$

That is, in one-dimensional motion, the acceleration also equals the second derivative of x with respect to time [10].

Figure 3 [10] illustrates how an acceleration–time graph is related to a velocity–time graph. The acceleration at any time is the slope of the velocity–time graph at that time. Positive values of acceleration correspond to those points in Figure 3 where the velocity is increasing in the positive x direction. The acceleration reaches a maximum at time t_A, when the slope of the velocity–time graph is a maximum. The acceleration then goes to zero at time t_B, when the velocity is a maximum (that is, when the slope of the v–t graph is zero). The acceleration is negative when the velocity is decreasing in the positive x direction, and it reaches its most negative value at time t_C [10].

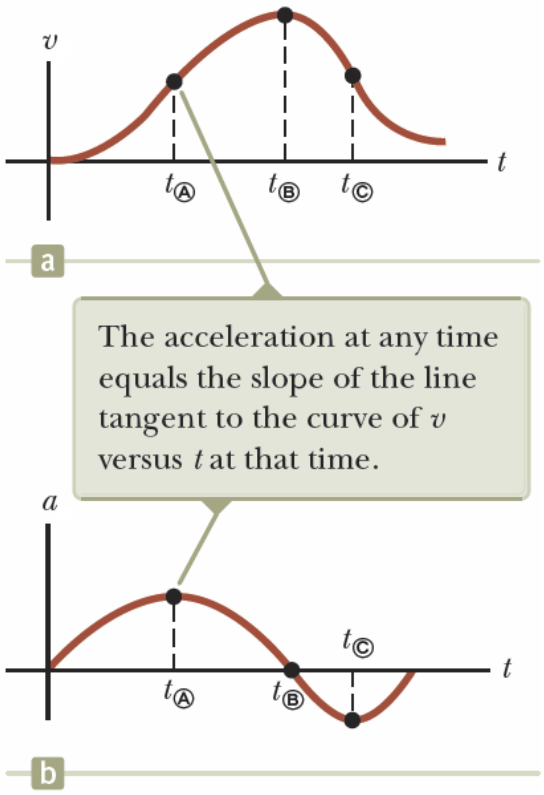


Figure 3 [10]. (a) The velocity–time graph for a particle moving along the x axis. (b) The instantaneous acceleration can be obtained from the velocity–time graph.

Example 4.1. Consider a particle moving along the x axis according to the equation

$$x(t) = 2.00 + 3.00t - 1.00t^2 \tag{20}$$

where x is in meters (m) and t is in seconds (s). Using *Mathematica*, derive general expressions for **(a)** $v(t)$ and **(b)** $a(t)$. At $t = 3.00$ s, find **(c)** the position of the particle, **(d)** its velocity, and **(e)** its acceleration. Plot **(f)** x vs t, **(g)** v vs t , and **(h)** a vs t for $t = 0…5$ seconds.

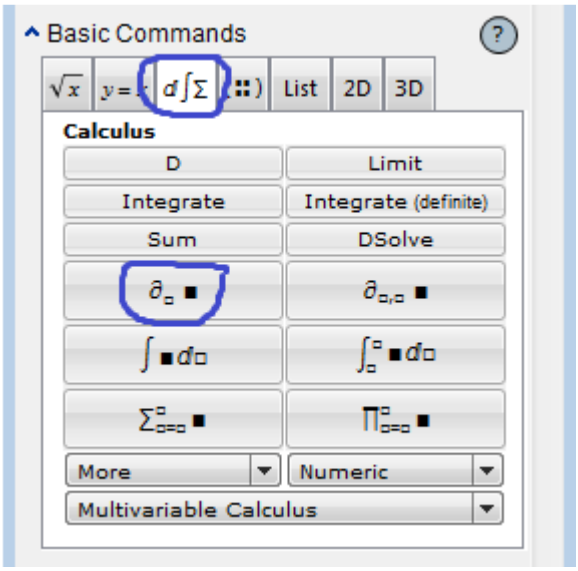
Open a new *Mathematica* notebook, and call it, say, "Example_4.1.nb". After typing in the project number, define function $x(t)$ given above:

```
In[1]:= (*                *)
        (*  EXAMPLE 4.1   *)
        (*                *)
        Clear["Global`*"];

In[2]:= (* define x(t) *)
        x = 2 + 3 * t - 1 * t^2;
        Print["x(t) = ", x, " m"];

        x(t) = 2 + 3 t - t^2 m
```

First, we derive a general expression for velocity at any t, i.e. find $v(t)$. Recall that $v(t) = \frac{dx(t)}{dt}$, so we need to differentiate $x(t)$ with respect to (w.r.t.) t. In *Mathematica*, differentiation can be done using the command D[]

https://reference.wolfram.com/language/ref/D.html?q=D

or using the $\partial_{\square}\,\blacksquare$ button in the $d\int\Sigma$ section of the "Basic Math Assistant" palette:



Here is my code that demonstrates how to use both options (of course, you should use only one, which makes more sense!):

```
In[2]:= (* define x(t) *)
        x = 2 + 3 * t - 1 * t^2;
        Print["x(t) = ", x, " m"];

        x(t) = 2 + 3 t - t^2 m

In[4]:= (* (a) velocity, v = dx/dt *)
        v = D[x, t]
        v = ∂t x
        Print["(a) v(t) = ", v, " m/s"];

Out[4]= 3 - 2 t

Out[5]= 3 - 2 t

        (a) v(t) = 3 - 2 t m/s
```

As expected, the two differentiation commands give the same result. Thus, $v(t) = 3 - 2t$. Note how I used the Print[] statement to print out answer for **(a)**.

How about the acceleration? Well, according to equation (18), we get $a(t)$ by differentiating $v(t)$ w.r.t. t once, or differentiate $x(t)$ w.r.t t twice: $a(t) = \frac{dv(t)}{dt} = \frac{d^2x(t)}{dt^2}$. Both methods can be easily implemented *Mathematica* (of course, you should use only one, whichever makes more sense!):

```
In[7]:= (* (b) acceleration, a = dv/dt  or a = d²x/dt² *)
        a = D[v, t]
        a = ∂t v
        a = D[x, t, t]
        a = ∂t,t x
        Print["(b) a(t) = ", a, " m/s²"];
Out[7]= -2
Out[8]= -2
Out[9]= -2
Out[10]= -2
        (b) a(t) = -2 m/s²
```

Again, as expected, a is the same regardless of the method used. Note that because a does not depend on t, we are dealing with a case of the *constant acceleration*. Thus, $a = -2$ m/s^2 (units come directly from the question) at any time t. To summarize,

$x(t)$ (in m) $= 2.00 + 3.00t - 1.00t^2$, where t is in seconds

$v(t)$ (in m/s) $= 3.00 - 2.00t$, where t is in seconds

a (in m/s^2) $= -2.00$ at any t

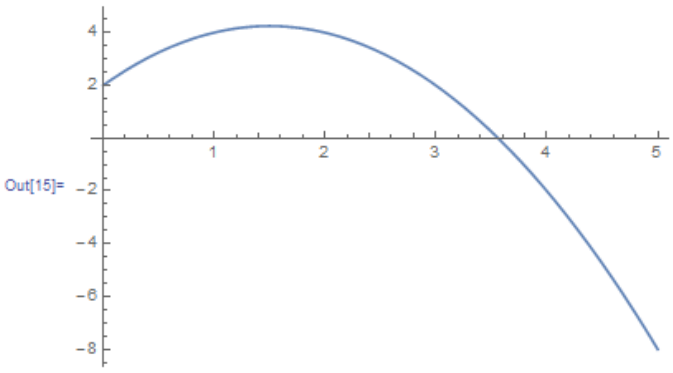
Finding x, v, and, especially, a (duh!) at $t = 3$ s is very easy:

```
In[12]:= (* (c,d,e) finding x, v, and a at t=3 s is easy *)
         Print["(c) x(t=3s) = ", x /. t → 3, " m"];
         Print["(d) v(t=3s) = ", v /. t → 3, " m/s"];
         Print["(e) a(t=3s) = ", a /. t → 3, " m/s²"];
         (* of course, a is the same for all t ! *)
         (c) x(t=3s) = 2 m
         (d) v(t=3s) = -3 m/s
         (e) a(t=3s) = -2 m/s²
```

Again, note how I use the Print[] statement to create a user-friendly output. Also note that acceleration, a, is of course constant at -2 m/s^2.

