SMath Suite: A Primer

The author wishes to thank Prof. Gilberto E. Urroz for his most kind permission to use some of his material from the SMath and his own websites.

Bernard V Liengme email: bliengme@stfx.ca website: http://people.stfx.ca/bliengme/SMath (Nov 2011)

Table of Contents

The SMath interface 2
Constructing Regions 3
Scratchpad Calculations 6
Preparing for Math Problems 7
Subscripted Variable Names 11
Working with Functions 11
User-Defined Functions 15
Numerical Differentiation and Integration 15
Working with Units 17
Non-Linear Units (°C → K) 19
Predefined constants 20
Some Notes on Units (adapted from Prof. Urroz's notes) 22
Matrices and Vectors 22
Matrix operations 22
Example of Matrix Math 23
Vector Operations 24
Simple Statistics 26
Note on Matrix/Vectors calculations 26
Drawing graphs 27
Solving equations and finding roots 30
Effect of changing Tools | Options | Decimals 35
Solving (not root finding) 36
Symbolic Differentiation 38
Programming 38
Linear Regression Example 43
Multi-linear Regression 44

The developers of SMath Suite call is a *mathematical program with paper-like interface and numerous computing features.* SMath has many of the features found in the expensive application Mathcad but differs in that SMath is free. The SMath user interface resembles that of Mathcad.

SMath may be downloaded from: http://en.smath.info/forum/default.aspx?g=topics&f=12

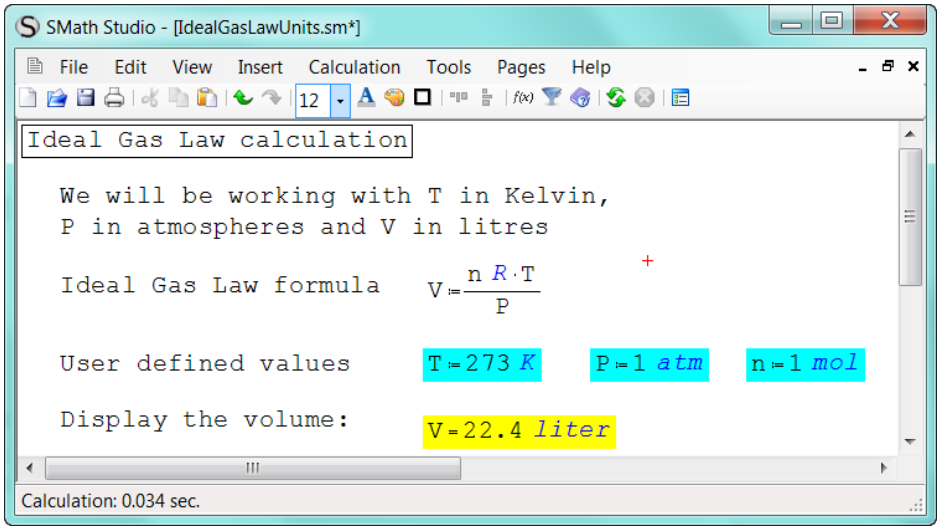
SMath Studio supports 18 interface languages, and has versions for Windows, Windows Mobile (PPC, communicators and Smartphones) and Linux. There is an EXE version that one need not install but can run from a thumb drive; useful for running on PCs in a university lab or work environment. These notes assume the reader is using the beta 0.89.8 version. You can also run SMath from your browser at http://smath.info/live/?lang=eng but I find it a bit buggy at the moment.

Main web page is http://en.smath.info/forum/

The site is useful http://www.neng.usu.edu/cee/faculty/gurro/SMathStudio.html

If you do an internet search use "SMath Studio" to avoid other unrelated products.

Here is a screen capture of a SMath page that solves Ideal Gas Law problems.



The designer of this page has shaded blue the areas where a user should enter new values, and yellow the final result.

The SMath interface

The interface shows a main window together with menus, a toolbar, and a number of palettes on the right-hand side of the interface. Just as in Excel one speaks of worksheets, so in SMath we speak of the main window as a *Page,* although the term 'worksheet' is also used. A SMath file contains only one page.

The menu items are: *File, Edit, View, Insert, Calculation, Tools, Pages,* and *Help*. As expected *File* leads to commands such as Save, Save As, Open, and Print. Note that SMath work can be saved in its native format (extension SM), as an HTML file, as a graphic file (PNG, BMP, GIF, or JPG – the user must add the appropriate graphic extension), and other formats that we will not discuss. Likewise *Edit* has the expected commands

such as Copy, Paste, Undo, etc. Within *View* we can decide if we wish to have the page display gridlines making it look like quarter-inch graph paper. The menu items Calculations and Tools are discussed as we proceed. The *Page* menu let us start a new page, close a page, or switch between open pages. The *Help* menu contains the option *Reference book…* which represents a modest attempt by the developer to address basic mathematical operations and definitions. There is little in *Help* to tell you how to use SMath but it does has some sample files.

The palettes contain mathematical, graphical, and programming functions that can be placed in the main window. The pallets may be toggled on/off with last item on the toolbar.

The *SMath Studio* toolbar contains 19 icons briefly described below:

1. New page
2. Open (existing worksheet)
3. Save (current worksheet)
4. Print (current worksheet)
5. Cut
6. Copy
7. Paste
8. Undo (recent action)
9. Redo (recent action)
10. Font size (of selected items on page)
11. Text color (of items to be entered)
12. Background color (of items to be entered)
13. Control border (frames selected items)
14. Align horizontally selected items
15. Align vertically selected items
16. Function (insert a function)
17. Unit (insert unit)
18. Reference book (see *Help*)
19. Recalculate page
20. Interrupt process
21. Show/hide side panel

Items (1) through (4) manipulate new or existing worksheets. Items (5) through (7) are well-known editing functions. Items (8) and (9) will un-do and re-do the most recent action. Items (10) through (12) adjust font or background properties – see *Construction Regions* below. Item (13) allows you to put a frame over an entry, for example, to show a solution to a problem. Items (14) and (15) re-align selected cells. Item (16) and (17) open the *Function* and *Unit* menus, which is also available in the *Insert* menu. Item (18) is also available under the *Help* menu. Items (19) and (20) are also available in the *Calculation* menu. Item (21) shows or hides the palettes on the right-hand side of the page.

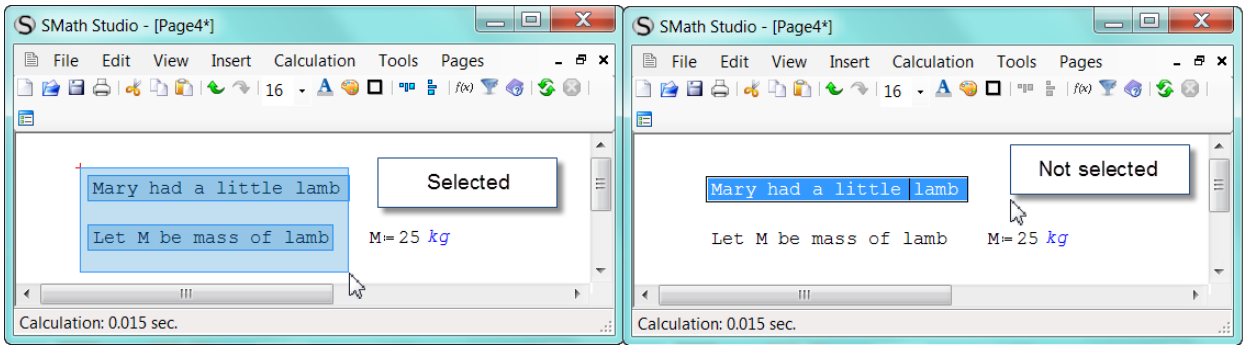
The on-line version of SMath lacks the menus and tools shown above. But you can both upload and download SM files to and from your computer.

Constructing Regions

There are two types of regions: text and math. In the diagram above, the words *Ideal Gas Law Calculation* are in a text region while the formula for V (V := RT/P) is a math region. As you move the mouse around you will see a red cross on the page; this is the insertion point and anything you type (or insert using a pallet) will begin at that point. There are some exceptions for operators inserted from a palette.

1) To enter text either use the menu command Insert | Text Region or, more conveniently, start with a double quote (“); this will not be displayed but will signal to SMath that you want to start a text region. To start a new line in a text region use SHIFT+ENTER not just ENTER. To start a math region just begin typing. When you have completed a region just click any empty space on the worksheet.

2) To select one or more regions for the purpose of deleting or aligning them, or changing their appearance: click somewhere above and to the left of the first region and drag the mouse around. A rectangle results and all items within the rectangle are selected. By default the rectangle is light blue. The rectangle will disappear when you release the mouse button but regions will remain coloured. Do not confuse this process with clicking on the region and dragging the mouse to highlighting just region’s contents – see figures below. The colour in the left is lighter than in the right.
It you want to select two or more items that are not contiguous, use the well-known Windows method of holding gown CTRL as you drag the mouse.



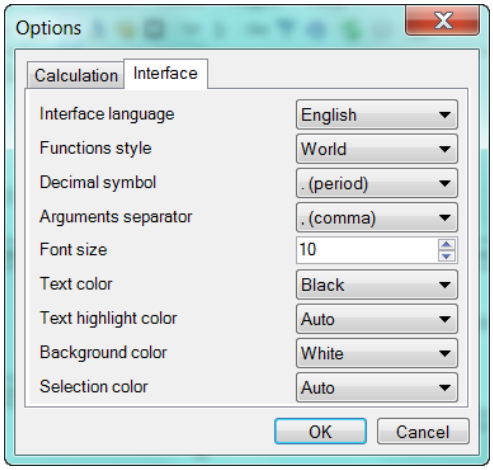
Selected regions may be copied or cut and then pasted using commands from the Edit menu or with the Window shortcuts CTRL+C, CTL+X and CTRL+V. Note that no regions from a SMath page can be copied into other applications such as Word. This is where saving as HTML or graphics is helpful. In Windows 7 one can also use the Snip tool to capture and paste one or more items from SMath to Word. However, you can copy text from other applications to a SMath page. The text 5+4/5-2= in Word can be pasted to SMath to give a text area but without the = sign it will give the start of a math expression.

$$5+\frac{4}{5}-2$$

3) Diagrams may be pasted from another application into a SMath worksheet. An image must be selected in order to move or resize it.

4) The font size and colour, and the background colour of the characters will depend on the setting within the Tool | Options | Interface dialog. If you change the font size, colour or background with this tool it will affect only subsequent entries. To change existing regions, select them and use one of the tools on the toolbar. To put a border around a region just click on the region and then the border tool – selection is not required for a single region.

5) Text regions, but not math regions, can be italicized, made bold or underlined by clicking anywhere within an entry and using the Windows shortcuts CTRL+I, CTRL+B and CRTL+U. To perform such on operation on several items just select them and use the shortcut. You cannot italicized, make bold or underlined only part of an entry – it is all or nothing.



6) To edit a region one must first click on it; a rectangle will result. Now enter the region and make the required edit. To delete something within a math region it is generally better to use the Backspace key rather than the Delete key.

7) Greek letters (for use in text or math entries) may be entered in several ways: a) by clicking on the appropriate symbol in the palette or b) by typing the Roman letter (see table below) followed by CTRL+G. Thus D followed by CTRL+G gives Δ. The symbol π (pi) may be inserted as p followed by CTRL+G; from the symbol or arithmetic palette, or with CTRL+SHIFT+p. When inserted into a math area π (pi) will have the value 3.14159265….

Lower case	**Upper case**	**Name**	**Roman letter**	**Lower case**	**Upper case**	**Name**	**Roman letter**
α	Α	alpha	a	ν	Ν	nu	n
β	Β	beta	b	ξ	Ξ	xi	x
γ	Γ	gamma	g	ζ	Ζ	xi	z
δ	Δ	delta	d	ο	Ο	omicron	o
ε	Ε	epsilon	e	π	Π	pi	p
ϕ	Φ	zeta	f	ρ	Ρ	rho	r
η	Η	eta	h	σ	Σ	sigma	s
θ	Θ	theta	q	τ	Τ	tau	t
ι	Ι	iota	i	υ	Υ	upsilon	u
κ	Κ	kappa	k	ψ	Ψ	psi	y
λ	Λ	lambda	l	φ	ϑ	phi	j
μ	Μ	mu	m	ω	Ω	omega	w

Scratchpad Calculations

NOTE: SMath always works with a precision of 15 places as required by the IEEE 754 convention. But the number of decimals displayed is set in the Tools | Options menu. Such a setting applies to the entire page.

SMath can be used as a simple calculator. For example, open a new page, and click on the main window in an area near the top left corner. A small red cross will indicate the location where you want to enter a calculation.