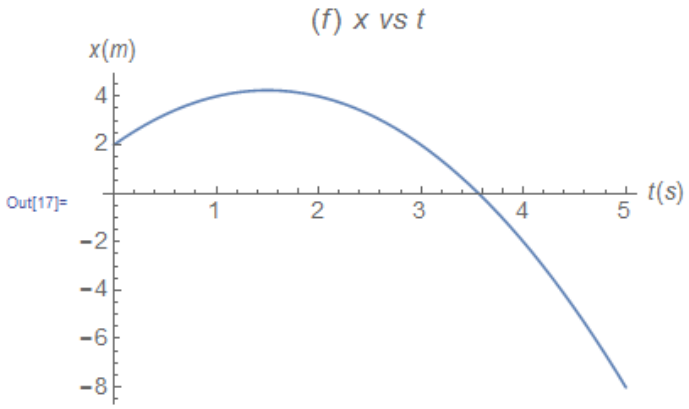
Now the fun part. We can easily plot $x(t)$, $v(t)$, and $a(t)$ in *Mathematica* using the Plot[] function. In the following, I show how to do it for $t = 0\ldots5$ s:

```
In[15]:= Plot[x, {t, 0, 5}]
```



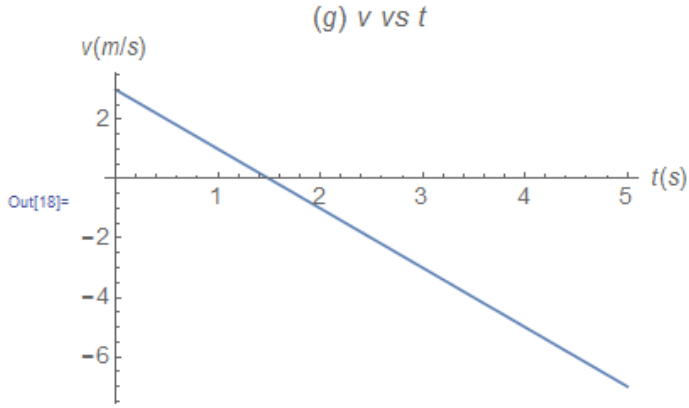
While this is nice plot, it is missing labels of the *x*- and *y*- axes, and the plot label (legend). Those can be added to any Mathematica plot as shown below:

```
In[16]:= SetOptions[Plot, BaseStyle → {FontSize → 14}];
In[17]:= Plot[x, {t, 0, 5}, AxesLabel → {Style["t(s)", Italic], Style["x(m)", Italic]},
          PlotLabel → Style["(f) x vs t", Italic]]
```



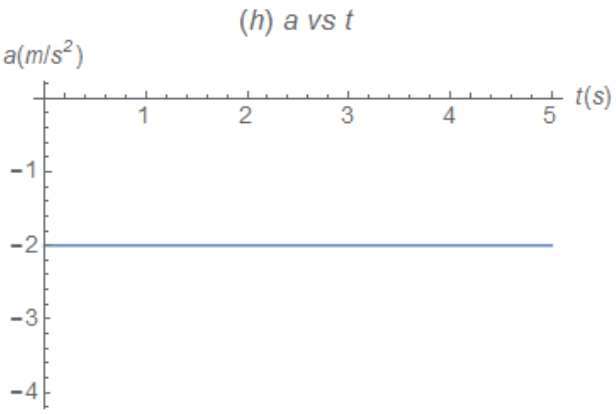
Much better! Note that I have used *Mathematica*'s function SetOptions[Plot, BaseStyle → {FontSize → 14}] to increase the font size in the plots. Now, let's plot $v(t)$ and $a(t)$:

```
In[18]:= Plot[v, {t, 0, 5}, AxesLabel → {Style["t(s)", Italic], Style["v(m/s)", Italic]},
          PlotLabel → Style["(g) v vs t", Italic]]
```



```
In[19]:= Plot[a, {t, 0, 5}, AxesLabel → {Style["t(s)", Italic], Style["a(m/s²)", Italic]},
          PlotLabel → Style["(h) a vs t", Italic]]
```

Out[19]=



I hope that it is obvious that because velocity changes uniformly with time (i.e. a *v*-*t* plot is a straight line), acceleration is constant, and vice versa.

Finally, it may be worthwhile to show how easy it is to plot two or more equations on the same graph in *Mathematica*. Consider the following three functions

$$y_1 = 1 + 2x, \qquad y_2 = 10 - 3x, \qquad y_3 = -5 + x^2$$

that we wish to plot on the same graph for $0 \le x \le 5$. First, we define the functions:

```
In[1]:= y1 = 1 + 2 * x
        y2 = 10 - 3 * x
        y3 = -5 + x^2
Out[1]= 1 + 2 x
Out[2]= 10 - 3 x
Out[3]= -5 + x^2
```

Then, we define the minimum and maximum values for *x*, and the font size for the plot:

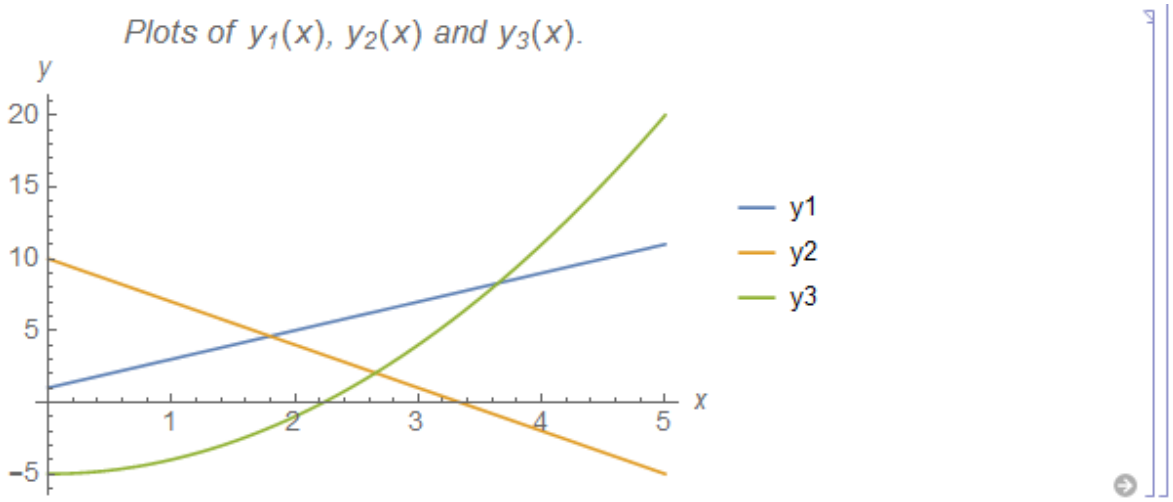
```
In[4]:= SetOptions[Plot, BaseStyle → {FontSize → 14}];
        xmin = 0; xmax = 5;
```

Because we terminated each command here with a semicolon, there is no output. Finally, we plot all three functions on the same plot using the same command Plot[] which was used to plot a single function:

```
In[6]:= Plot[{y1, y2, y3}, {x, xmin, xmax},
         AxesLabel → {Style["x", Italic], Style["y", Italic]},
         PlotLabel → Style["Plots of y1(x), y2(x) and y3(x).", Italic],
         PlotLegends -> "Expressions"]
```

Note how we included a list of functions that we want to plot in the curly brackets [thus forming as so-called "list" in *Mathematica*]. Also note that in addition to already familiar options AxesLabel and PlotLabel, we added the PlotLegends option that, as you can guess from its name, adds a legend to the graph.

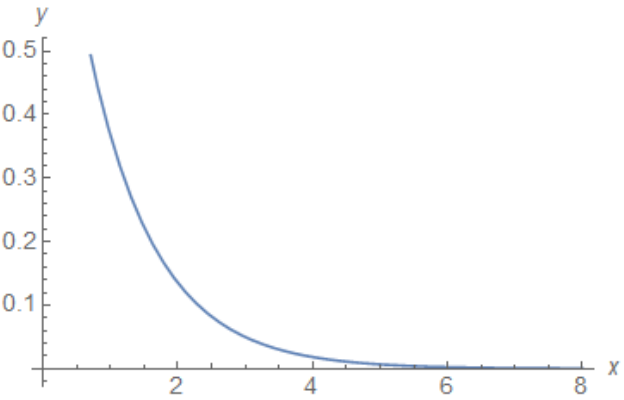
Out[6]=



Sometimes *Mathematica* does not plot the entire range for *x* and/or *y* variable(s). For example,

```
In[1]:= SetOptions[Plot, BaseStyle → {FontSize → 14}];
        y = Exp[-x];
        Plot[y, {x, 0, 8}, AxesLabel → {Style["x", Italic], Style["y", Italic]}]
```

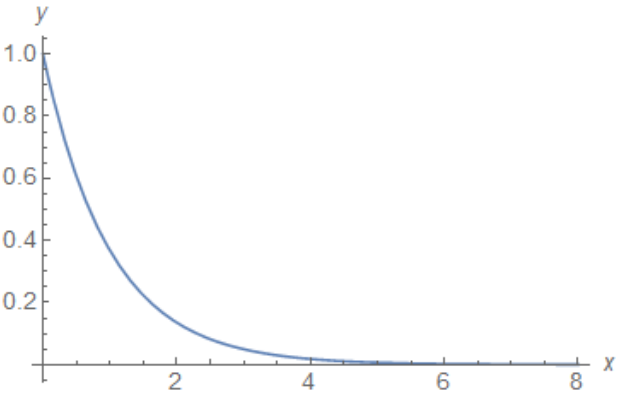
Out[3]=



To fix that, simply add the PlotRange→Full option to the Plot[] command:

```
In[1]:= SetOptions[Plot, BaseStyle → {FontSize → 14}];
        y = Exp[-x];
        Plot[y, {x, 0, 8}, AxesLabel → {Style["x", Italic], Style["y", Italic]}, PlotRange → Full]
```

Out[3]=



Ok, now it is your turn to perform a couple of calculations in *Mathematica*.

Exercise 4.1. An object moves along the *x* axis according to the equation

$$x(t) = \mathcal{A} + \mathcal{B}t + \mathcal{C}t^2 + \mathcal{D}t^3$$

where *x* is in cm, *t* is in seconds, and $\mathcal{A}$, $\mathcal{B}$, $\mathcal{C}$, and $\mathcal{D}$ are some constants. Using the notebook from Example 4.1 as a template, write a *Mathematica* code to derive expressions for **(a)** velocity and **(b)** acceleration as a function of time *t* and general coefficients $\mathcal{A}$, $\mathcal{B}$, $\mathcal{C}$, and $\mathcal{D}$. If $\mathcal{A} = 0.1$ cm, $\mathcal{B} = 0.04$ cm s^{-1}, $\mathcal{C} = -0.3$ cm s^{-2} and $\mathcal{D} = 0.05$ cm s^{-3}, calculate **(c)** the distance, **(d)** velocity, and **(e)** acceleration at *t* = 4.0 s. Plot **(f)** *x* vs *t*, **(g)** *v* vs *t*, and **(h)** *a* vs *t* for *t* = 0… 7 s. Save the notebook as "Exercise_4.1.nb". **Warning! All calculation steps must be performed in your *Mathematica* notebook, and all items (a-h) must be clearly labelled in the Print[] functions, or the score for this exercise will be zero!**

Answers: **(a)** $\mathcal{B} + 2\mathcal{C}t + 3\mathcal{D}t^2$
(b) $2\mathcal{C} + 6\mathcal{D}t$
(c) -1.3 cm **(d)** 0.04 cm s^{-1} **(e)** 0.6 cm s^{-2}
(f,g,h) …

Exercise 4.2. An object moves along the x axis according to the equation

$$x(t) = \alpha(2t^3 - \beta) - \gamma \sin\left(\frac{2\pi}{3}t\right)$$

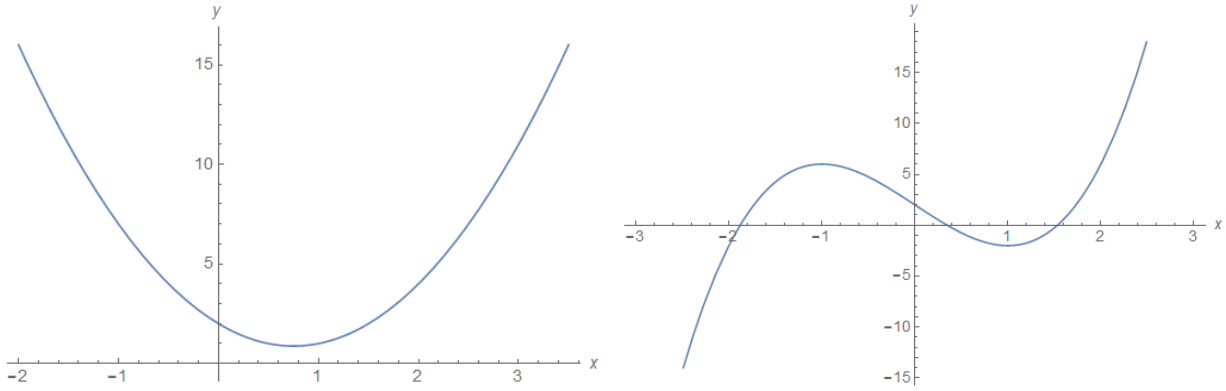
where *x* is in meters, *t* is in seconds, and α, β and γ are some constants. Using the notebook from Example 4.1 as a template, write a *Mathematica* code to derive expressions for **(a)** velocity and **(b)** acceleration as a function of time *t* and general coefficients α, β and γ. If $\alpha = 0.3$ m s^{-3}, $\beta = 0.1$ s^3 and $\gamma = 0.8$ m, calculate **(c)** the distance, **(d)** velocity, and **(e)** acceleration at *t* = 2.0 s. Plot **(f)** *x* vs *t*, **(g)** *v* vs *t*, and **(h)** *a* vs *t* for *t* = 0… 3 s. Save the notebook as "Exercise_4.2.nb". **Warning! All calculation steps must be performed in your *Mathematica* notebook, and all items (a-h) must be clearly labelled in the Print[] functions, or the score for this exercise will be zero!**

Answers: **(a)** $6\alpha t^2 - \frac{2}{3}\pi\gamma \cos\left(\frac{2\pi t}{3}\right)$
(b) $12\alpha t + \frac{4}{9}\pi^2\gamma \sin\left(\frac{2\pi}{3}t\right)$
(c) 5.5 m **(d)** 8.0 m s^{-1} **(e)** 4.2 m s^{-2}
(f,g,h) …

5. Geometrical interpretation of derivatives [12]

In the previous section we introduced the idea that in the limiting case the derivative represents an instantaneous rate of change of two variables [12]. Hence, if, for example, $y = f(x)$ is plotted on a two-dimensional Cartesian coordinate system, then dy/dx is the slope of the curve at any point (x, y) on the curve. With exception of the function $y(x) =$ constant, functions either increase or decrease as the value of x increases. By looking at the derivative (or slope) evaluated at the point (x, y), we can determine whether the function $f(x)$ is increasing or decreasing as x increases without having to graph the function. If dy/dx is positive, then $f(x)$ increases as x increases. If dy/dx is negative, then $f(x)$ decreases as x increases [12].

Certain functions, such as parabolas (Fig. 4a), or functions of higher order, such as cubic functions (Fig. 4b), have either a maximum or a minimum value, or both [12]. Differential calculus can be used to help us determine the point or points along the curve where maxima or minima occur. Since the slope of the curve must be zero at these points, the first derivative also must be zero [12].



a) $y = 2x^2 - 3x + 2$ b) $y = 2x^3 - 6x + 2$

Figure 4 [12]. Sample graphs of (a) quadratic and (b) cubic functions.

For example, the parabola shown in Fig. 4a is described by the equation [12]

$$y = 2x^2 - 3x + 2 \tag{21}$$

Taking the first derivative gives [12]

$$\frac{dy}{dx} = 4x - 3 \tag{22}$$

Setting the first derivative equal to zero and solving for x, we have [12]

$$4x - 3 = 0 \quad \text{or} \quad x = \frac{3}{4} \tag{23}$$

Substituting $x = \frac{3}{4}$ into the equation $y(x)$, yields $y = 0.875$, which gives the minimum point on the curve [12]. In *Mathematica*, the same calculation can be performed as follows:

```
In[1]:= y = 2 * x^2 - 3 * x + 2
        dydx = D[y, x]
        Solve[dydx == 0, x]
Out[1]= 2 - 3 x + 2 x^2
Out[2]= -3 + 4 x
Out[3]= {{x → 3/4}}
```

In order to determine whether the curve is a maximum or a minimum at a point without actually plotting the curve, we can substitute values for x that are greater than the value at the point where the derivative is zero and the values for x that are smaller than the value at the point where the derivative is zero (a.k.a. *stationary point*). Then we note the behavior of y. A simpler way to test whether the function is a maximum or a minimum is to look at the *second* derivative of the function [12].

If $\frac{d^2y}{dx^2} < 0$, then the function is a maximum.

If $\frac{d^2y}{dx^2} > 0$, then the function is a minimum.

If $\frac{d^2y}{dx^2} = 0$, then the function is at a *point of inflection* – a point on the curve where the curve changes from one that exhibits a maximum at some point (i.e., is "concave downward") to one that exhibits a minimum at some point (i.e., is "concave downward"), or vice versa; note that inflections points may be stationary points (as shown in Fig. 5), but are *not* local maxima or local minima.

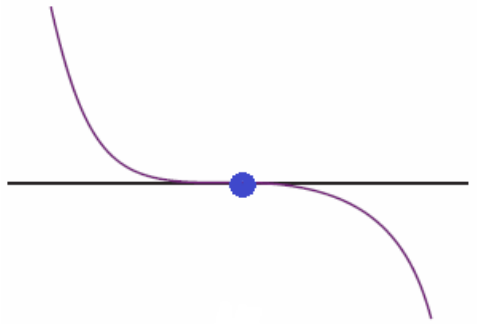


Figure 5 [13]. Example of an inflection point.

Thus, in the parabola example, the second derivative at x = ¾ (and, incidentally, everywhere else) is

```
In[4]:= y = 2 * x^2 - 3 * x + 2
        d2ydx2 = D[y, x, x]
Out[4]= 2 - 3 x + 2 x^2
Out[5]= 4
```

This makes sense because any parabola is "concave downward" everywhere...
Now, consider the cubic function shown in Fig. 4b [12]:

$$y = 2x^3 - 6x + 2 \tag{24}$$

Taking the first derivative and setting it equal to zero yields [12]

$$\frac{dy}{dx} = 6x^2 - 6 = 0\,; \quad x^2 - 1 = 0\,; \quad x = \pm 1 \tag{25}$$

which indicates that there is maximum or a minimum at x = +1 and x = -1 [12]. Taking the second derivative of this cubic equation gives

$$\frac{d^2y}{dx^2} = 12x \tag{26}$$

For x = +1, $d^2y/dx^2 = 12$, which indicates that the curve is a minimum at this point. For x = -1, $d^2y/dx^2 = -12$, which indicates that the curve is a maximum at this point.