Type the following: 1+2/3+4/5 =. The result is the following expression:

$$1+\frac{2}{3+\frac{4}{5}}=1.53$$

Notice the way that *SMath* interprets the two fractions. Try the following example also: click somewhere else in your worksheet and type: 5 + 2/3 [→]+ 4/5 = (where → represents the right arrow key). The result is

$$5+\frac{2}{3}+\frac{4}{5}=6.47$$

Now try this example by typing √ 2^2→+4^2→= Where the √ comes either from the arithmetic palette or from the shortcut backslash (\), the key found just above ENTER, and the caret key (^) found with SHIFT+6 is used to make an exponentiation. This yields the expression

$$\sqrt{2^2+4^2}=4.47$$

Consider these two expressions here. While there are mathematically equivalent, the second is more please to the eye. All that is needed is a tap on the right arrow key after typing the pi and before inserting the / (division operator)

$$3\cdot\frac{\pi}{2}=4.71 \quad \text{and} \quad \frac{3\cdot\pi}{2}=4.71$$

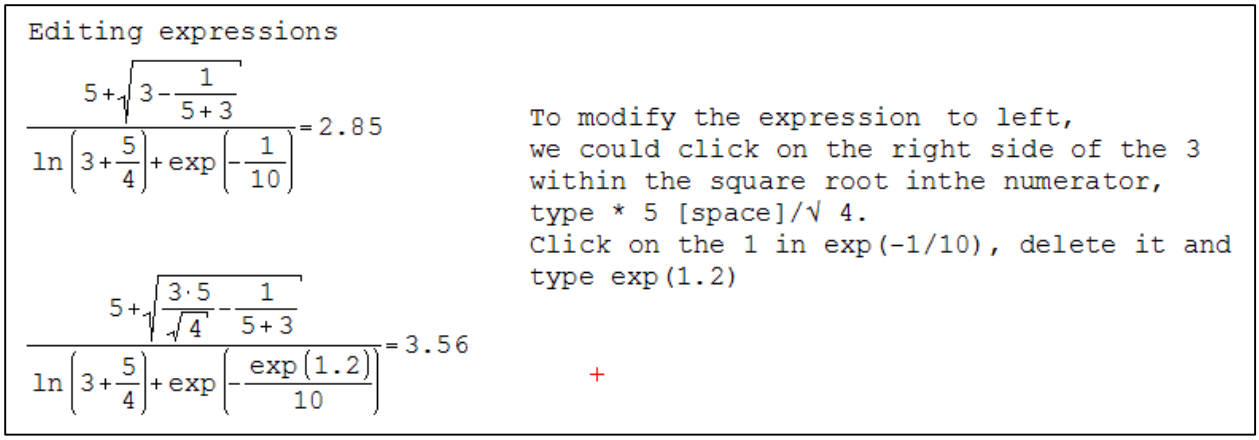
Note: the multiplication operator is the asterisk (*) as in most application, but SMath converts it to a middle dot.

Experiment to make this expression

$$\frac{6+4}{2}+\frac{10+6}{8}=7$$

You will get frustrated with SMath as it will keep giving the wrong expression. But here is a workaround: 6 + 4 [space] / 2 [space] + / 10 + 6 [right] [right] 8 [right]=. The trick is to insert the second divisor early.

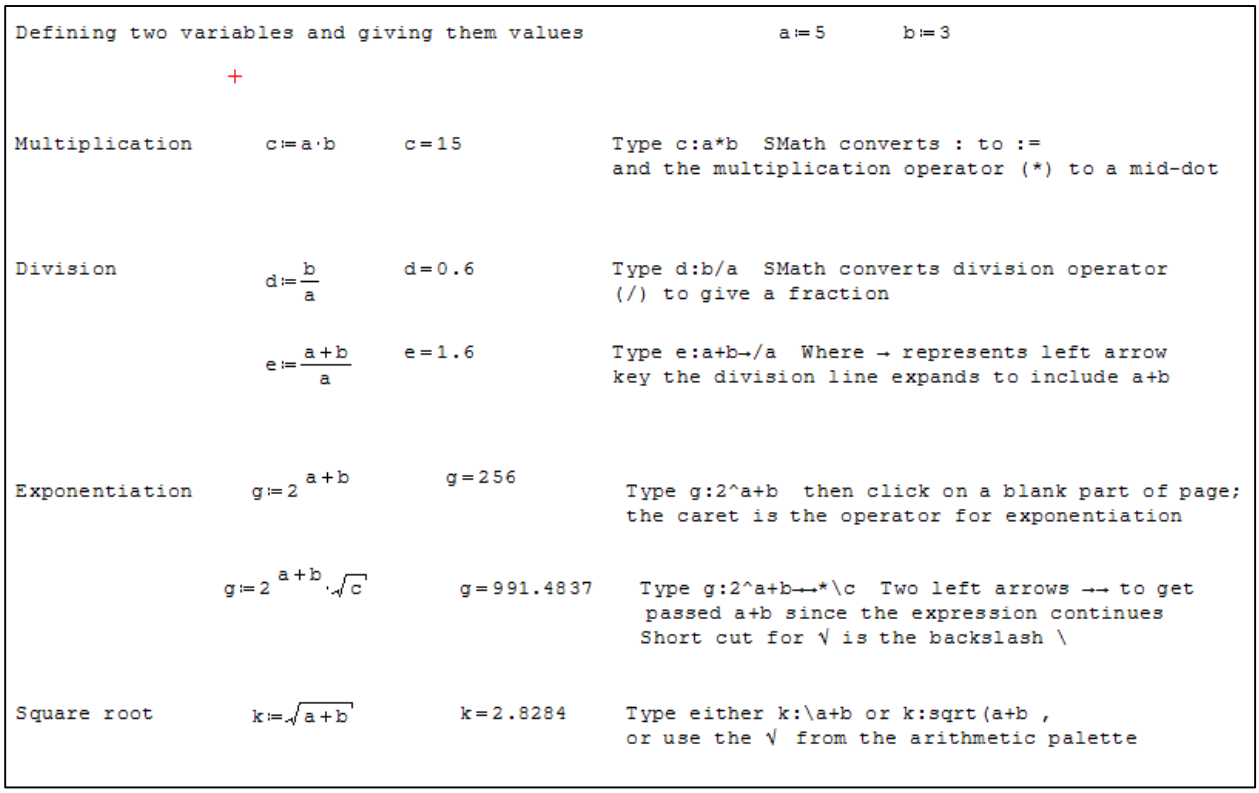
Here is an editing exercise to practice with.



Always use the right arrow key to expand the underscore line or to move to the right. The spacebar can be used in some circumstances and not others: it is simpler never to use it! One must not try to inset spaces in a math region to enhance appearance.

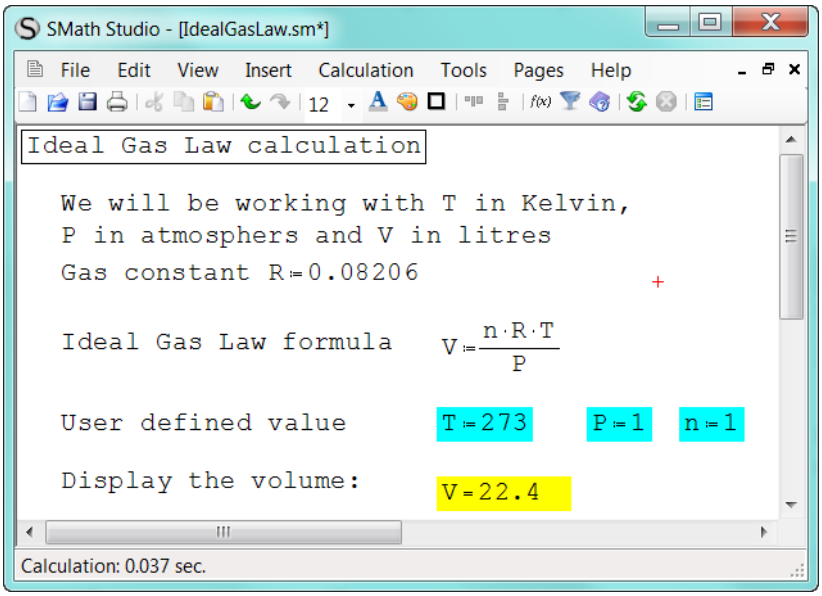
Preparing for Math Problems

The figure below shows examples of math regions



Below is a simple SMath page that can be used to find the volume of a gas given the number of moles (n), its temperature (T), and pressure (P) using the Ideal Gas Law. The user can change the n, T, and P values to perform any number of calculations. Note that *Gas constant R:=0,08206* consists of a text and a math region with space between the two.

REMEMBER: to give a value (or expression) to a variable as in x:=3, your type x:3. SMath automatically replaces the colon by the := operator. To have SMath display a value use the simple equal sign as in c=.



Note the asterisk at the end of the file name in the title bar: *IdealGasLaw.sm**

SMath is warning the user that the file has been altered since the last time it was saved.

The quantities R, T, P, V and n are called *variables*. Since these are readily recognized there is no need to define them or to use more descriptive names like Temp and Pressure. In other cases this is a good practice to follow.

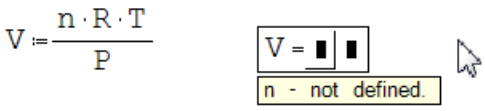
There are several important features to note:

a) The *definition* operator for assigning a value or expression to a variable is :=. Let's look at R:=0.08206, we type the R and then either click on the definition operator in the arithmetic palette, or we hold down SHIFT and type the colon (:) whereupon SMath will automatically replace it with :=. Then we type the value or expression.

b) The *evaluate numerically* operator that causes SMath to display a variable's value is the simple equals symbol. It can be inserted from the arithmetic palette or from the keyboard.

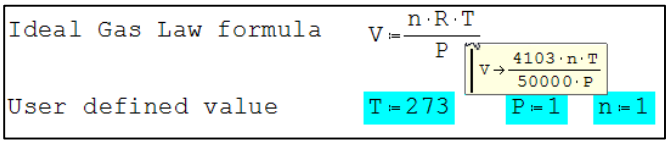
c) SMath is case sensitive with regard to variable names. If for our last line, we had typed v=, SMath would have changed it to v:=□ to show it is expecting a definition of variable v.

d) SMath works 'top-down' so if we added an entry to display V and placed this near the definition of V but before n, T, and P had been assigned values, SMath would not have an answer. Clicking on the expression would display



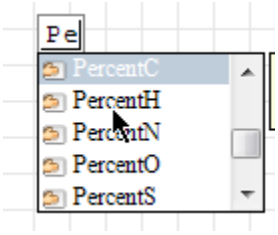
e) The entry points to give T, P and n values have been given a blue background to make them stand out while the result has a yellow background.

f) If you let the mouse pointer hover over a math region a popup will appear (see below) showing the region in symbolic representation. We will not explore this topic. For more information see http://smath.info/wiki/GetFile.aspx?File=Symbolic%20mathematic%2fsymbolic.pdf.



In the worksheet below see an example of SMath page that can be used to solve a simple chemistry problem. Once again we have used colour coding to show input and output area. Clearly one would not want to spend the time setting this up for one or two problems but if one repeatedly needs to solve such problems, it is well worth while.

Unlike some computer languages, SMath does not require you define every variable. Furthermore, some users are reluctant to use long names because of the chore of typing and the possibility of mistyping. SMath comes to the rescue with the Dynamic Assistant (toggled on/off from the View menu). Suppose you want to enter *PercentC*, as soon as you type *Pe*, the Dynamic Assistant shows a list of variable beginning with these two letters; you just need to click on one to insert it.



Note that in the worksheet above we have defined some formulas before entering the data. Generally this is acceptable but there are times when it fails – see file TopDown.sm.

Percentage composition

This page may be used to find the percentage composition of a compound containing C, H, 0, N and S from its formula

Atomic Weights

$C := 12.011 \quad H := 1.008 \quad O := 15.999 \quad N := 14.007 \quad S := 32.06$

Molecular Weight $MW := (c \cdot C) + (h \cdot H) + (o \cdot O) + (n \cdot N) + (s \cdot S)$

where variable in lower case are the stoichiometric numbers from the formula

Defintion of Percentage Formulas

$PercentC := \frac{c \cdot C}{MW} \cdot 100 \quad PercentH := \frac{h \cdot H}{MW} \cdot 100$

$PercentO := \frac{o \cdot O}{MW} \cdot 100 \quad PercentN := \frac{n \cdot N}{MW} \cdot 100 \quad PercentS := \frac{s \cdot S}{MW} \cdot 100$

So far we have entered universal data (atomic weights) and some formulas

Now we define the problem to be solved - specific data

Find percentage compostion of Ethanol C2H5OH

$c := 2 \quad o := 1 \quad h := 5 + 1 \quad n := 0 \quad s := 0$

Results

$MW = 46.069$

$PercentC = 52.1435 \quad PercentH = 13.1281$

$PercentO = 34.7283 \quad PercentN = 0 \quad PercentS = 0$

Check calculations

$Check := PercentC + PercentH + PercentO + PercentN + PercentS$

$Check = 100$

Subscripted Variable Names

Sometimes simple variable names are not appropriate. Suppose we are working on a problem involving two pipes and need variables for pipe 1 and pipe 2, or we need variables for the initial and final temperatures. The figure below gives example.