In *Mathematica*, these calculations can be done as follows:

```
In[1]:= y = 2 * x^3 - 6 * x + 2
        dydx = D[y, x]
        sol = Solve[dydx == 0, x]
        d2ydx2 = D[y, x, x]
        x1 = x /. sol[[1]]
        d2ydx2 /. x → x1
        x2 = x /. sol[[2]]
        d2ydx2 /. x → x2
Out[1]= 2 - 6 x + 2 x^3
Out[2]= -6 + 6 x^2
Out[3]= {{x → -1}, {x → 1}}
Out[4]= 12 x
Out[5]= -1
Out[6]= -12
Out[7]= 1
Out[8]= 12
```

Now you may wonder at which point the sign of the curvature changes from negative (maximum) to positive (minimum). Well, clearly it should be at a point where the second derivative vanishes, i.e.

$$\frac{d^2y}{dx^2} = 12x = 0 \tag{27}$$

It follows that $x = 0$ is an inflection point of this function. *Mathematica* can find this point easily:

```
In[9]:= Solve[d2ydx2 == 0, x]

Out[9]= {{x → 0}}
```

Example 5.1. The mathematical expression that tells us the probability, P, that the molecules in a sample of gas have a speed that lies in a particular range at any instant is called the *distribution of molecular speeds* [14]. The precise form of the distribution was worked out by James Clerk Maxwell (1831-1879) towards the end of the nineteenth century, and his expression is known as the *Maxwell distribution of speeds* [14]. A sample Maxwell distribution of molecular speeds for the argon gas at 500 K is shown in Fig 6 [15].

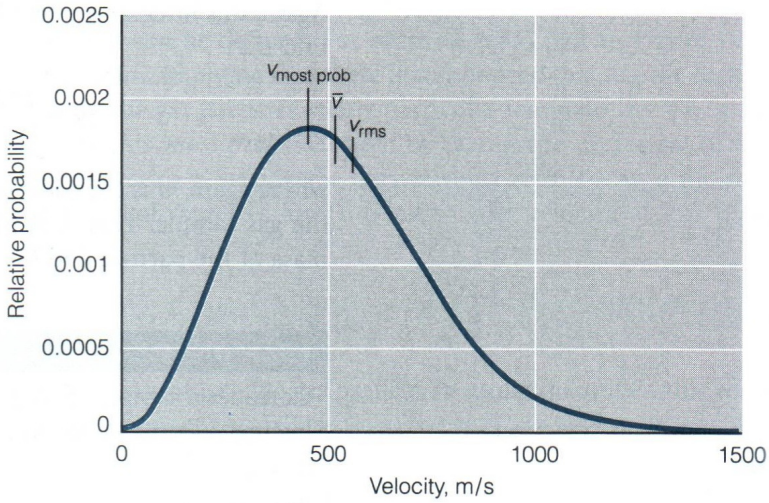


Figure 6 [15]. The Maxwell distribution of molecular speeds for the argon gas at 500 K.

According to Maxwell's equation, the probability $P(v, v+\Delta v)$ that the molecules have a speed in a narrow range between v and $v+\Delta v$ (for example, between 300 m/s and 301 m/s, corresponding to v = 300 m/s and Δv = 1 m/s) is [14]

$$P(v, v+\Delta v) = \rho(v)\Delta v \quad \text{with} \quad \rho(v) = 4\pi\left(\frac{M}{2\pi RT}\right)^{3/2} v^2 e^{-\frac{Mv^2}{2RT}} \tag{28}$$

where M is the molar mass of the gas, R is the gas constant, and T is temperature. We wish to derive an expression for the most probable speed (identified as $v_{\text{most prob}}$ in Fig. 6). Note that the most probable speed is the speed at which the probability is a maximum. Thus, we must maximize the function

$$\rho(v) = 4\pi\left(\frac{M}{2\pi RT}\right)^{3/2} v^2 e^{-\frac{Mv^2}{2RT}} \tag{29}$$

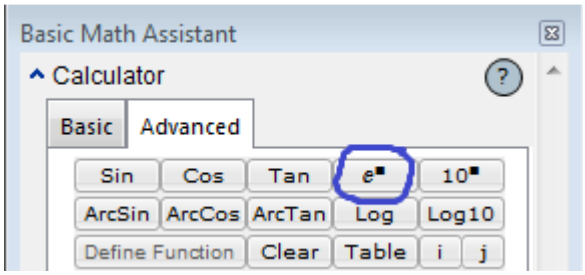
with respect to v, set the derivative to zero, and solve for v. This is a simple task for *Mathematica*. We always start with the project title (and so should you!). Then, we define function $\rho(v)$:

```
In[1]:= (*                *)
        (*  EXAMPLE 5.1   *)
        (*                *)
        Clear["Global`*"];

In[2]:= (* define ρ(v) *)
        ρ = 4 * π * (M / (2 * π * R * T))^(3/2) * v^2 * Exp[-(M * v^2) / (2 * R * T)];
        Print["function ρ(v) = ", ρ];

        function ρ(v) = e^(-Mv^2/(2RT)) Sqrt[2/π] (M/(RT))^(3/2) v^2
```

A couple of notes. Instructions on how to enter Greek letters in *Mathematica* can be found at

http://reference.wolfram.com/language/tutorial/EnteringGreekLetters.html

Of course, constants, such as π, i, and e can also be entered using the "Basic Math Assistant" palette. Alternatively, you can use the "Special Characters" palette under the "Palettes" menu. Second, an exponential can be taken using the Exp[] function, or using the $e^{\blacksquare}$ button in the "Basic Math Assistant" palette



Naturally, the result is the same:

```
In[1]:= (* define ρ(v) *)
        ρ = 4 * π * (M / (2 * π * R * T))^(3/2) * v^2 * e^(-(M*v^2)/(2*R*T));
        Print["function ρ(v) = ", ρ];

        function ρ(v) = e^(-Mv^2/(2RT)) Sqrt[2/π] (M/(RT))^(3/2) v^2
```

Now, we calculate the derivative of $\rho(v)$ w.r.t. v, $\frac{d\rho(v)}{dv}$, and store the result in variable dρdv:

```
In[4]:= (* evaluate dρ/dv and copy the result to variable dρdv *)
        dρdv = D[ρ, v];
        Print["dρ/dv = ", dρdv];
```

$$\frac{d\rho}{dv} = 2\, e^{-\frac{M v^2}{2RT}} \sqrt{\frac{2}{\pi}} \left(\frac{M}{RT}\right)^{3/2} v - \frac{e^{-\frac{M v^2}{2RT}} M \sqrt{\frac{2}{\pi}} \left(\frac{M}{RT}\right)^{3/2} v^3}{RT}$$

We can also ask *Mathematica* to try to simplify the result [to make it more "visually appealing"] using command Simplify[] or, even better, FullSimplify[]:

```
In[6]:= (* if you wish, you can somewhat simply the expression for dρ/dv *)
        dρdv = FullSimplify[dρdv];
        Print["dρ/dv simplified = ", dρdv];
```

$$\frac{d\rho}{dv} \text{ simplified} = -\frac{e^{-\frac{M v^2}{2RT}} \sqrt{\frac{2}{\pi}} \left(\frac{M}{RT}\right)^{5/2} v \left(-2RT + M v^2\right)}{M}$$

We set the derivative (stored in variable dρdv) to zero, and use the Solve[] function to solve the equation for *v*:

```
In[8]:= (* set dρ/dv to zero, and solve for v *)
        sols = Solve[dρdv == 0, v]
```

$$\text{Out[8]}= \left\{\{v \to 0\}, \left\{v \to -\frac{\sqrt{2}\,\sqrt{R}\,\sqrt{T}}{\sqrt{M}}\right\}, \left\{v \to \frac{\sqrt{2}\,\sqrt{R}\,\sqrt{T}}{\sqrt{M}}\right\}\right\}$$

There are three solutions. While all of them are mathematically correct, the first two solutions are **not** physically meaningful (I hope you see why…). Thus, the only acceptable solution is number three, which we then copy to variable vmostprob, and print it out:

```
In[9]:= (* copy the acceptable result to variable vmostprob *)
        vmostprob = v /. sols[[3]];
        Print["most probable speed = ", vmostprob];
```

$$\text{most probable speed} = \frac{\sqrt{2}\,\sqrt{R}\,\sqrt{T}}{\sqrt{M}}$$

Now, compare the expression derived in *Mathematica* to that given in [almost] every single Physical Chemistry textbook [13,15]:

$$v_{\text{most prob}} = \left(\frac{2RT}{M}\right)^{1/2} \tag{30}$$

Not bad, eh?!

Well, it is time for you to differentiate a couple of simple functions in *Mathematica*.

Exercise 5.1. An object is oscillating along the x-axis according to the following equation

$$x(t) = 0.1 \sin\left(\frac{\pi}{8}t + \frac{\pi}{4}\right) \tag{31}$$

where t is time. Using the final notebook from Example 5.1 as a template, create a *Mathematica* notebook that determines the maximum and minimum x-coordinate of the object. Save the notebook as "Exercise_5.1.nb". **Your *Mathematica* notebook must include the following items in the printout, all clearly labelled using the Print[] function**:

(a) define function $x(t)$,

(b) derive formula for the first derivative of $x(t)$ with respect to (w.r.t.) t, dx/dt (*recall that the first derivative of position w.r.t. time is a velocity!*)

(c) locate the stationary points of $x(t)$ by determining values of t at which $dx/dt = 0$; you should get two points: $t_1 = -6$ and $t_2 = 2$;
note that since this function is periodic, it is sufficient to consider $-10 \le t \le 6$;
depending on the version of *Mathematica* that you use, you may have to explicitly define the region for t in the Solve[] command: Solve[…&&$t \ge -10$&& $t \le 6$], where symbol "&&" is the *Mathematica* logical AND function:
http://reference.wolfram.com/language/ref/And.html

(d) derive formula for the second derivative of $x(t)$ w.r.t. t, d^2x/dt^2 (*recall that the second derivative of position w.r.t. time is an acceleration!*),

(e-f) for each point found at step **(c)**, calculate numerical value of the second derivative, d^2x/dt^2, and decide whether $x(t)$ at this point is a minimum, maximum, or an inflection point (t_1 – minimum, t_2 – maximum),

(g) plot $x(t)$ vs t for $-12 \le t \le 8$ to confirm your conclusions for **(e-f).**

Warning! All calculation steps must be performed in your *Mathematica* notebook, or the score for this exercise will be zero!

Exercise 5.2. Using the final notebook from Example 5.1 as a template, create a *Mathematica* notebook that determines the maximum velocity in the positive and negative x-directions for the object whose position is given by the function $x(t)$ defined in Exercise 5.1:

$$x(t) = 0.1 \sin\left(\frac{\pi}{8}t + \frac{\pi}{4}\right) \tag{32}$$

Save the notebook as "Exercise_5.2.nb". **Your *Mathematica* notebook must include the following items in the printout, all clearly labelled in the Print[] functions**:

(a) define function $x(t)$,

(b) differentiate $x(t)$ w.r.t. t to calculate $v(t)$,

(c) derive formula for the first derivative of $v(t)$ with respect to (w.r.t.) t, dv/dt (*recall that the first derivative of velocity is an acceleration!*)

(d) locate the stationary points of $v(t)$ by determining values of t at which $dv/dt = 0$ (*i.e. when the acceleration is zero!*);

note that since this function is periodic, it is sufficient to consider $-10 \leq t \leq 6$;

depending on the version of *Mathematica* that you use, you may have to explicitly define the region for t in the Solve[] command: Solve[…&&$t \geq -10$&& $t \leq 6$], where symbol "&&" is the *Mathematica* logical AND function:

http://reference.wolfram.com/language/ref/And.html

you should get three points: $t_1 = -10$, $t_2 = -2$ and $t_3 = 6$;

(e) derive formula for the second derivative of $v(t)$ w.r.t. t, d^2v/dt^2 (*this would be the first derivative of acceleration!*),

(f,g,h) for each point found at step **(d)**, calculate numerical value of the second derivative, d^2v/dt^2, and decide whether $v(t)$ at this point is a minimum, maximum, or an inflection point (t_1 and t_3 – minima, t_2 – maximum),

(i) plot $v(t)$ vs t for $-12 \leq t \leq 8$ to confirm your conclusions for **(f-g).**

Warning! All calculation steps must be performed in your *Mathematica* notebook, or the score for this exercise will be zero!

6. Integral Calculus

In Section 4 we discussed that the velocity of a particle moving in a straight line can be obtained if its position as a function of time is known [10]. Mathematically, the velocity equals the derivative of the position with respect to time [10], i.e.

$$v(t) = \frac{dx(t)}{dt}$$

It is also possible to find the position of a particle if its velocity is known as a function of time [10]. In calculus, the procedure used to perform this task is referred to either as *integration* or as finding the *antiderivative*. Graphically, it is equivalent to finding the area under a curve [10].

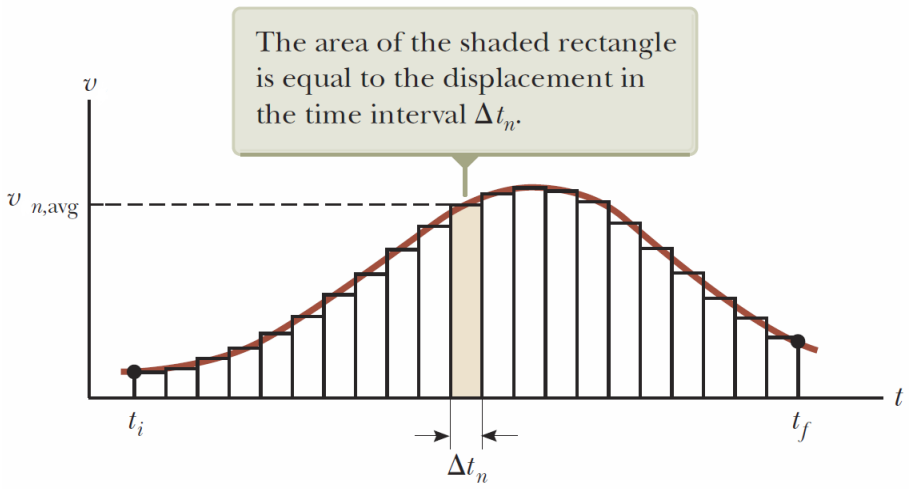


Figure 8 [10]. Velocity versus time for a particle moving along the *x* axis. The total area under the curve is the total displacement of the particle [10].

Suppose the $v-t$ graph for a particle moving along the *x* axis is as shown in Figure 8 [10]. Let us divide the time interval $t_f - t_i$ into many small intervals, each of duration Δt_n. From the definition of average velocity, we see that the displacement of the particle during any small interval, such as the one shaded in Figure 8, is given by $\Delta x_n = v_{n,\text{avg}} \cdot \Delta t_n$, where $v_{n,\text{avg}}$ is the average velocity in that interval [10]. Therefore, the displacement during this small interval is simply the area of the shaded rectangle in Figure 8 [10]. The total displacement for the interval $t_f - t_i$ is the sum of the areas of all the rectangles from t_i to t_f [10]:

$$\Delta x = \sum_n v_{xn,\text{avg}} \Delta t_n \tag{33}$$

Now, as the intervals are made smaller and smaller, the number of terms in the sum increases and the sum approaches a value equal to the area under the curve in the velocity–time graph [10]. Therefore, in the limit $n \to \infty$, or $\Delta t \to 0$, the displacement is [10]

$$\Delta x = \lim_{\Delta t_n \to 0} \sum_n v_{n,\text{avg}} \Delta t_n \tag{34}$$

If we know the $v-t$ graph for motion along a straight line, we can obtain the displacement during any time interval by measuring the area under the curve corresponding to that time interval [10]. The limit of the sum shown in equation (34) is called a *definite integral* and is written [10]

$$\lim_{\Delta t_n \to 0} \sum_n v_{n,\text{avg}} \Delta t_n = \int_{t_i}^{t_f} v(t)dt \tag{35}$$

where $v(t)$ denotes the velocity at any time t [10]. If the explicit functional form of $v(t)$ is known and the limits are given, the integral can be evaluated [10].

Sometimes the $v-t$ graph for a moving particle has a shape much simpler than that shown in Figure 8 [10]. For example, suppose an object is described with the particle under a constant velocity model [10]. In this case, the $v-t$ graph is a horizontal line as in Figure 9 and the displacement of the particle during the time interval Δt is simply the area of the shaded rectangle [10]:

$$\Delta x = v_{i,\text{avg}} \Delta t \ \ (\text{when } v = v_i = \text{constant}) \tag{36}$$

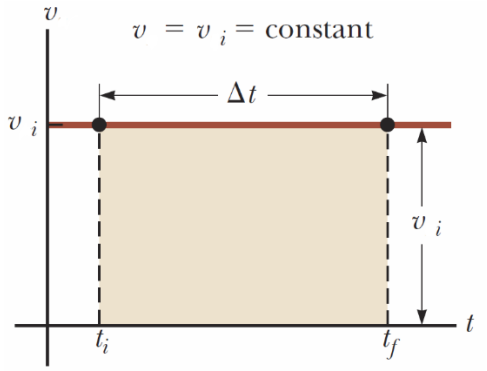


Figure 9 [10]. The velocity–time curve for a particle moving with constant velocity v_i. The displacement of the particle during the time interval t_f - t_i is equal to the area of the shaded rectangle.