Diameter of pipe 1 $d_1 := 3$ Diameter of pipe 2 $d_2 := 5$

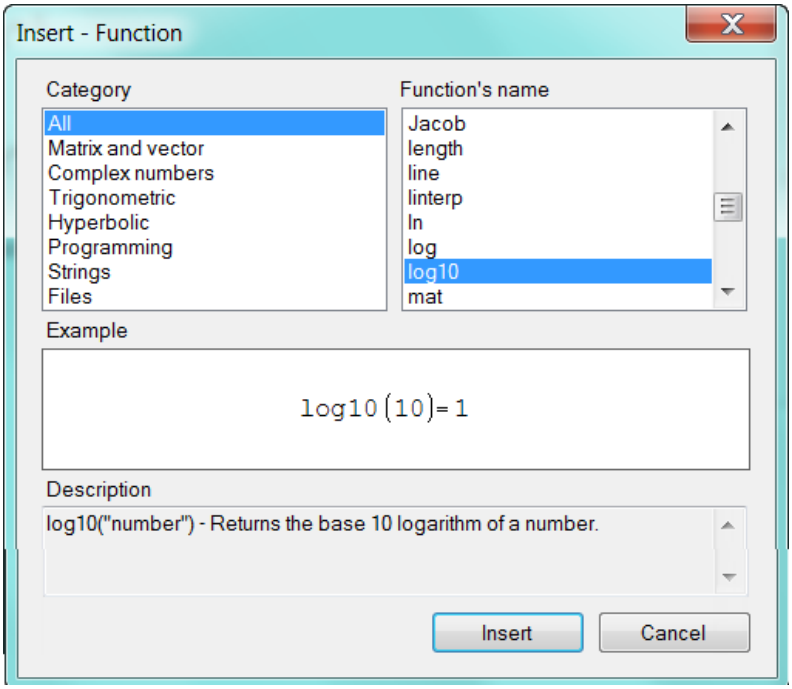
Inital Temperature $T_i := 273$ Final Temperature $T_f := 298$

To make $d_1 := 3$ Type d followed by a period (.) and then continue with : 3

DO NOT confuse the subscripts 1 and 2 in this example with index value to refer to elements of a vector. Such index values are entered in a different way (see later).

Working with Functions

SMath contains a large number of predefined, or intrinsic, functions for both real an imaginary numbers. We will not discuss the last group. The complete list of functions can be found in Appendix A. The entire collection of functions is accessible in SMath by using the Insert | Function menu or by clicking the *f(x)* tool – see figure below. Alternatively, one may just type the function name; the Dynamic Assistant helps get it right.



Some of the functions are placed in categories to make it easier to locate a specific function. These categories may be summarized as:

· The *Complex numbers, Trigonometric, Hyperbolic*, and *Programming* function groups include a relatively small number of easily recognizable functions.
· The *Matrix and vector* group includes a total of 30 functions useful for the manipulation of and calculations with vectors and matrices.
· The *Strings* group includes functions used in manipulating and operations with strings. These can be useful in programming.
· The *Files* group includes functions for reading, writing, and deleting files, as well as for importing data into a worksheet.

The *All* category display the entire collection of functions; it is helpful to group some of those functions according to their applications:

- Functions specific to *SMath Studio*: *eval*, *range*, *sys, error,*
- Functions for real numbers: *abs, exp*, *Gamma, ln, log, log10, mod, nthroot, perc, round, sign, sqrt*
- Functions for algebraic manipulation: *expand*
- Functions for solving equations: *polyroots, solve*
- Functions for Calculus applications: *diff*, *int, product, sum*
- Functions for interpolation: *ainterp, cinterp*, *linterp*
- Statistics functions: *random*

Note there is no separate category for the simple functions for real numbers which are list below:

· *abs* absolute value
· *exp* exponential function
· *Gamma* Gamma (Γ) function
· *ln* natural logarithm, i.e., logarithm of base *e*
· *log* logarithm of any base
· *log10* logarithm of base 10 (typing, for example log(12,2) results in the expression $\log_2(12)$
· *mod* modulus (the remainder after division: mod(12,10) = 2
· *nthroot* the *n*-th root of a number
· *numden* decompose a fraction into numerator and denominator
· *perc* percentage
· *round* rounds to an integer
· *sign* extracts the sign
· *sqrt* square root; gets replaced by the symbol √
· *random* generates a random number

Most of these functions will be familiar already to the user; so typing them is often quicker than finding them in the menu. The Function palette is also a quick way to locate some commonly used functions.

Functions

$x := \exp(3.4)$ Entered by typing x: followed by opening the Insert |Function menu,locating exp in All and clicking the Insert button; then adding 3.4 with typing

$y := \sin\left(\frac{\pi}{2}\right)$ After typing y:, the expression was composed by clicking on items in the Functions and Arithmetic palettes

$z := \tan\left(\frac{\pi}{4}\right)$ Entered entirely by typing

Trig functions work with radians. But it is possible to use Tools | Snippet Manger to set up an area that uses degrees; the author advises that you do not use this feature until some bugs have been ironed out. So we must make appropriate conversion when working in degrees; see also the topic User Defined Functions below.

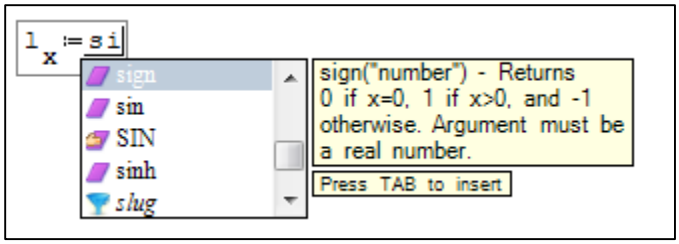
$angle := 0.75$	Here we mean 0.75 radians	$x := \sin(angle)$	$x = 0.6816$
$\alpha := 22.5$	We are working with degrees here	$x := \sin\left(\alpha \cdot \frac{\pi}{180}\right)$	$x = 0.3827$
$z := 0.866025$	We want the inverse sin of z in degrees	$\beta := \operatorname{asin}(z) \cdot \frac{180}{\pi}$	$\beta = 90$

Additional examples of some functions are shown in the large figure below. Please note the following:

a) The argument *n* in *round(x ,n)* must be 0 <n <15; unlike in Excel, it cannot be negative to round to tens, hundreds, etc.

b) The *random* function is dynamic; press F9 to recalculate the page and a new value results. So do not expect your result to match that in the figure or for the result to be the same when the file is reopened.

c) The Dynamic Assistant can be helpful when inserting a function. The user wished to enter l_x:=sin(alpha); when the first part had been typed the Dynamic Assistant popped us as shown below. Note the entry SIN in the popup – this appeared because the user has a user-define function of the name in his page (see User-Defined Functions topic). To select a function, scroll down until it is highlighted and either double click the selection or use the tab key.



Type "abs(-3.25)=" to produce $|-3.25|$

Type "exp(-0.51)=" to produce $\exp(-0.5)=0.6065$

Type "e^(-0.51)=" to produce $e^{-0.5}=0.6065$

Remember the number of decimals displayed
depends on the setting in Tools | Options

Type "log(8,2)=" to produce $\log_2(8)=3$

Type "perc(25,5)=" to find 5% of 25 $\text{perc}(25, 5)=1.25$

The function call random(x) retunes a number between 0 and x
from a uniform distribution of numbers

Type "random(50)=" to produce $\text{random}(50)=29$

For a real number between 0 and 1 with n decimal places

$n:=4$

$$\frac{\text{random}\left(10^{n}\right)}{10^{n}}=0.5993$$

Examples of the round function

$x:=5.67846$ $\text{round}(x, 4)=5.6785$ $\text{round}(x, 2)=5.68$

Use the function palette (or by typing "sum(k^2,1,10"="),find the sum of the squares of the first 10 numbers

$$\sum_{k=1}^{10}\left(k^{2}\right)=385$$

Find the value of the definite integral of $1/x^3$

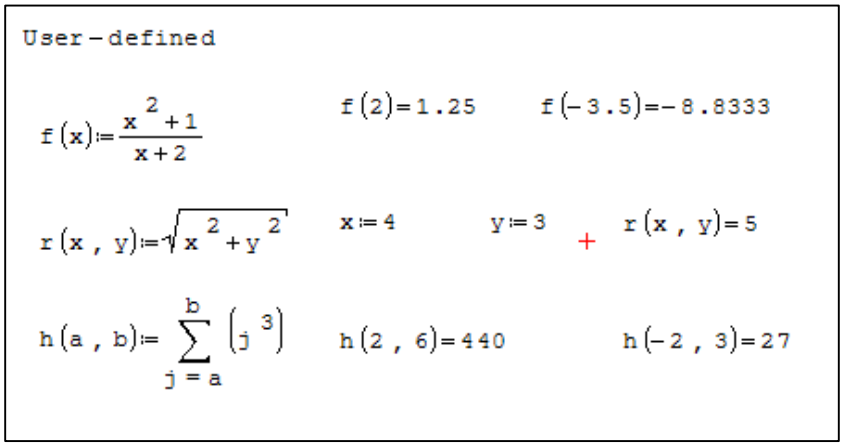
$$\int_{1}^{5}\frac{1}{x^{3}}\,dx=0.48$$

User-Defined Functions

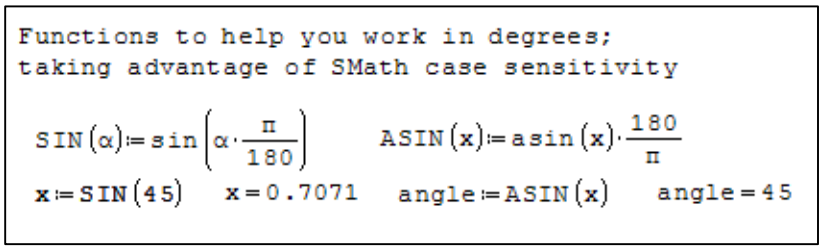
You can define your own functions by using the expression

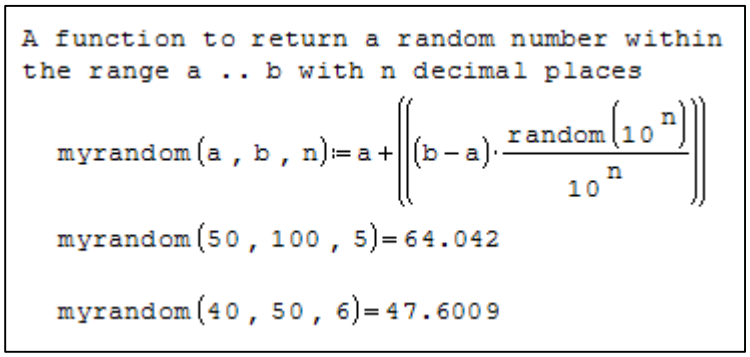
function_name(*argument*(*s*)):= *function_expression*

Examples of definitions and evaluations of functions are shown in the figure below.



Here are two more examples





It may appear that the function is misbehaving since the number of decimals is not equal to n. This is because SMath does not display trailing zeros

Numerical Differentiation and Integration

The figure below shows a simple example of numerical differentiation to find the slope of a curve given by *f(x)* at given points, and numerical integration to find the area under a curve. Also shown is a very simple use of the symbolic evaluation operator → to generate an expression for *df(x)/dx*.

Numerical Differentiation and Integration

$f(x) := -2 \cdot x^2 - 4 \cdot x + 12$

$\frac{d}{dx} f(x) \rightarrow -4 \cdot (1 + x)$

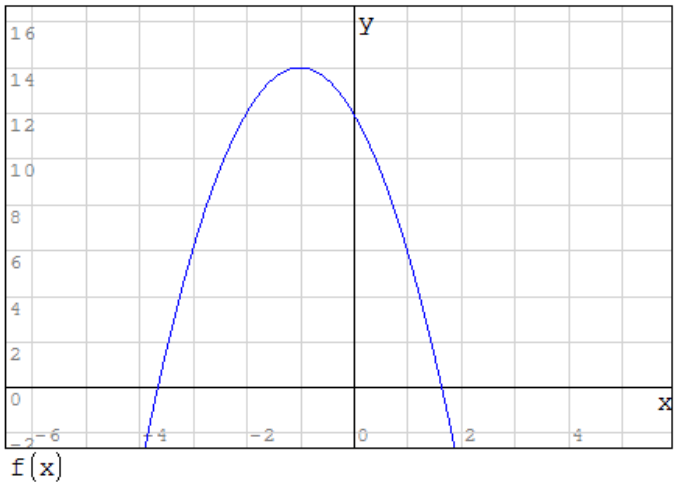
$slope(x) := \frac{d}{dx} f(x)$

$pos_a := -4 \qquad slope(pos_a) = 12$

$pos_b := -1 \qquad slope(pos_b) = 0$

$Roots := solve(f(x), x)$

$Roots = \begin{pmatrix} -3.645751 \\ 1.645751 \end{pmatrix}$



$area := \int_{Roots_1}^{Roots_2} f(x)\, dx \qquad area = 49.387358$

The second math region is made by clicking on the derivative tool in the Functions palette to get the template shown below to the left. The user then fills in the two place holders. Similarly for the numerical integration, you click the integration tool in the same palette, and fill in the four place holders.

$\frac{d}{d\blacksquare}\blacksquare$

$area := \int_{\blacksquare}^{\blacksquare} \blacksquare\, d\blacksquare$

The current version of SMath cannot perform a numerical integration when the variables have units associated with them. This will be fixed in the next version.

As can bee seen in the figure above, SMath can also do symbolic differentiation. A fuller description can be found later in this document. SMath cannot do symbolic integration.

Below is another example of using the numerical integration function.