Recall, that the acceleration is defined as

$$a = \frac{dv}{dt}$$

First, we rewrite this equation as

$$dv = adt \tag{37}$$

We now integrate the two sides between the initial (i) and final (f) values:

$$\int_{v_i}^{v_f} dv = \int_{t_i}^{t_f} adt \tag{38}$$

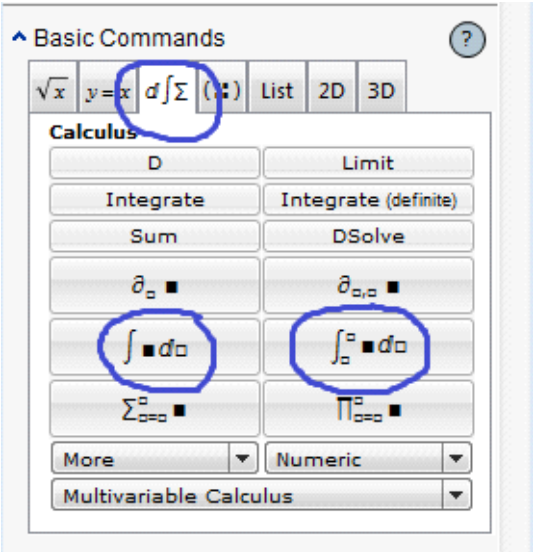
The left hand side (lhs) is trivial, it evaluates to

$$\int_{v_i}^{v_f} dv = v_f - v_i = \Delta v \tag{39}$$

In *Mathematica*, integration is easy. You can either type in the command Integrate[]

http://reference.wolfram.com/language/ref/Integrate.html

or use the integral buttons in the $\partial\int\Sigma$ section of the "Basic Math Assistant" palette:



Here is my *Mathematica* code that demonstrates how to use both options:

```
In[1]:= Integrate[1, {v, vi, vf}]

Out[1]= vf - vi

In[2]:= ∫_vi^vf 1 dv

Out[2]= vf - vi
```

As expected the result is the same, and, of course, agrees with that given in equation (39). Now we deal with the right hand side (rhs) of equation (38). Note that we integrate between the initial time (t_i) and final time (t_f). By convention, the initial time is set to zero, i.e. $t_i = 0$: recall that when you time an event, you usually start at zero, i.e. $t = 0$! On the contrary, the initial velocity is not necessarily zero (the object can be moving when we start the timer), so $v_i = v_0$ (which may or may not be zero). Then, we can drop the index (f) on the final time, $t = t_f$, and velocity ($v_f = v$) and write:

$$\Delta v = v - v_0 = \int_0^t a\,dt \tag{40}$$

This shows that the change in the x-velocity is the time integral of the x-acceleration. In general, acceleration may change with time, $a = a(t)$, which gives:

$$v = v_0 + \int_0^t a(t)\,dt \tag{41}$$

Using the same arguments, we can rewrite the equation for the displacement as

$$x = x_0 + \int_0^t v(t)\,dt \tag{42}$$

Here x and v are the position and x-velocity at time t. If we know the x-acceleration as a function of time and we know the initial velocity we can use equation (41) to find the x-velocity at any time, i.e. as a function of time. Once we know this function, and given the initial position we can use equation (42) to find the position x at any time.

Equations (41) and (42) are called the *kinematic equations*. Unfortunately, the right-hand side of each equation is impossible to evaluate without the explicit knowledge of a time-dependence of velocity or acceleration, i.e. $v(t)$ and $a(t)$.

In your basic Physics course, you considered the special case in which the acceleration is constant, a = constant. Recall that a constant can be moved out of an integral. As such, the integral in equation (41) can be evaluated as

```
In[3]:= ∫_0^t a dt

Out[3]= a t
```

Which gives a very simple formula for the time-dependence of velocity $v(t)$:

$$v = v_0 + \int_0^t a dt = v_0 + a\int_0^t dt = v_0 + a(t-0) = v_0 + at \tag{43}$$

This powerful expression enables us to determine an object's velocity at any time t if we know the object's initial velocity v_0 and its (constant) acceleration a [10]. A velocity–time graph for this constant-acceleration motion is shown in Figure 10a [10]. The graph is a straight line, the slope of which is the acceleration a; the (constant) slope is consistent with $a = dv/dt$ being a constant [10]. Notice that the slope is positive, which indicates a positive acceleration [10]. If the acceleration were negative, the slope of the line in Figure 10a would be negative [10]. When the acceleration is constant, the graph of acceleration versus time (Fig. 10b) is a straight line having a slope of zero [10].

Figure 10 [10]. A particle under constant acceleration a moving along the x axis: (a) the velocity–time graph, and (b) the acceleration–time graph [10].

Because we know how velocity depends on time, $v = v_0 + at$, we can use this expression to evaluate the integral on the right-hand side of equation (42):

$$x = x_0 + \int_0^t (v_0 + at)dt \tag{44}$$

First we recall that the integral of a sum is a sum of integrals, i.e.

$$x = x_0 + \int_0^t v_0 dt + \int_0^t (at)dt \tag{45}$$

The first integral is easy because the initial velocity (v_0) is a constant (naturally, it does not change with time), so we get:

$$\int_0^t v_0 dt = v_0\int_0^t dt = v_0(t-0) = v_0 t \tag{46}$$

The second integral is not much more complicated. Because acceleration is constant, we get:

$$\int_0^t atdt = a\int_0^t tdt = a\left(\frac{t^2}{2} - \frac{0^2}{2}\right) = \frac{1}{2}at^2 \tag{47}$$

Combining the results for the two integrals, we get:

$$x = x_0 + v_0 t + \frac{1}{2}at^2 \tag{48}$$

This equation provides the final position of the particle at time t in terms of the initial position, the initial velocity, and the constant acceleration. Of course in *Mathematica*, equation (48) can be obtained from equation (44) in one simple step:

```
In[4]:= x0 + ∫_0^t (v0 + a * t) dt

Out[4]= (a t^2)/2 + t v0 + x0
```

Example 6.1. Suppose, a car is moving along a straight highway. At exactly 4 hours (h) after beginning its journey, it is 200 km from the starting point (x_0), and the speed is 120 km/h (v_0). At this point the driver begins to decrease the speed exponentially (with a constant acceleration) such that at t = 5 h the car's speed is 16.240 km/h. **(a)** Calculate the numerical value of acceleration Find expressions for **(b)** x-velocity v and **(c)** position x as functions of time after t = 4 hours. Compute the **(d)** velocity and **(e)** distance of the car from its starting point at t = 6 hours. Plot **(f)** x vs t, and **(g)** v vs t for t = 4…7 hours

In the first cell, specify the project number:

```
In[1]:= (*                  *)
        (*  EXAMPLE 6.1     *)
        (*                  *)
        Clear["Global`*"];
```

In the next couple of statements define x_0 and v_0 as given in the text:

```
In[2]:= (* GIVEN *)
        (* x0 - distance from a starting point (in km) *)
        x0 = 200.;
        (* speed at x0 (in km/h) *)
        v0 = 120.;
```

Note that because we do not wish to print those out, we terminate each statement with the ";" symbol. Now there is a more difficult part. The question says that speed decreases exponentially with time but acceleration is constant, i.e. e^{-at}, where t is time (in h) a is the acceleration (in km/h^2). Note that in this equation t is being counted from 4 hours (because that is when deceleration started). To bring t to the absolute time scale (i.e. counting t from the beginning of the journey), I modify t in e^{-at} to t-4, i.e. $e^{-a(t-4)}$. Note that this equation shows *the change in*

speed, but the initial speed was v_0, so I include it in $e^{-a(t-4)}$ which completes the expression for $v(t)$:

$$v(t) = v_0 e^{-a(t-4)} \quad \text{for } t \geq 4 \text{ hours} \tag{49}$$

where v is speed (in km/h) after t = 4 hours, t is time (in h), and a is the acceleration (in km/h^2). Of course, this equation is valid only for $t \geq 4$ hours, because we have no knowledge of how the car was moving between t = 0 and t = 3.99999… hours. Now I enter this expression into *Mathematica* and include a bunch of comments (so I do not forget what I am doing):

```
In[4]:= (* after t = 4 hours, speed is reduced exponentially *)
        (* this can be described by the following equation *)
        v = v0 * Exp[-a * (t - 4)];
        Print["(b) v(t) = ", v, " km/h"];
        (* where v - speed after t = 4 hours (in km/h),
        a - acceleration (in km/h^2),
        t - time (h) *)
        (b) v(t) = 120. e^(-a (-4+t)) km/h
```

Note that in this expression acceleration, a, is not known. But we can certainly determine it because we know that at t = 5 h the speed (v) was 16.240 km/h. All we need to do is insert these values of t and v into the formula for $v(t)$ (equation 49) and solve it for a :

```
In[6]:= (* acceleration is not known, so we need to find it *)
        (* we know that at t = 5 hours, v is 16.240 km/h *)
        t5 = 5;
        v5 = 16.240;
        (* insert t5 and v5 into v(t) to get acceleration, a *)
        sols = Solve[v0 * Exp[-a * (t5 - 4)] == v5, a]
Out[8]= {{a → 2.00001}}

In[9]:= (* there is only one solution - this is the acceleration, a, in km/h^2 *)
        a = a /. sols[[1]];
        Print["(a) a = ", a, " km/h^2"];
        (a) a = 2.00001 km/h^2
```

Once a is known, we can complete the expression for $v(t)$

```
In[11]:= (* parameters v0 and a in equation (1) are now known: *)
         Print["(b) v(t) = ", v, " km/h"];
         (b) v(t) = 120. e^(-2.00001 (-4+t)) km/h
```

If $v(t)$ is known, we can calculate $x(t)$ using equation (42).

```
In[12]:= (* we can now easily find expression for x(t) after t = 4 hours *)
         x = x0 + ∫_4^t v dt;
         Print["(c) x(t) = ", x, " km"];
         (c) x(t) = 260. - 178866. e^(-2.00001 t) km
```

Note that x_0 is given in the text. You can easily verify that our expressions for $x(t)$ and $v(t)$ are correct, by using t = 4 h and t = 5 h and comparing the calculated values of x and v with those given in the text:

```
In[14]:= (* verify that our expressions for x(t) and v(t) are correct *)
         Print["x at t = 4 hours = ", x /. t → 4, " km"];
         Print["v at t = 4 hours = ", v /. t → 4, " km/h"];
         Print["x at t = 5 hours = ", x /. t → 5, " km"];
         Print["v at t = 5 hours = ", v /. t → 5, " km/h"];

         x at t = 4 hours = 200. km
         v at t = 4 hours = 120. km/h
         x at t = 5 hours = 251.88 km
         v at t = 5 hours = 16.24 km/h
```

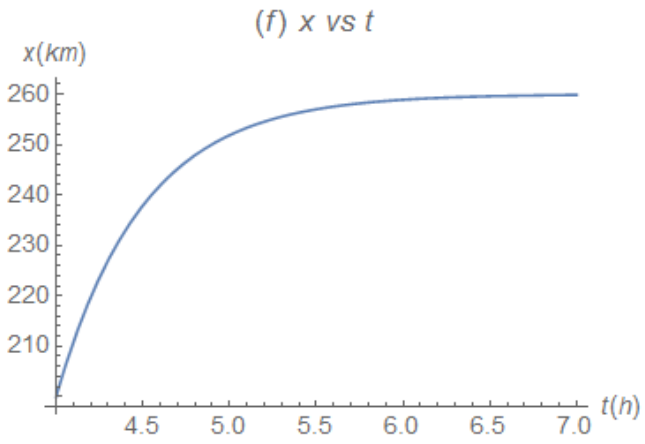
They agree!!! Now we can calculate x and v at $t = 6$ hours:

```
In[18]:= (* compute speed and distance at t = 6 hours *)
         Print["(d) v at t = 6 hours = ", v /. t → 6, " km/h"];
         Print["(e) x at t = 6 hours = ", x /. t → 6, " km"];

         (d) v at t = 6 hours = 2.19781 km/h
         (e) x at t = 6 hours = 258.901 km
```

Finally, plotting $x(t)$ is straightforward:

```
In[20]:= (* plot x(t) and v(t) for t = 4...7 hours *)
         SetOptions[Plot, BaseStyle → {FontSize → 14}];
         Plot[x, {t, 4, 7}, AxesLabel → {Style["t(h)", Italic], Style["x(km)", Italic]},
          PlotLabel → Style["(f) x vs t", Italic]]
```

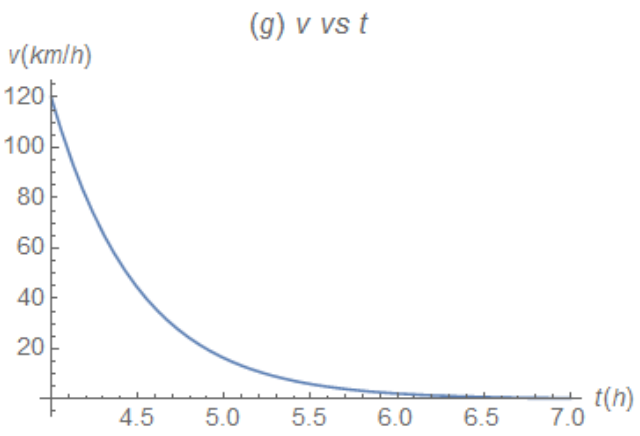
Out[21]=



Note that we, again, used *Mathematica*'s function SetOptions[Plot, BaseStyle → {FontSize → 14}] to increase the font size in the plots.

Let's see how the $v(t)$ plot looks like:

```
In[22]:= Plot[v, {t, 4, 7}, AxesLabel → {Style["t(h)", Italic], Style["v(km/h)", Italic]},
  PlotLabel → Style["(g) v vs t", Italic]]
```



Out[22]=

Now, that wasn't too bad, was it?!

Integral Calculus is, in general, not easy, so we shall consider another example. This time it is your favorite subject – Chemistry!

Example 6.2. Consider the following chemical reaction [8,13,14]

$$\mathrm{A} \xrightarrow{k} \mathrm{P} \tag{50}$$

where A is a reactant, P is a product, and k is the rate constant [10,6]. Suppose the reaction follows the *first order* mechanism, which means that the rate of consumption of A is proportional to the first power of the concentration of A, [A]:

$$-\frac{d[\mathrm{A}]}{dt} = k[\mathrm{A}] \tag{51}$$

We need to **(a)** derive the expression for [A] = $f(t)$ (called the *integrated rate law*), and **(b)** determine the expression for the parameter half-life, $t_{1/2}$ at which the concentration [A] falls to half of its original concentration $[\mathrm{A}]_0$.

Before we get to *Mathematica*, we need to rearrange equation (51) in such a way as to collect terms related to the concentration of A (i.e. [A] and d[A]) on one side of the equation (suppose, left hand side, lhs) and terms related to time (t) on the other side (suppose, right hand side, rhs). It does not matter where you put constants such as k and (-1) because they always factor out of integrals. To this purpose, I rearrange equation (51) as

$$\frac{1}{[\mathrm{A}]} d[\mathrm{A}] = -k dt \tag{52}$$

In the next step, we need to integrate both sides. Suppose, the left hand side is integrated between the initial concentration $[\mathrm{A}]_0$ and concentration [A] at time t. Then, the right hand side is integrated

between the initial time (which we always consider as zero), and time t at which the concentration of A falls to [A]. The following equation demonstrates that:

$$\int_{[A]_0}^{[A]} \frac{1}{[A]} d[A] = \int_0^t -k dt \tag{53}$$

Now, it is time to go to *Mathematica*. As always, the first cell contains the project number:

```
In[1]:= (*                *)
        (*  EXAMPLE 6.2   *)
        (*                *)
        Clear["Global`*"];
```

Then, we add a comment summarizing the preliminary steps:

```
In[2]:= (* first-order rate equation is rearranged such as to have terms with [A]
           on one side, *)
        (* and terms with t on the other side *)
        (* d[A]/[A] = -k*dt *)
```

We integrate the left hand side:

```
In[3]:= (* integrate left hand side *)
        (* because Mathematica always assumes that symbolic variables are complex,
        we help it by telling it that the concenrtraion of A, [A],
        is always non-negative, and that [A]≤ [A]0,
        i.e. concentration decreases with time *)
        lhs = Integrate[1/A, {A, A0, A}, Assumptions → 0 < A < A0];
        Print["lhs = ", lhs];
        lhs = Log[A/A0]
```

Note that because *Mathematica* always thinks that a given symbolic parameter may be complex, I had to give it additional information about the integrand and the limits of the integration. Indeed, the initial concentration $[A]_0$ must always be positive, while [A] is always less than $[A]_0$. Note that while in general, $[A] = [A]_0$ at $t = 0$, in this particular case it is safer to use the relationship shown above. As you can see, we make use of the *Mathematica* command Assumptions:

http://reference.wolfram.com/language/ref/Assumptions.html

As expected, the integral on rhs evaluates to $\ln\left(\frac{[A]}{[A]_0}\right)$. Note that in *Mathematica* the natural logarithm is entered as Log[ ],

http://reference.wolfram.com/language/ref/Log.html

while the base-10 logarithm is written as Log10[]:

http://reference.wolfram.com/language/ref/Log10.html

Ok, now we need to evaluate the right hand side of equation (53). This, of course, is very easy:

```
In[5]:= (* integrate right hand side *)
        rhs = ∫_0^t -k dt;
        Print["lhs = ", rhs];
        lhs = -k t
```

Now we combine the two sides, and solve for [A]:

```
In[7]:= (* solve equation lhs==rhs for [A] *)
        (* note that we are interested in Real solutions only *)
        sols = Solve[lhs == rhs, A, Reals]
Out[7]= {{A → A0 e^(-k t)}}
```

Note how I told *Mathematica* that we are interested in *Real* solutions only (concentrations can not be complex!). In the next step, I copy the solution to variable "A" and print it out:

```
In[8]:= (* copy solution to variable A *)
        A = A /. sols[[1]];
        Print["(a) first order integrated rate law: A = ", A];
        (a) first order integrated rate law: A = A0 e^(-k t)
```

This is exactly the same result as given in your Physical Chemistry textbook [8,13,14].

$$[\mathrm{A}]=[\mathrm{A}]_0 e^{-kt} \tag{54}$$

Now, all we need to do is solve this equation for half-life ($t_{1/2}$). From the definition of half-life, it is time (t) at which [A] falls to $[\mathrm{A}]_0/2$. We can certainly do that in *Mathematica*:

```
In[10]:= (* solve for half-life, t_1/2, time at which [A]=[A]0/2 *)
         sols = Solve[A == A0 / 2, t, Reals]
Out[10]= {{t → Log[2]/k}}

In[11]:= (* copy solution to variable t12 *)
         t12 = t /. sols[[1]];
         Print["(b) half-life for the  first order integrated rate law: t_1/2 = ", t12]
         (b) half-life for the  first order integrated rate law: t_1/2 = Log[2]/k
```

Compare our formula with that given in the textbook [8,13,14],

$$t_{1/2}=\frac{\ln(2)}{k} \tag{55}$$

Success!