Find what fraction of the molecules in a sample of oxygen at 600K have speeds between 0.5 and 1.5 of the mean speed

$T := 600 \qquad M := 32 \cdot 10^{-3} \qquad R := 8.31451$

$u_{mean} := \sqrt{\frac{8 \cdot R \cdot T}{\pi \cdot M}} \qquad u_{mean} = 630.07$

$f(u) := 4 \cdot \pi \cdot \left(\frac{M}{2 \cdot \pi \cdot R \cdot T}\right)^{\frac{3}{2}} \cdot u^2 \cdot \exp\left(-\frac{M \cdot u^2}{2 \cdot R \cdot T}\right)$

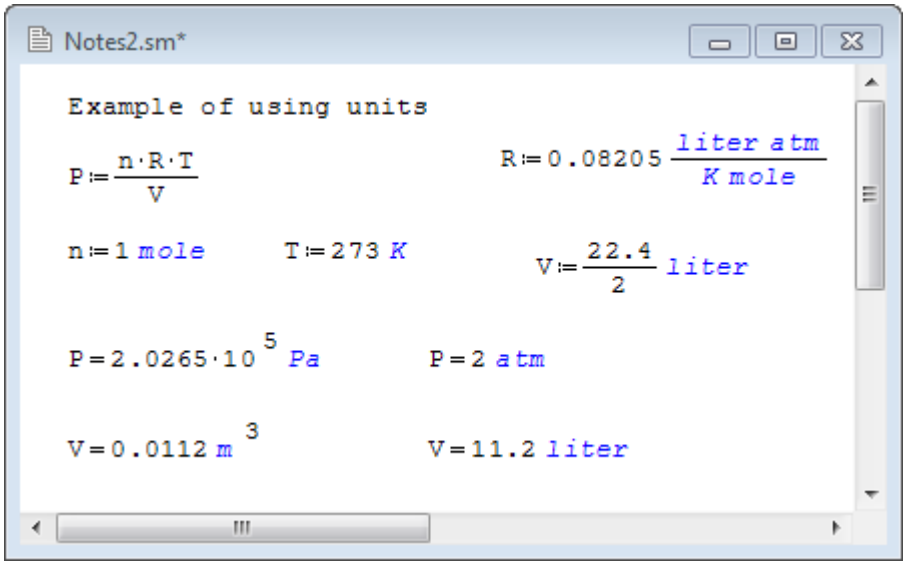
$s_1 := 0.5 \cdot u_{mean} \qquad s_2 := 1.5 \cdot u_{mean}$

$fraction := \int_{s_1}^{s_2} f(v)\, dv$

$fraction = 0.76$

SI units: T in K, M in kg/mol, R in J/K mol

Working with Units

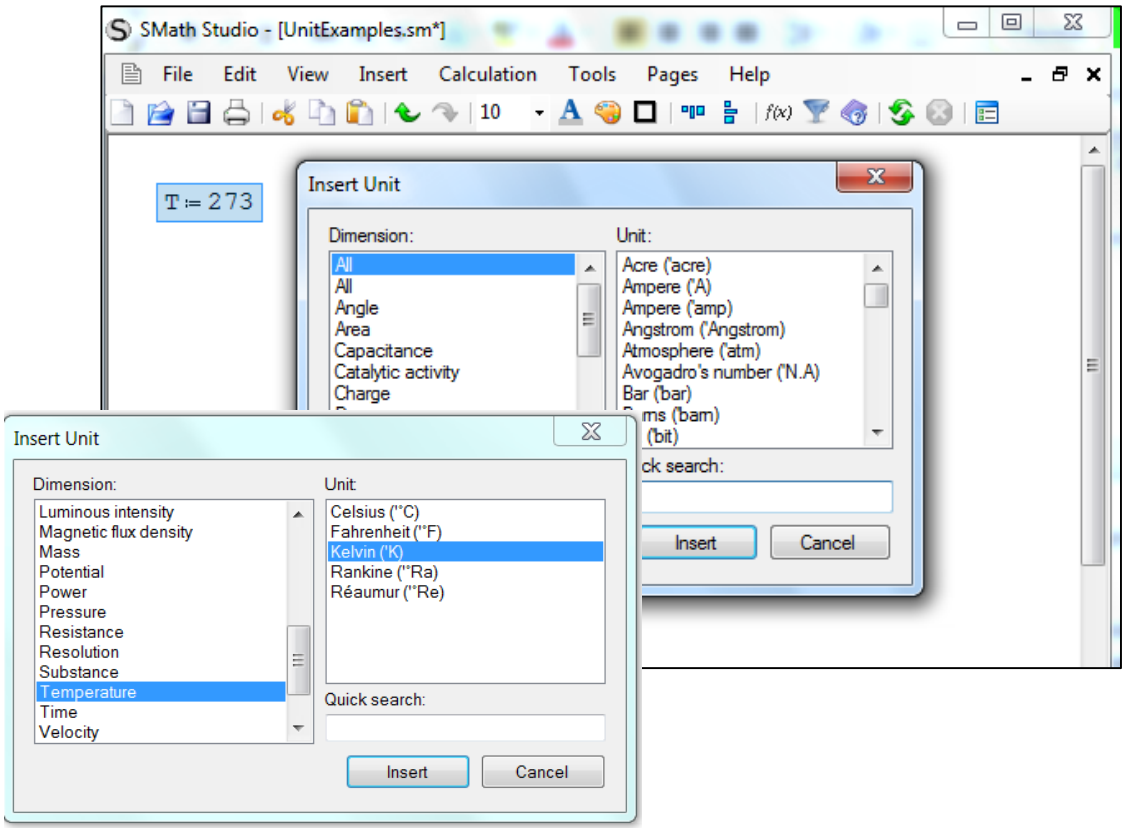


The figure above is a simple demonstration of using units in a calculation which is one of SMath's excellent features.

The example begins with a expression defining how to compute P (you will recognize the Ideal Gas Law); next values are assigned variables *R, n, T* and *V*. For now we will treat *R* as a variable but it is one of SMath's constants: see topic Predefined Constants. Then the page displays the computed value of P (and of *V* which was assigned an expression rather than a value. To the left is the way SMath automatically displays variables with units - using the SI system, while to the right we have had SMath display our units.

We need to learn the two major ways for entering units and how to make conversions from one set to another.

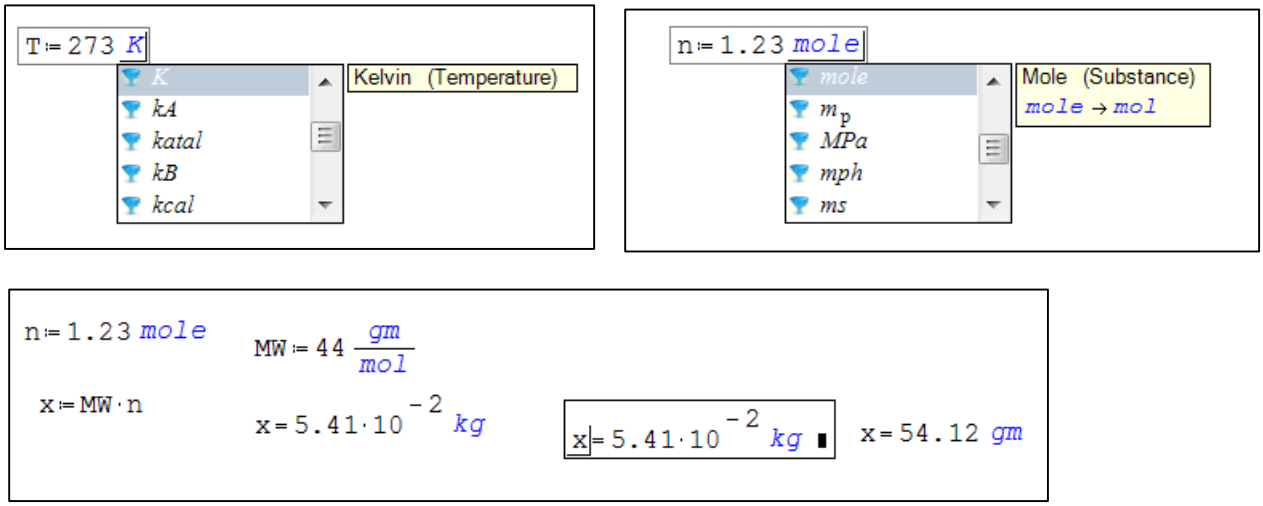
Method 1: To enter T:=273 K begin by typing T:273 in the normal way. Next use Inset | Units to open the units menu (or click the Units tool which resembles a blue funnel, or use the shortcut CTRL+W). At the left of the Insert Function dialog is a list of categories; you can select a category and then, in the right hand window, scroll to select the required unit and click the Insert button, or double click the required unit. You can also type all or part of the unit name in the Quick Search window.



Method 2: Type T:273 to produce T:=273, then type a single quote (aka apostrophe) to open the Dynamic Assistant in units mode – see left hand figure below. You can now scroll to find the required unit, or, to speed up the process, type the first letter of the unit. We want *K*, so we could type an upper case *K*. If you are sure of the unit's' name you can type it in full (right hand figure below); but be careful as there are some pitfalls. Thus the unit for gram is *gm* not *g* but milligram is *mg*.

NOTE: SMath displays units in colour, blue by default

SMath has two simples for mole (*mole* and *mol*) but treats them as synonyms – figure below. It has only one unit for *kmole*, *millimole* and *micromole*. Likewise SMath included *J* and *joule* but the units millijoule, megajoule and gigajoule use only symbols (*mJ, MJ, GJ*, respectively) not words..

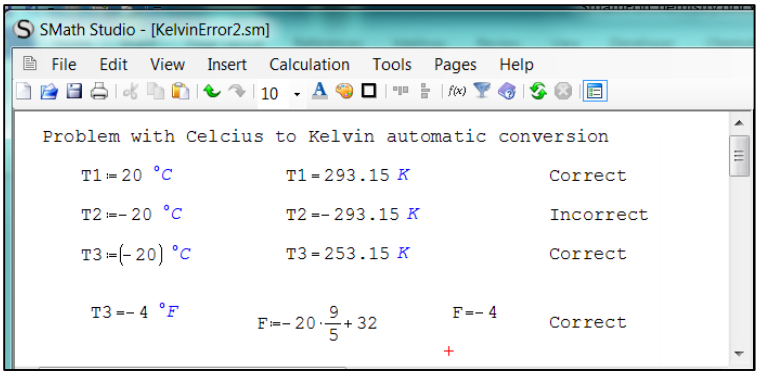


To enter compound units like *gm/mole* or *kW·hr*, the user must type the division or multiply operator (/ or *) These two definitions are synonymous: *J/K·mol* and *J·K^{-1}·mol^{-1}*; but the exponentials will not appear in colour.

In the last figure above the simple calculation $x = MW\,\frac{gm}{mol} \times n\,\text{mol}$ yields a result in kilograms since that is the standard SI unit of mass. The user want the answer in grams. The user has type x= and has not click outside the entry. Note the placeholder to the right. The user can move to this and insert a mass unit (*gm, mg*, etc) using any of the method used above; the simplest thing here is to type *'gm.* You must remember the single quote.

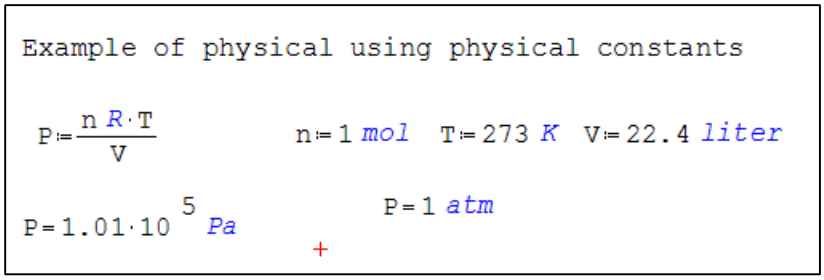
Non-Linear Units (°C → K)

Be aware that there is a bug: SMath (like Mathcad, and Google until recently) has trouble with non-linear units. Most unit conversions are made by multiplying by a *conversion factor*. So 3 inch converts to centimeters with *3 inch × 2.54 cm/inch = 7.62 cm*. But conversion for Celsius to Kelvin or Fahrenheit is said to be non-linear as other mathematical operations are need (addition or subtraction). SMath runs into problems when the non-linear quantify is negative. The problem is solved by the use of parentheses: to produce T:=(-20) °C, begin with T:=20 °C and add the negation after adding the units °C – SMath will automatically add the parentheses.



Predefined constants

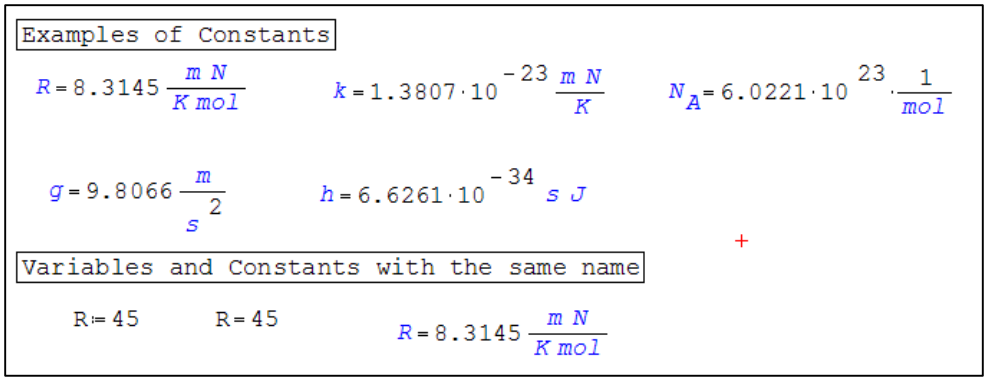
SMath has a number of predefine physical constants. An example of the use is shown below; compare this to the figure at the start of the Units topic. The gas constant R was entered by typing *'R* (note the leading apostrophe) and note also how *R* is displayed in blue in the definition of *P*.



The predefined physical constants are shown in the table below.