Exercise 6.1. An object is oscillating along the *x*-axis with acceleration a (cm s^{-2}) given by the equation

$$a(t) = -\pi^2 \sin\left(\frac{\pi}{2}t\right) \tag{56}$$

where *t* is in seconds. Suppose, the object initial velocity was 2π (cm s^{-1}) and the initial position was 1 cm. Using your *Mathematica* notebook for Example 6.1 as a template, derive the expressions for **(a)** $v(t)$ and **(b)** $x(t)$. **(c)** On the same graph (set FontSize to 14, set ImageSize to 600, and include the plot legend as shown in section 4 of this tutorial), plot *x,* v, and a vs t for *t* = 0…6 s. Save the notebook as "Exercise_6.1.nb". **Warning! All calculation steps must be performed in your *Mathematica* notebook, and all items (a-c) must be clearly labelled in the Print[] functions, or the score for this exercise will be zero!**

Answers: **(a)** $2\pi \cos\left(\frac{\pi}{2}t\right)$ (cm s^{-1}) **(b)** $1 + 4\sin\left(\frac{\pi}{2}t\right)$ (cm)

(c) …

Exercise 6.2. An object is moving along the *x*-axis with acceleration a (m s^{-2}) given by the equation

$$a(t) = \beta e^{-\alpha t} \tag{57}$$

where α and β are some constants (α is in s^{-1}, and β is in m s^{-2}), and *t* is in seconds. Suppose the object initial position (at *t* = 0 s) was x_0 and its initial velocity was v_0. Using your *Mathematica* notebook for Example 6.1 as a template, determine equations for the object **(a)** *x*-velocity and **(b)** *x*-position as a function of time t and general constants x_0, v_0, α and β. Suppose, at t= 2 s the object was moving with velocity $\frac{1}{\beta} - \frac{\beta e^{-2\alpha}}{\alpha}$ (m s^{-1}), and at t = 4 s the object was located at $\alpha + \frac{4}{\beta} + \frac{\beta e^{-4\alpha}}{\alpha^2}$ (m). Determine and simplify the values of **(c)** v_0 and **(d)** x_0 in terms of coefficients α and β. Derive and simplify the expressions for **(e)** $v(t)$ and **(f)** $x(t)$ in terms of time t and coefficients α and β. Save the notebook as "Exercise_6.2.nb". **Warning! All calculation steps must be performed in your *Mathematica* notebook, and all items (a-f) must be clearly labelled in the Print[] functions, or the score for this exercise will be zero!**

Answers: **(a)** … **(b)** …

(c) $\frac{1}{\beta} - \frac{\beta}{\alpha}$ (m s^{-1}) **(d)** $\alpha + \frac{\beta}{\alpha^2}$ (m)

(e) … **(f)** $\alpha + \frac{t}{\beta} + \frac{\beta e^{-t\alpha}}{\alpha^2}$ (m)

Exercise 6.3. In this final exercise you will follow a 1911 determination of Avogadro's number (the currently established IUPAC value is $N_A = 6.02214179\times10^{23}$ mol^{-1} [16]) based on the rate of emission of α-particles by radium (Ra, $Z = 88$), and the rate of production of helium by radium. In case you have forgotten, an α-particle is just a bare nucleus of the helium-4 isotope ($^4_2$He) that carries a positive charge of +2. Radium undergoes α-decay to yield radon (Rn, $Z = 86$) [17]:

$$^{226}_{88}\text{He} \rightarrow {}^{222}_{86}\text{He} + {}^{4}_{2}\text{He}$$

The rate of emission of α-particles by radium was determined by Rutherford and Geiger [18] to be 3.4×10^{10} α-particles per second per 1 g of radium.

Then, Boltwood and Rutherford [19] separated some radium salt from its decomposition products and measured the volume of helium produced after known times. They found that from 192 mg of radium 6.58 mm^3 at STP of helium was produced in 83 days, and 10.38 mm^3 at STP in 132 days. The half-life of radon ($t_{1/2}$) is 3.83 days (d). STP stands for "standard temperature and pressure". Until 1982, STP was defined as a temperature of 273.15 K and pressure of exactly 1 atm; since 1982, STP is defined as a temperature of 273.15 K and pressure of exactly 1 bar = 10^5 Pa [20].

Boltwood and Rutherford also derived the following formula for the total quantity of helium produced (Q):

$$Q = xT + 3x\int_0^T \left(1 - e^{-\lambda t}\right)dt \tag{58}$$

where x is the rate of production of helium by the radium itself, T - period of accumulation of helium, and λ is the decay constant of radon. From x we can obtain the number of moles of helium produced by 1 g of radium per second. Comparing this with the actual number of α-particles emitted by 1 g of radium per second we obtain Avogadro's number. **Warning! All calculation steps (a-j) must be performed in your *Mathematica* notebook, and all result (a-j) must be clearly labelled in the Print[] functions, or the score for this exercise will be zero! The following sequence of steps is required:**

(a) evaluate the right hand side (r.h.s) of equation (58), $f(T, x, \lambda)$

(b) rearrange equation $Q = f(T, x, \lambda)$ to $x = f(Q, T, \lambda)$,

(c) calculate the decay constant of radon, $\lambda = \frac{\ln 2}{t_{1/2}}$ (d^{-1}),

(d) determine the value of x_1 $\left(\frac{\text{mm}^3 \text{ at STP}}{\text{d}}\right)$ - the rate of production of helium by 192 mg of radium for T_1 = 83 d and Q_1 = 6.58 mm^3 at STP,

(e) determine the value of x_2 $\left(\frac{\text{mm}^3 \text{ at STP}}{\text{d}}\right)$ - the rate of production of helium by 192 mg of radium for T_2 = 132 d and Q_2 = 10.38 mm^3 at STP,

(f) using the *Mathematica* function Mean[] calculate the average value of x $\left(\frac{\text{mm}^3 \text{ at STP}}{\text{d}}\right)$,

(g) convert the average value of x from $\left(\frac{\text{mm}^3 \text{ at STP}}{\text{d}}\right)$ to $\left(\frac{\text{mol}}{\text{s}}\right)$; recall that …

- 1 mol of gas at STP occupies 22.4 L,
- 1 L = 10 cm × 10 cm × 10 cm,
- 1 cm = 10 mm,
- 1 d (day) = 24 hours, 1 hour = 60 min, 1 min = 60 s,

(h) calculate the rate of production of helium $\left(\frac{\text{mol}}{\text{s}}\right)$ by 1 g of radium (note that the value of x calculated at the previous step is the rate of production of helium by 192 **m**g of radium),

(i) since the rate of emission of α-particles by 1 g of radium is 3.4×10^{10} s^{-1}, our (technically, Rutherford's) experimental Avogadro's number, $N_{\text{A,exp}} = \frac{3.4\times10^{10}\left(\frac{1}{\text{s}}\right)}{x\left(\frac{\text{mol}}{\text{s}}\right)}$,

(j) calculate the percent error for $N_{A,\text{exp}}$ relative to the currently established value; how well do you think they did in 1911 ?

Answers: **(a)** … **(b)** $\frac{e^{T\lambda}Q\lambda}{3+e^{T\lambda}(-3+4T\lambda)}$ **(c)** …

(d) … **(e)** … **(f)** 2.06×10^{-2} $\frac{\text{mm}^3 \text{ at STP}}{\text{d}}$

(g) … **(h)** 5.54×10^{-14} $\frac{\text{mol}}{\text{s}}$ **(i)** …

(j) …

Whew, wasn't it fun?! 😉 Believe it or not, but this concludes our brief *Mathematica* tutorial. We hope that we have been able to convince you that this amazing software can provide a significant help in learning subjects like Mathematics, Physics, and Chemistry (recall that Physical Chemistry = Mathematics + Physics + Chemistry).

7. References

1. Wolfram Research, Inc. *Mathematica 9.0*. Champaign, Illinois, USA, 2012.
2. http://www.wolfram.com/mathematica
3. https://en.wikipedia.org/wiki/Mathematica
4. https://en.wikipedia.org/wiki/Stephen_Wolfram
5. https://en.wikipedia.org/wiki/Wolfram_Research
6. Tam, P. T. *A Physicist's Guide to Mathematica*, 2nd edition, Academic Press, 2008.
7. https://www.duolingo.com/comment/15264140/How-To-Type-Accented-Characters-On-An-English-Keyboard-Windows-10-Tutorial
8. Atkins, P; de Paula, J. *Elements of Physical Chemistry*, **5**th edition, W. H. Freeman, New York, 2009.
9. Serway, R. A., Vuille, C. *College Physics*, 9th edition. Cengage Learning, 2011.
10. Serway, R. A., Jewett, J. W. *Physics for Scientists and Engineers with Modern Physics*, 9th edition. Brooks Cole, 2013.
11. Young, H. D., Freedman, R. A., Ford, A. L. *Sears and Zemansky's University Physics with Modern Physics*, 13th edition. Pearson Education, Inc., 2012.
12. Barrante, J. R. *Applied Mathematics for Physical Chemistry*, 3rd edition. Prentice Hall; 2003.
13. Atkins, P; de Paula, J. *Physical Chemistry*, 8th edition, W. H. Freeman and Company, New York, 2006.
14. Atkins, P; de Paula, J. *Elements of Physical Chemistry*, **6**th edition, W. H. Freeman, New York, 2012.
15. Ball, D. W. *Physical Chemistry*, 1st edition. Cengage Learning; 2002.
16. http://goldbook.iupac.org/html/A/A00543.html
17. Silberberg, M. Principles of General Chemistry, 1st edition. McGraw-Hill, 2007.
18. Rutherford, E. & Geiger, H. Proc. Roy. Soc. A, 81, 141, 1908.
19. Boltwood, B. B. & Rutherford, E. Phi. Mag., 22, 586, 1911.
20. http://en.wikipedia.org/wiki/Standard_conditions_for_temperature_and_pressure