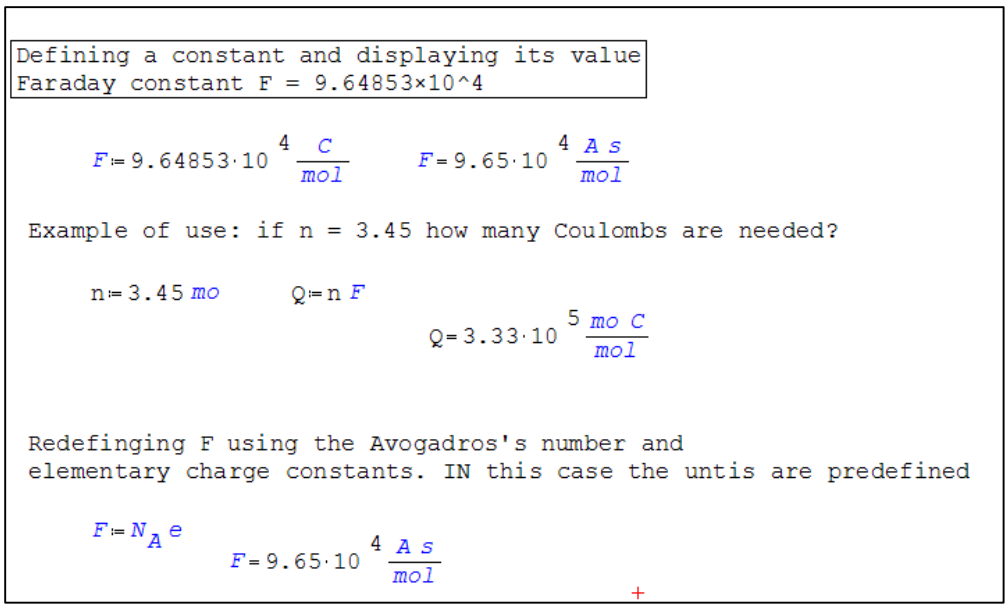
G_N	Newtonian constant	$6.6742867 \cdot 10^{-11}$ m³/kg·s²
g	Gravitational acceleration	9.80665 m / s²
h	Planck's constant	6.6260689633 kg· m²/ s
N_A	Avogadro's number	$6.022141510 \cdot 10^{23}$ mol^{-1}
μ_0	Magnetic constant	4·π·10-7 kg·m/s²·A²
e	Elementary charge	$1.60217648740 \cdot 10^{-19}$ A·s
m_e	Electron mass	$9.1093821545 \cdot 10^{-31}$ kg
m_p	Proton mass	1.67262163783·10= kg
m_u	Unified atomic mass (Dalton)	$1.6605388628 \cdot 10^{-27}$ kg
R	Gas constant	8.31447215 kg·m²/K·mol·s²
k	Boltzmann constant	1.380650424·10-23 kg· m²/ K·s²

To construct the definition of P shown in the figure above: after typing P:=n*, simply type 'R as if you were entering a unit. The constant is displayed in blue just like units. Observe that, while R is defined with units kg·m²/K·mol·s², we can use it with the typical chemistry units in the figure above; SMath looks after the conversion behind the scenes. Note that the actual numerical value of the constants are as shown in the table but that the precision of the numbers displayed depends on the setting in Tool | Options | Decimals.



In the lower part of the figure above we see that, technically, a variable and a constant can share the same name (not exactly, since the constant's hidden name is *'R*). However, your author believes this practice should be avoided due to the dangers of ambiguity.

It is possible to define your own constant as shown below. To begin the definition type 'F (that is why it is displayed in blue)



There are also two predefined mathematical constants: pi and Euler number – see table below.

π	Pi	3.1415926535897932
e	Euler's number e	2.71828182845590452

These mathematical constants are entered as *p* followed by CTRL+G while Euler's number (the value of exp(1)) is entered with a simple *e*. These two values are displayed in the same colour as other variables (default is black.) Of course, you could enter e:=24 (or π= 27) but that would override the predefined values in subsequent calculations (unlike the permitted use of R and 'R mentioned earlier.)

Some Notes on Units (adapted from Prof. Urroz's notes)

Units of the Us Customary System.

In the United States of America the traditional system of units is still very much in use. Therefore, *SMath Studio* includes in their collection of units such customary units as *horsepower* (see example above), *lb* (pounds), *ft* (foot, or feet), *in* (inches), etc. For example, we can display the values of the speed of light in vacuum and the standard acceleration of gravity in SI and US units:

$c = 299792458 \frac{m}{s}$ $c = 983571056.430446 \frac{ft}{s}$ $g = 9.80665 \frac{m}{s^2}$ $g = 32.17404856 \frac{ft}{s^2}$

It is incorrect to call these the English or Imperial System as the US developed its own standards before the Imperial system was standardized; furthermore some quantities differ in the two systems and SMath uses the US values.

$a := 1\ gal$ $a = 3.78541197\ liter$ 1 Imperial gallons = 4.54609188 litres

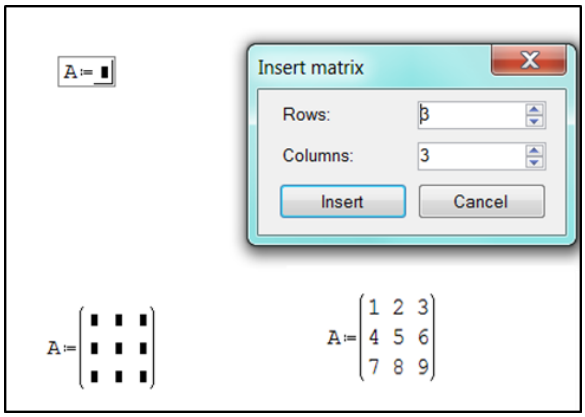
Are pounds units of mass or force?

Modern usage in the physical sciences requires that the pound, *lb*, be used as a unit of force, with the corresponding unit of mass being the *slug,* defined as *1 slug = 14.5939 kg*. However, in buying produce (e.g., meat, grain, etc.), the pound can be used as a unit of mass with the conversion *1 kg = 2.2 lb*. In *SMath Studio* the pound, *lb,* as well as the ounce, *oz*, are used as a units of mass (Note: *1 lb = 16 oz*). The corresponding unit of force is the *pound-force*, *lbf*, defined as *1 lbf = 4.4482 N*. Thus, in physical science applications use the *pound-force*, *lbf*, when you are required to use *pounds* as a unit of force. Another commonly used unit of force is the *kip* (*kilopound*) equal to *1000 lbf*.

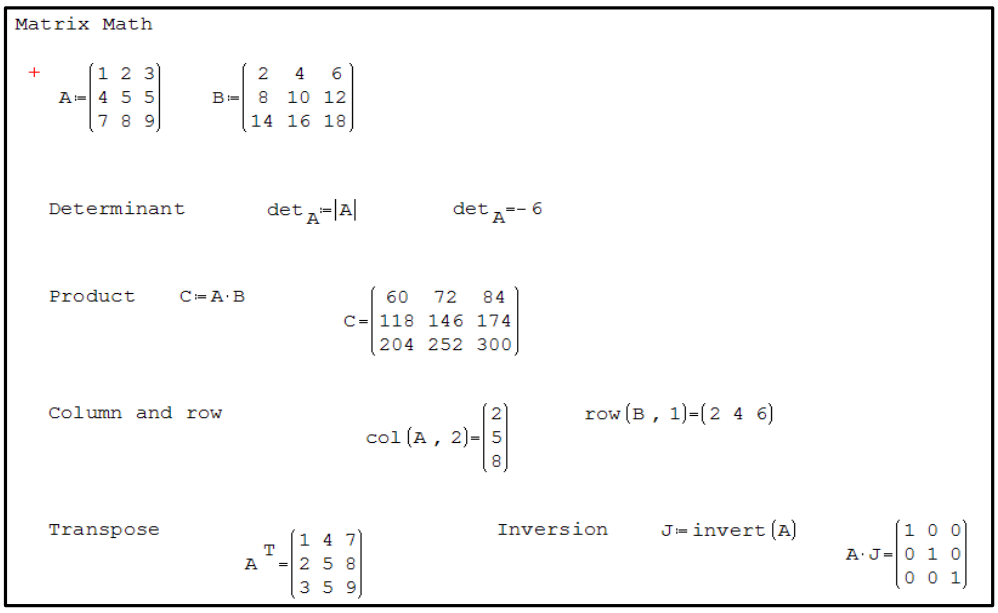
Matrices and Vectors

Matrix operations

The Insert Function dialog has a category called Matrix and Vector; some of the functions are available in the Matrix palette. To define matrix A, we type A:= and then click the first icon in the Matrix palette to bring up a dialog where we specify the number of rows and columns. Next we enter values for all elements by successively clicking on the placeholders.. The sequence is shown below.



The next figure illustration some simple matrix operations. The reader is encourage to experiment with others.



Example of Matrix Math

A system of linear equations may be represented in matrix form. Thus the system of two equations

$x - 2y = -1$

$3x + 4y = 17$

may be represent by

$$\begin{bmatrix} 1 & -2 \\ 3 & 4 \end{bmatrix} \begin{bmatrix} x \\ y \end{bmatrix} = \begin{bmatrix} -1 \\ 17 \end{bmatrix}$$

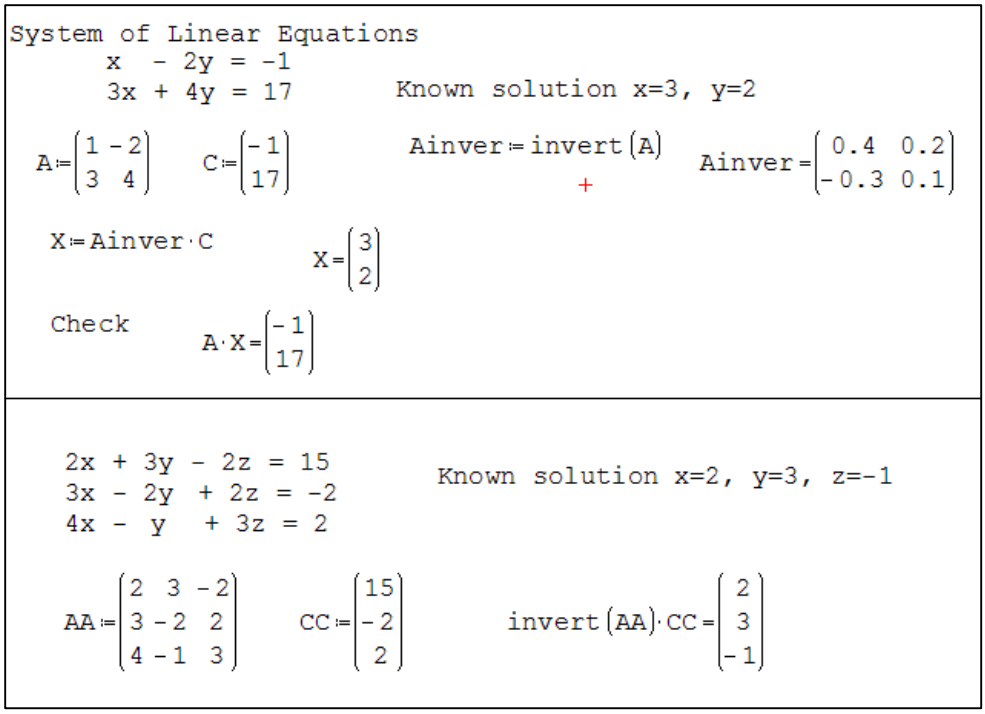
Let us write this in the form: $AX = C$

Multiply both sides by A^{-1} giving $A^{-1}AX = A^{-1}C$

But since $A^{-1}A = I$, this becomes $IX = A^{-1}C$ (I being the identify matrix)

We know that IX = X, therefore $X = A^{-1}C$

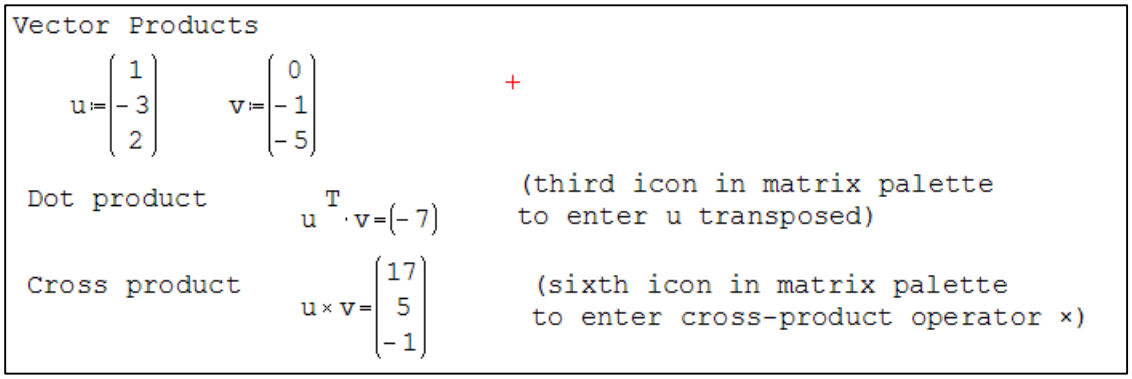
From this we see that the value of the X matrix may be obtained by computing $A^{-1}C$. The following figure shows a SMath implementation of this method.



The line between the two calculations is generated with Insert | Separator: note that all this does is draw a line, it has no effect on variables.

Vector Operations

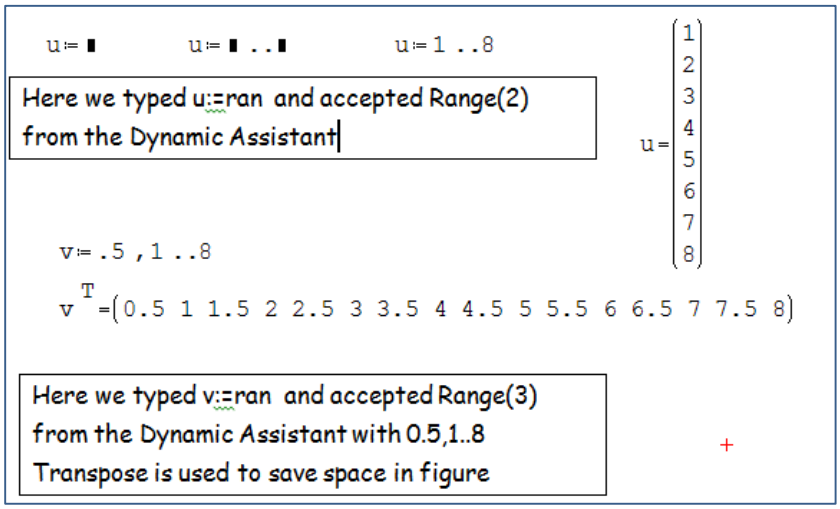
In the first figure below we define two vectors and then find the dot and cross products. The vectors are defined as with matrices but specifying one column. Note that one may define a row vector but the SMath functions are designed for column vectors.



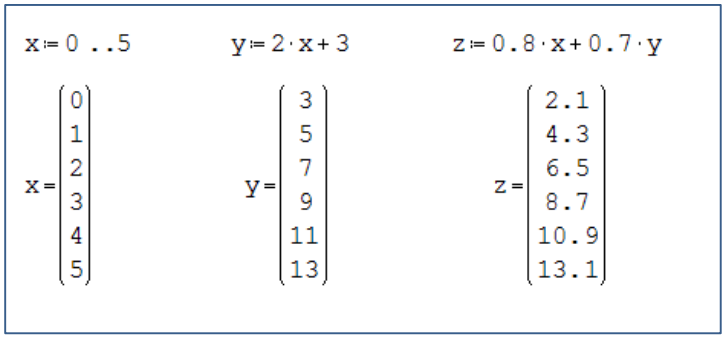
The range functions may be used to generate vectors with numerical elements.

- *range*(*start*, *end*): shows up in the worksheet as *start..end*, and produces a vector whose elements are *start*, *start* + 1, *start* + 2, etc. The last element is the lowest value smaller than *last* by less than one.

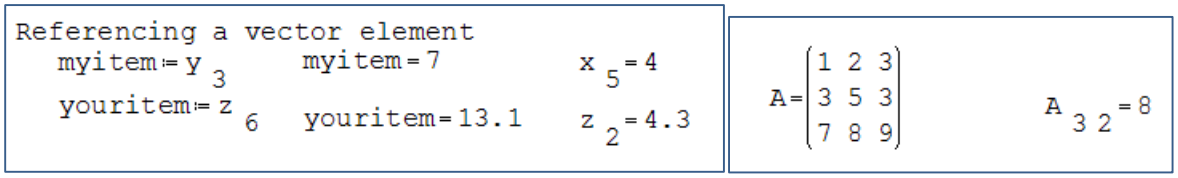
• *range*(*start, end, start + increment*): shows up in the worksheet as *start, start + increment .. end*, and produces a vectors whose elements are *start, start + increment, start + 2 increment,* etc. The last element is the lowest value smaller than *end* by less than *increment*.



The next figure shows how to make linear combinations of vectors. Note that it must be linear, so $y:=2x^2-4$ would not yield a y vector but a scalar value.

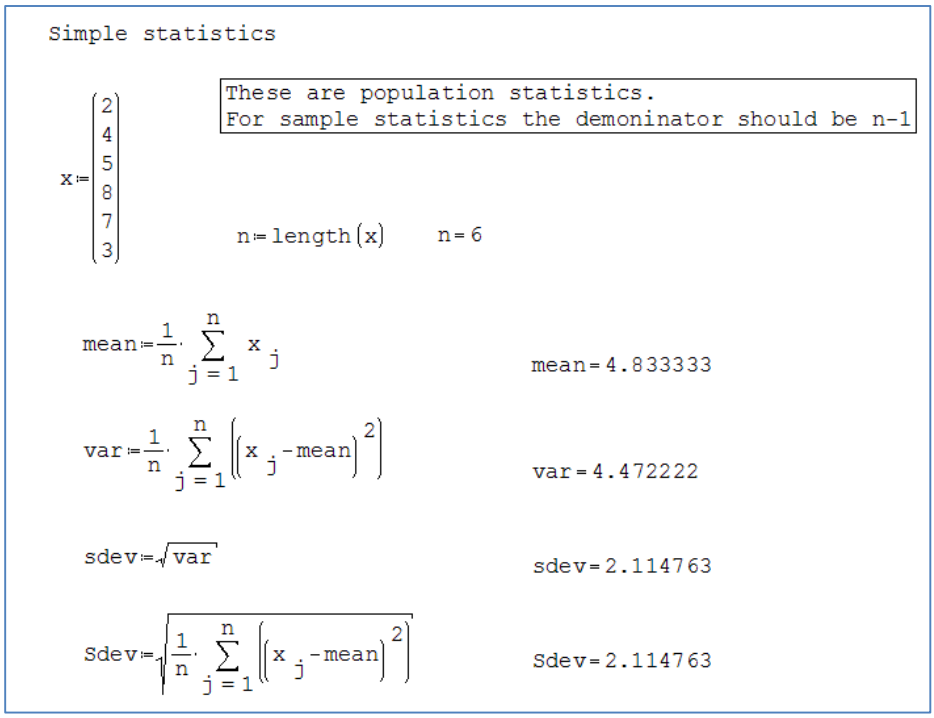


We can reference a single item in a vector using subscripts. To get x_5 we type x[5. In the same way we may reference a single matrix element; to get $A_{3\,2}$ we type A[3,2 and SMath removes the comma.



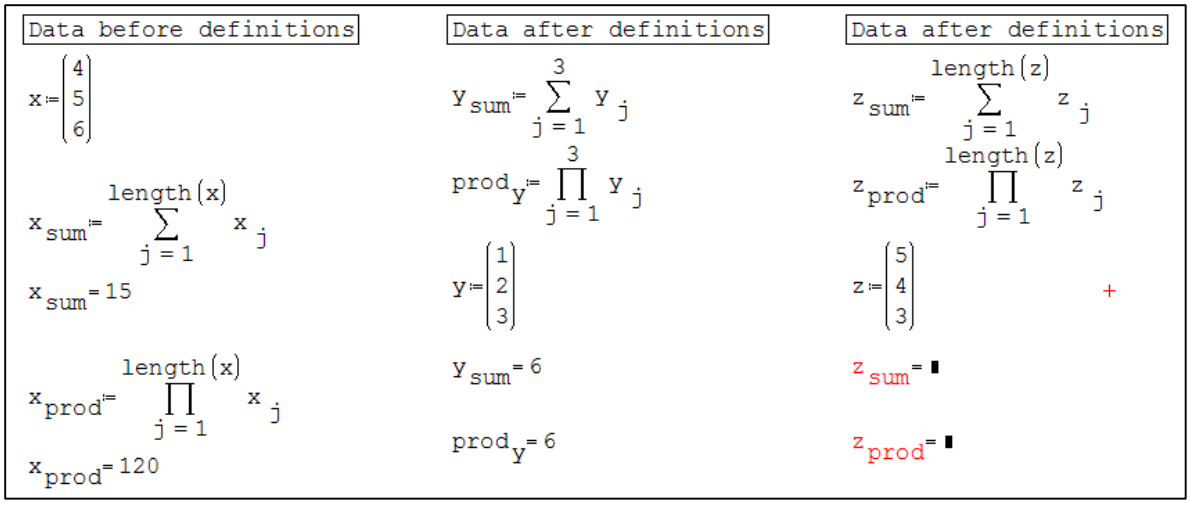
Simple Statistics

SMath contains no function for computing elementary statistics such as mean and standard deviation. However, as the figure below demonstrates they are readily computed. We show two formulas for the standard deviation; the second is useful when the variance is not required.



Note on Matrix/Vectors calculations

At the start of these notes we talked about SMath being 'top down'. We cannot display a result on a worksheet at some point above the place where the data is entered. In the leftmost calculation in the figure below we have this layout: data, definitions, and displays. As expected, this works. Generally, however, we can put definitions at the top of page, followed by the data and then the statements that display the results. Look at the centre calculation; it also works. But the rightmost calculation fails. If you hover the mouse pointer over either definition a tooltip states "z is not defined". Originals the z vector was above the definitions and correct results were displayed. When the vector was repositioned, the display statements turned red. If we deleted them now we would be unable to enter z_{sum}=; SMath would keep replacing the = with := and displaying zsum:=■



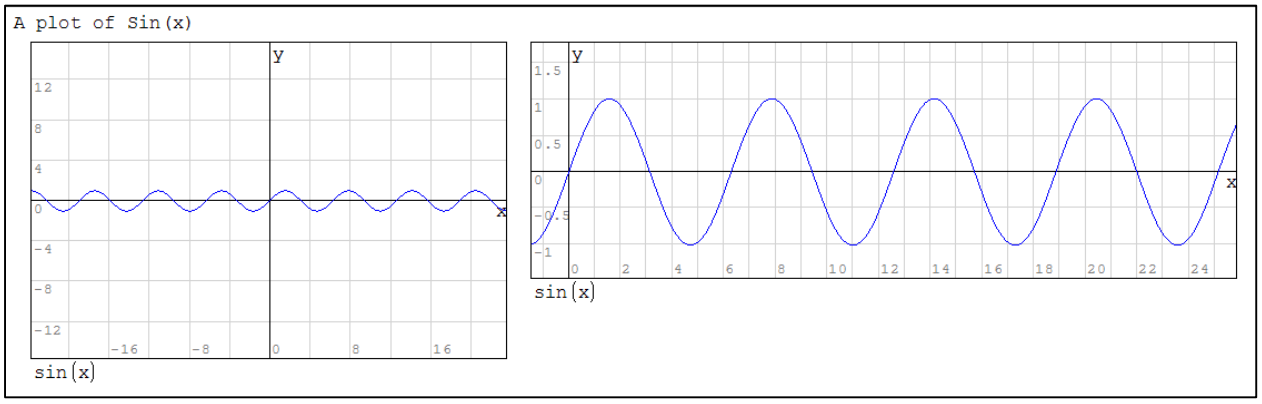
The problem lies with using *length(z)* in the definition. Using n:=length(z) and entering *n* as the upper limit in the summation does not help. The moral seems to be "it the definition mysteriously fails, put the data first !"

Drawing graphs

The SMath graphing tools are somewhat limited but are useful for everyday work other than creating graphs for publications. SMath can generate 2- and 3-dimensional graphs. We will explore only the 2D graphs.

We begin with a very simple example: a plot of sin(x).

1) Click on point in your worksheet where upper left corner of graph will go.
2) Click on the **2D** option in the Functions palette or use the Insert | Graph | 2D menu option.
3) Type the function name in the placeholder in the lower left corner of the graph. So type *sin*(x).



In the figure above, the graph on the left is what SMath produced; the one on the right has been modified. The methods for modifying a graph are listed below; the phrase "activate the graph" means click on the graph so that the handles are displayed:

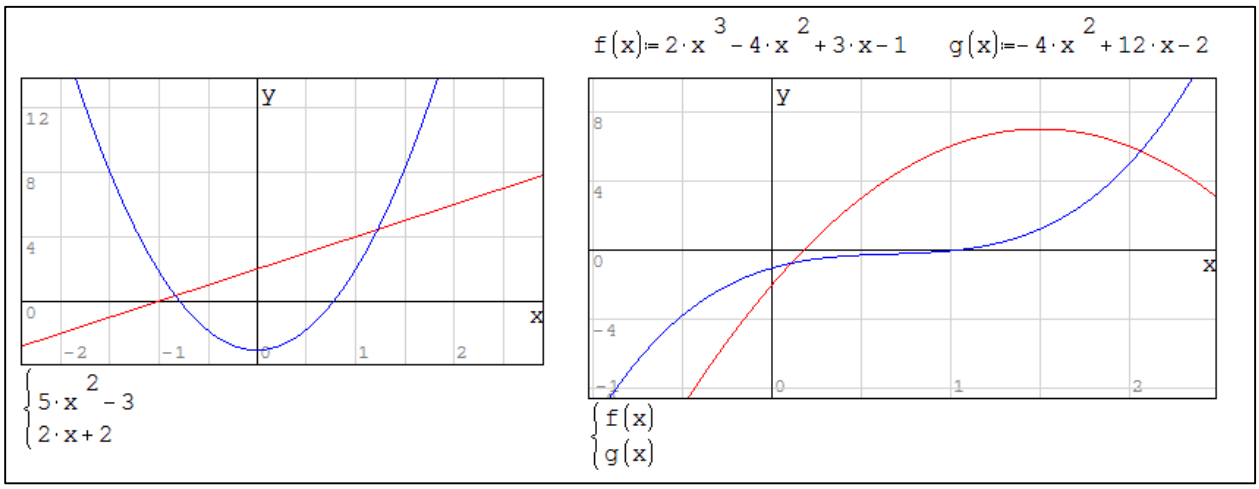
1) Activate the graph and drag one of the handles (small solid squares) to resize the graph.

2) Activate the graph and drag the mouse pointer reposition the origin.
3) Activate the graph and scroll the mouse wheel to rescale both axes.
4) Activate the graph and scroll the mouse wheel while hold CTRL to rescale the y-axis.
5) Activate the graph and scroll the mouse wheel while hold SHIFT to rescale the y-axis.

If you want to relocate the graph on the page: click outside the graph and drag the mouse until the graph is selected (blue background with default setting); now drag it to new position. To copy or delete a graph you need to select it in the same way. Alternatively, a graph (or any entry) can be move by dragging it by the outside frame when the mouse pointer is a four-pointed arrow.

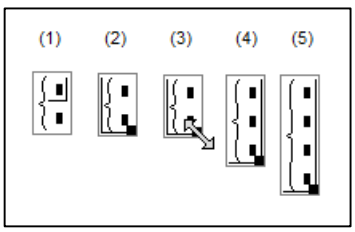
Sometimes when you plot a function the graph appear empty. In such cases it is necessary to rescale the graph to find the plotted data. Try plotting x^2+75*x-2500; you will see nothing until the x scale is about ±64 and the y scale ±1024.

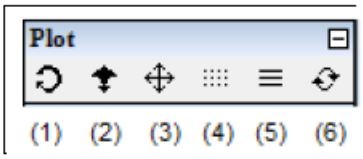
The next figure shows two graphs each having 2 functions.



To insert two functions: first make a blank chart with the 2D tool in the Functions palette, then Select the Multiple Values tool which is the fourth item in the bottom row of the Functions palette. Finally, enter a function in each of the two placeholders produced by the Multiple Values tool.

The Multiple Values icon produces two entries by default. However, you can insert more entries by following the procedure shown to the left below. To go from (1) to (2), select the brace not on a placeholder.

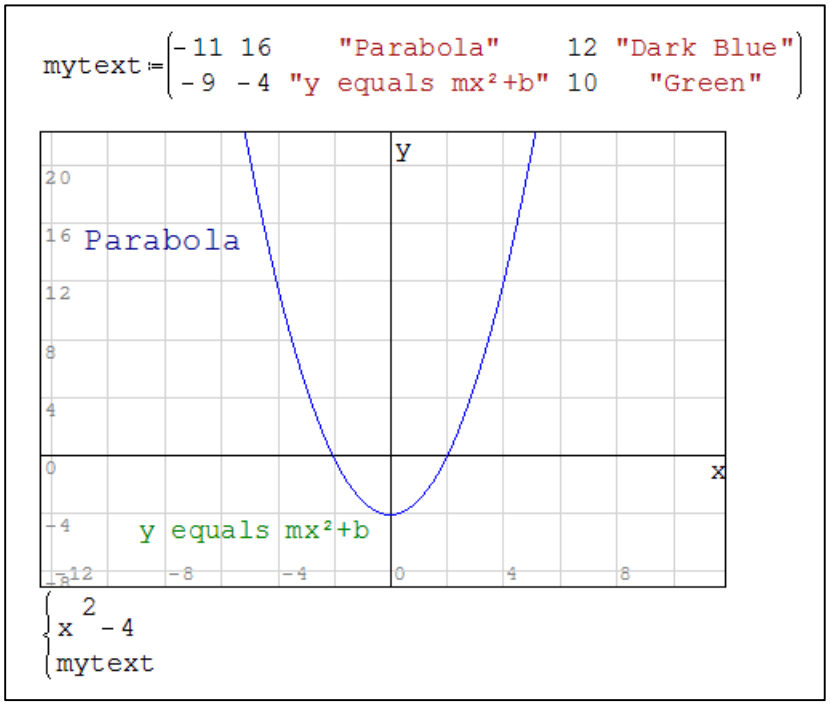




The Plots palette has the following icons

(1) Rotate: Rotate a 3D graph only
(2) Scale (not necessary to use)
(3) Move (not necessary to use)
(4) Graph by points: show points instead of lines
(5) Graph by lines: show lines (default)
(6) Refresh: restore to original version of graph

In the figure below we see a method of adding text to a file. The variable *mytext* is a 2 × 5 matrix: the first two elements in a row specify the starting position (x, y), then the text to be displayed, followed by the font size and colour. SMath offers over 140 colour choices from *aliceblue* to– see
http://en.smath.info/forum/default.aspx?g=posts&m=3663.
If you happen to misspell the name of a colour it will result in black being used.

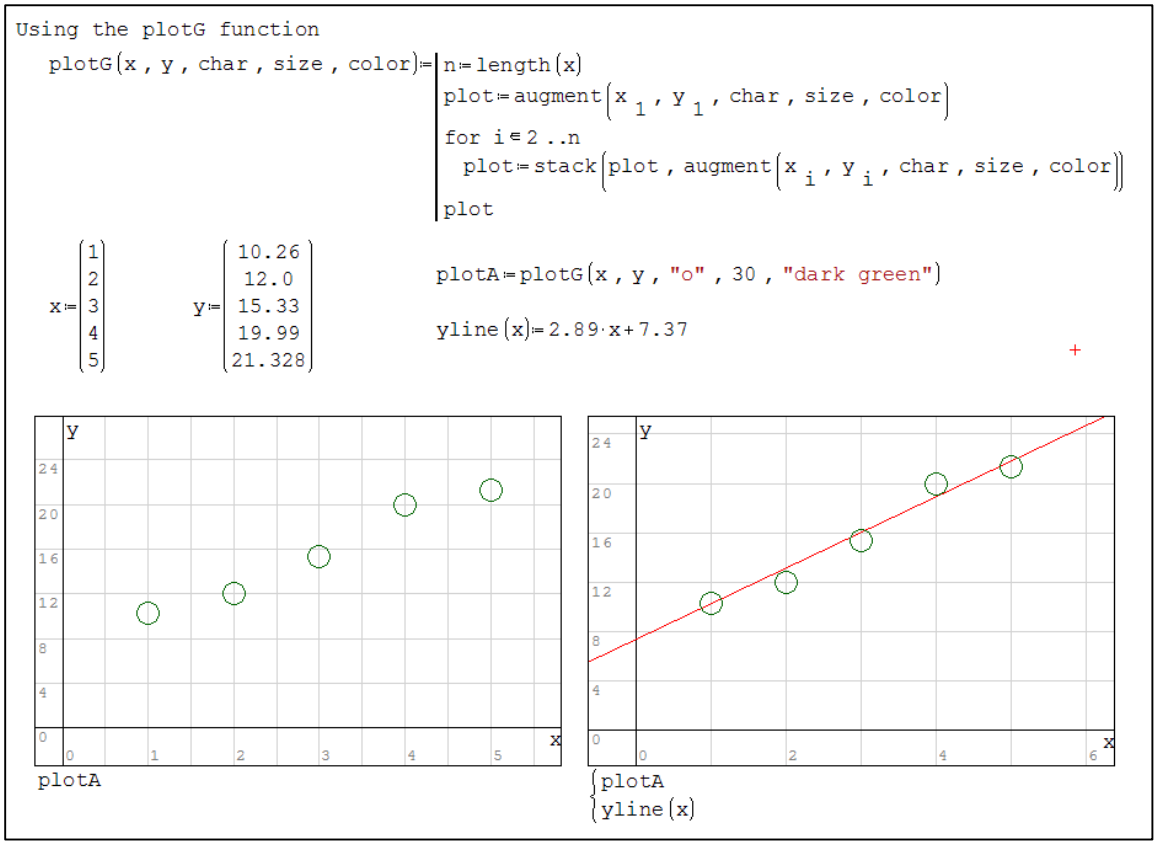


In the following figure we use the plotG function which was made available by Prof. Radovan Omorjan in the SMath Studio Forum. The program takes vectors of values (x, y) and creates a plot matrix for the data using a specified character (*char*), *size*, and *color*. This function operates only in SMath Studio version 0.89 release 8, and later. We will not attempt to explain how it works!

Use only the lower case characters *o* and *x*, and the asterisk (*) as the value of *char*; other characters are plotted incorrectly – too low and too far to the right.

It is suggested that you copy and paste the *plotG* function rather than try to code it: it is available in the file LinearRegress.sm. This example also shows how a function (here *yline*) may be plotted as the same

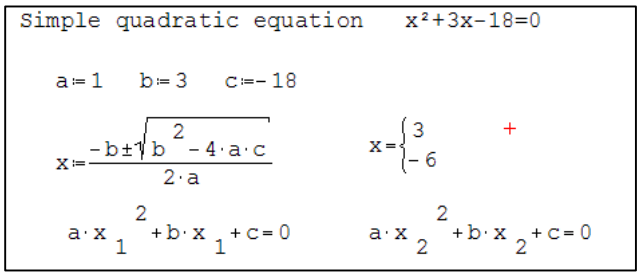
time as *plotA*. We could add this to the LinearRegression worksheet shown above in the Matrix section of these note. Of course, in that case we would use the expression *yline(x):=slope·x + intercept* rather than numeric values; and we would not need the two vectors since these are already defined.



Solving equations and finding roots

When we speak of solving an equation we mean: find the value(s) of x that satisfy the equation f(x) = g(x). For example we might wish to know what value of x gives the function $\exp(x^2+x)$ the value of 10. When we speak of finding the roots of f(x) we mean finding what values of *x* makes f(*x*) = 0.

We will start this topic showing the solution to a simple quadratic equation.

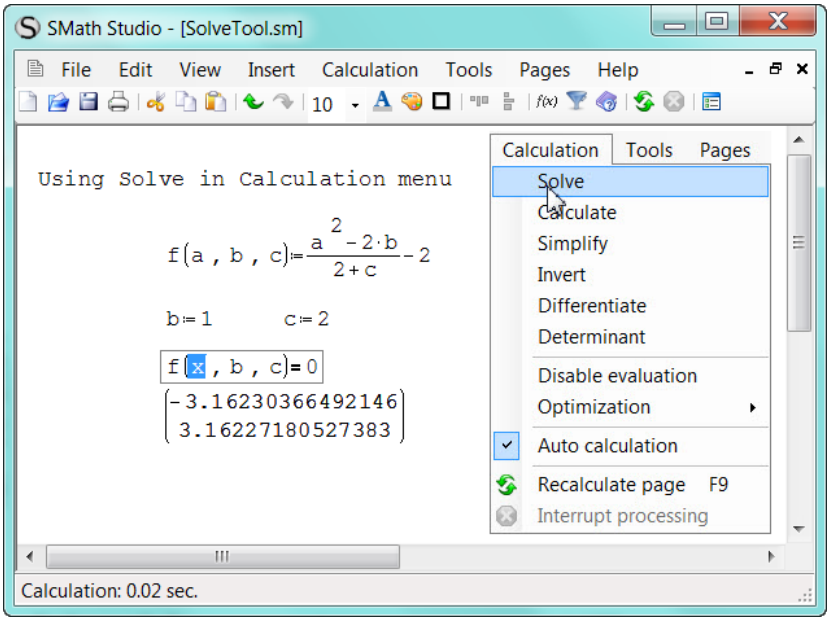


In the centre we have the well-known quadratic solution. The ± symbol comes from the top row of the Arithmetic palette. To the right the user typed *x:=* and SMath gave both values of the x variable in vector

form. On the last line we see how to refer to an individual root (x_1 and x_2); used here to check that $ax^2 + bx + c$ does in fact equal zero with each x-value: clearly $x_1 = 3$ and $x_2 = -6$.

But in some circumstances there is analytical solution and we must rely on numerical approximations. SMath has three functions for these operations: *roots, polyroots* and *solve*. In addition there is a tool within the Calculations menu for solving equations.

Below is an example of the Solve tool from the Calculations menu.



Referring to the figure above, the user entered the line of text and the next three lines. Carefully note that the entry f(*x, b, c*)=0 has a Boolean equality symbol inserted using the Boolean palette. Such symbols appear bold on the page. Then with the *x* with the function selected (hence is shows in blue box), the Calculation menu was opened and Solve was clicked. SMath responded by inserting the two valued vector with the solutions to the equation. Note how the results are given to the full 15 digit precision notwithstanding that in Tools | Options decimal places were set to 4.

Note: the results from this tool are static; suppose you change the value of *c* from 2 to 4, the roots will not alter and will now be incorrect.

This tool is not trouble free. For example is you enter *x–20=5* (note the Boolean =), select the x and use the tool then SMath response with an error message "No real roots". However the equation *x·20=5* correctly returns the value of 0.25

The roots function has two forms; here are simplified syntaxes for each: (a) *roots(function, variable)* and (b) *roots(function, variable, guess).* We shall see that, whenever possible, it is advisable to provide a value for *guess* – to use the 3-argument form rather than the 2-argument form. The *guess* value tells SMath were to start seeking a solution.

In the example below we find the two roots of $e^x + x^2 - 5x = 0$. The graph indicates roots at about 0.25 and 1.75. The expression *root1:=roots(f(x),x)* returns the first value. When we give a value to *guess*, as in the next two calls to *roots*, we can control which root gets returned. Note that the *roots*() results do not make *f(x)* equal exactly zero but results that do so close to the precision of the system.

Using the roots function

$f(x) := e^{x} + x^{2} - 5 \cdot x$

$root1 := roots(f(x), x)$

$root1 = 0.28$

$root1 := roots(f(x), x, 0)$

$root1 = 0.28$

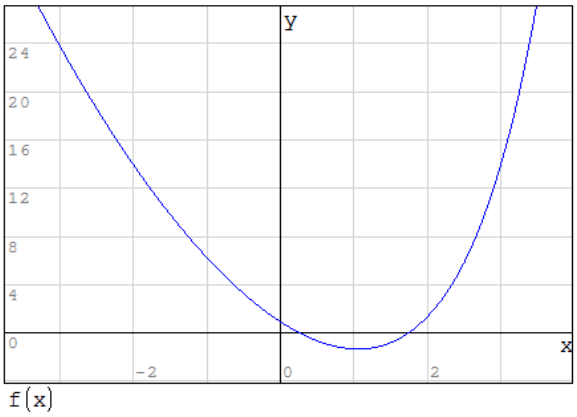
$root2 := roots(f(x), x, 4)$

$root2 = 1.73$

check

$f(root1) = -4.56 \cdot 10^{-15}$

$f(root2) = -5.8 \cdot 10^{-15}$



It is important that the variable to be solved (here the variable x) has not previously been given a numeric value. Try typing x:=1 anywhere on the page; all calls to the *roots* function following that assignment will fail. For this reason you will see expression like *root1:=roots(f(_x),_x,0)* on the SMath website. It is unlikely that the variable name *_x* has been assigned on a page before the calls to *roots()*.

We have over-simplified things a little by talking about *variables* within the roots function. Actually each argument is technically a vector. Hence we can use the roots function to solve systems of equations. The example in the next figure is taken from the SMath website (http://smath.info/wiki/(S(onncf2bm541kyljyy3fiizeg))/GetFile.aspx?File=roots/roots.png

System of non-linear equations

Method 1

$f1(x, y) := x^{2} + y^{2} - 5$ $\quad f2(x, y) := 5 \cdot x - 2 \cdot y - 1$

$r := roots\left(\begin{pmatrix} f1(x, y) \\ f2(x, y) \end{pmatrix}, \begin{pmatrix} x \\ y \end{pmatrix}\right)$ $\quad r = \begin{pmatrix} 1 \\ 2 \end{pmatrix}$

check: $f1(r_1, r_2) = 0$ $\quad f2(r_1, r_2) = 0$

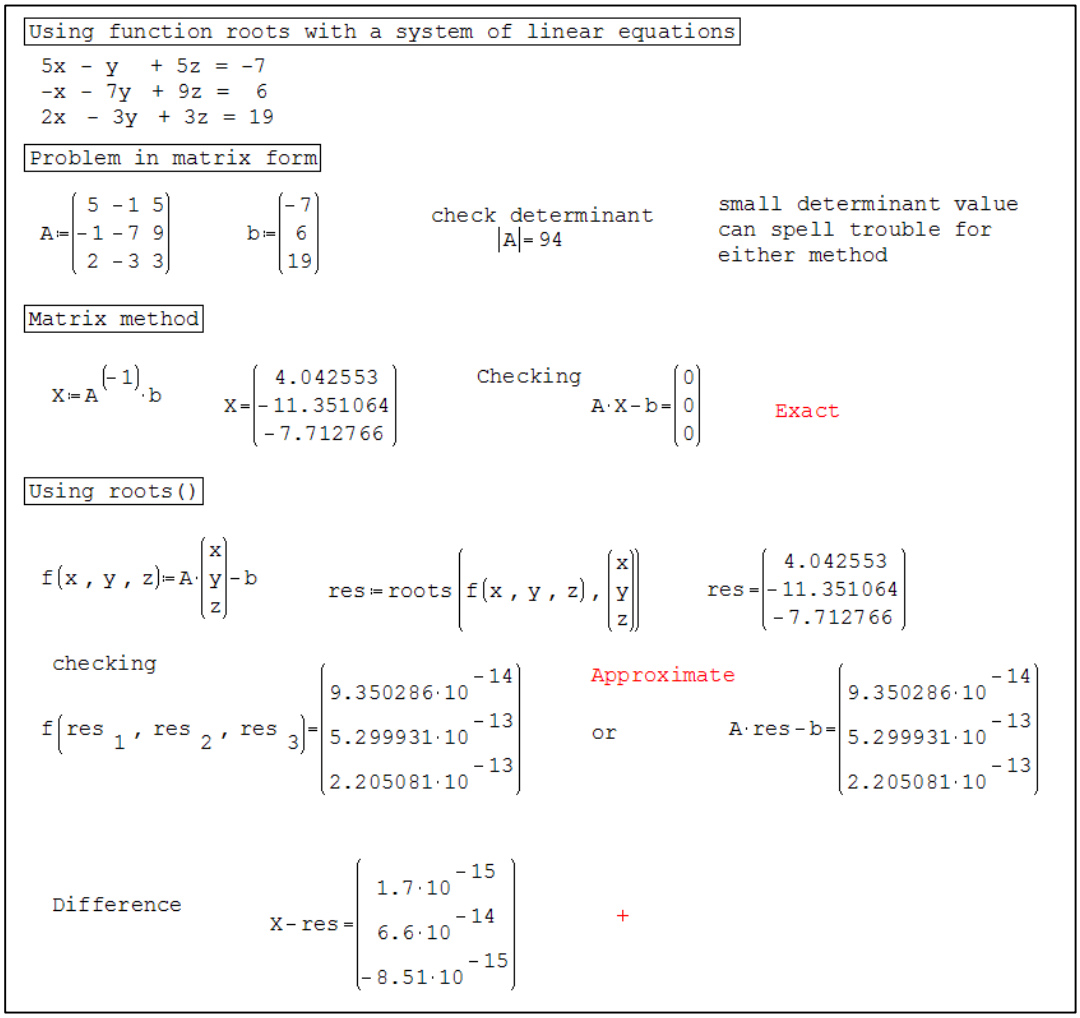
Method 2 - vector form

$F(x) := \begin{pmatrix} x_1^{2} + x_2^{2} - 5 \\ 5 \cdot x_1 - 2 \cdot x_2 - 1 \end{pmatrix}$ $\quad x := \begin{pmatrix} x_1 \\ x_2 \end{pmatrix}$ $\quad guess := \begin{pmatrix} 0 \\ 0 \end{pmatrix}$

$roots(F(x), x, guess) = \begin{pmatrix} 1 \\ 2 \end{pmatrix}$

In Method 1 we build vectors within the roots function while in Method 2 we construct the vector outside the function and within the function we use their names

In the section dealing with matrices we saw how to use matrix math to solve a system of linear equations. The roots function can also be used as shown below but the algebraic method is preferred. The two sets of results are the same to the displayed 6 decimal places but the last region shows they differ slightly; note how the check for the first method yields three zeros while that for the second method yields small numbers.



In the next figure we see how solve() may be used to find the roots of a cubic equation. There is an analytical method for cubics but it is complicated and fraught with problems. Note that the first argument in the solve() example is simply *f(x).* We could have used *f(x)=0* (with a Boolean =) but it is not necessary. The example also shows how to limit the result to positive (strictly speaking to non-negative roots since zero could be returned) by specifying a range with the use of the third and fourth arguments; and how to discover then f(x) evaluates to 64 with f(x)=64 for argument-1; the equality symbol is from the Boolean palette. The graph is used merely to show that the solve() results are correct!

Function solve() to solve cubic

$f(x) := 6 \cdot x^3 + 7 \cdot x^2 - 94 \cdot x + 105$

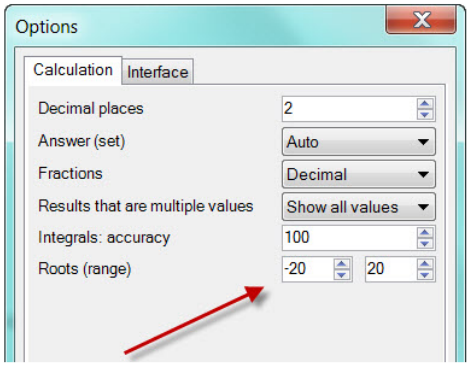
Known roots are: -5, 3/2 and 7/3

$X := solve(f(x), x)$ $\quad X = \begin{pmatrix} -5 \\ 1.5 \\ 2.33 \end{pmatrix}$

Using just two arguments, solve() will search in the region set in Tools | Options Roots
Default is -20 to +20

checking

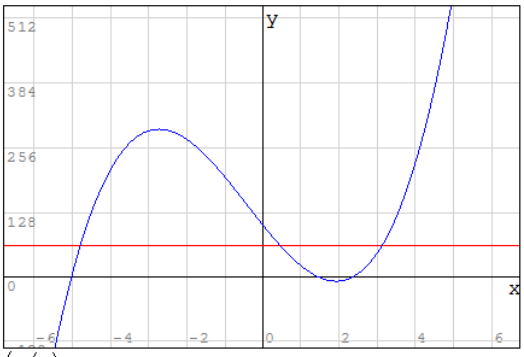
$f(X_1) = 0 \quad f(X_1) = 0 \quad f(X_1) = 0$



Using the 4-argument form to restrict roots to positive values

$Q := solve(f(_x), _x, 0, 10)$

$Q = \begin{pmatrix} 1.5 \\ 2.33 \end{pmatrix}$



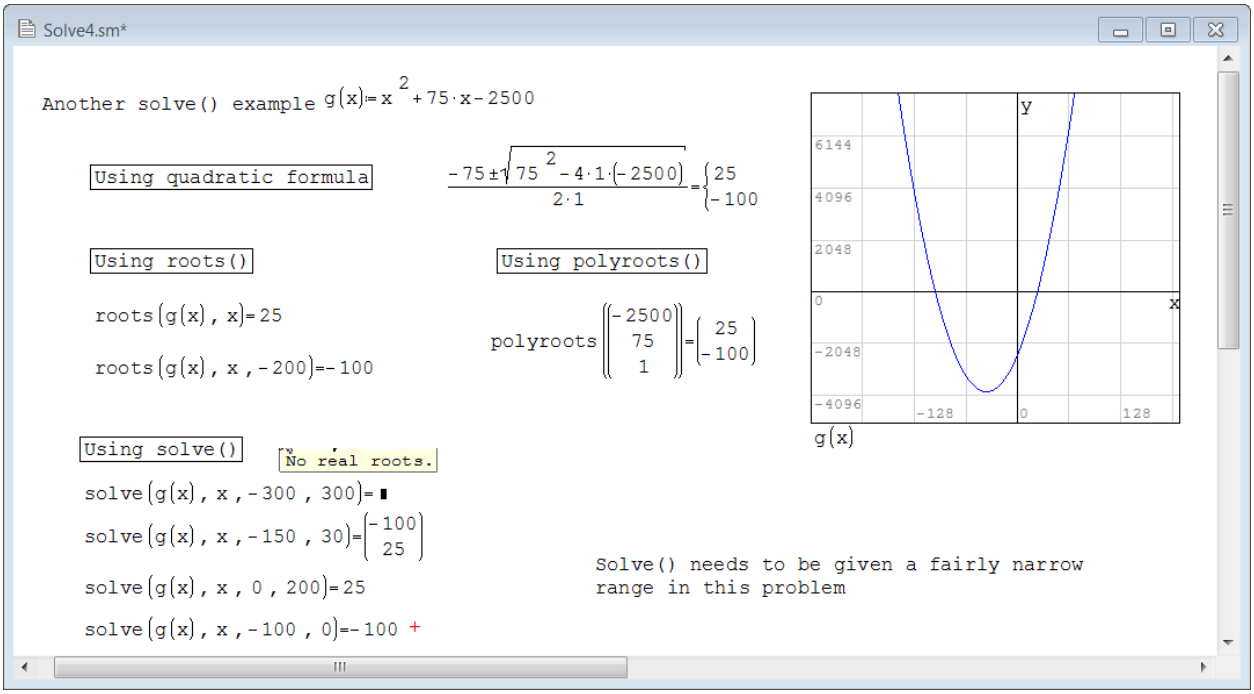
$\begin{cases} f(x) \\ 64 \end{cases}$

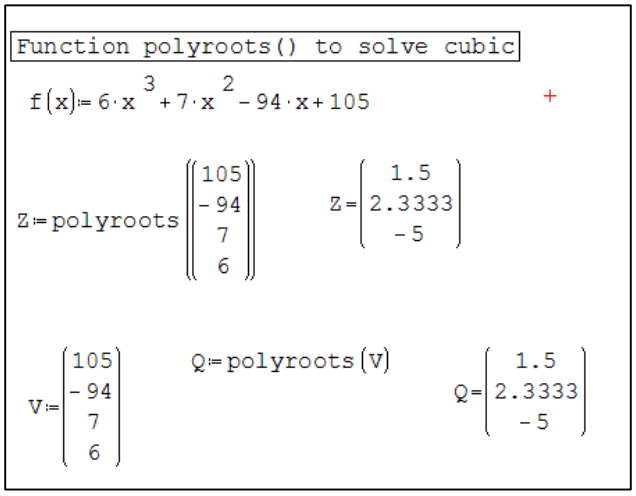
Using a Boolean = in the first argument

$solve(f(x) = 64, x) = \begin{pmatrix} -4.76 \\ 0.46 \\ 3.14 \end{pmatrix}$

There are problems with *solve*() when the roots are far apart as is demonstrated below. Since we are finding roots of a quadratic it is easy to get the correct answers. So we can use them to demonstrate the results from *roots*() and *polyroots*(); details of *polyroots*() follow. Solver() is disappointing; it gives results only when given narrow ranges. While *solve(f(x), x, 0, 100)* fives the result of 25, *solve(f(x), x, 0, 200)* fails.

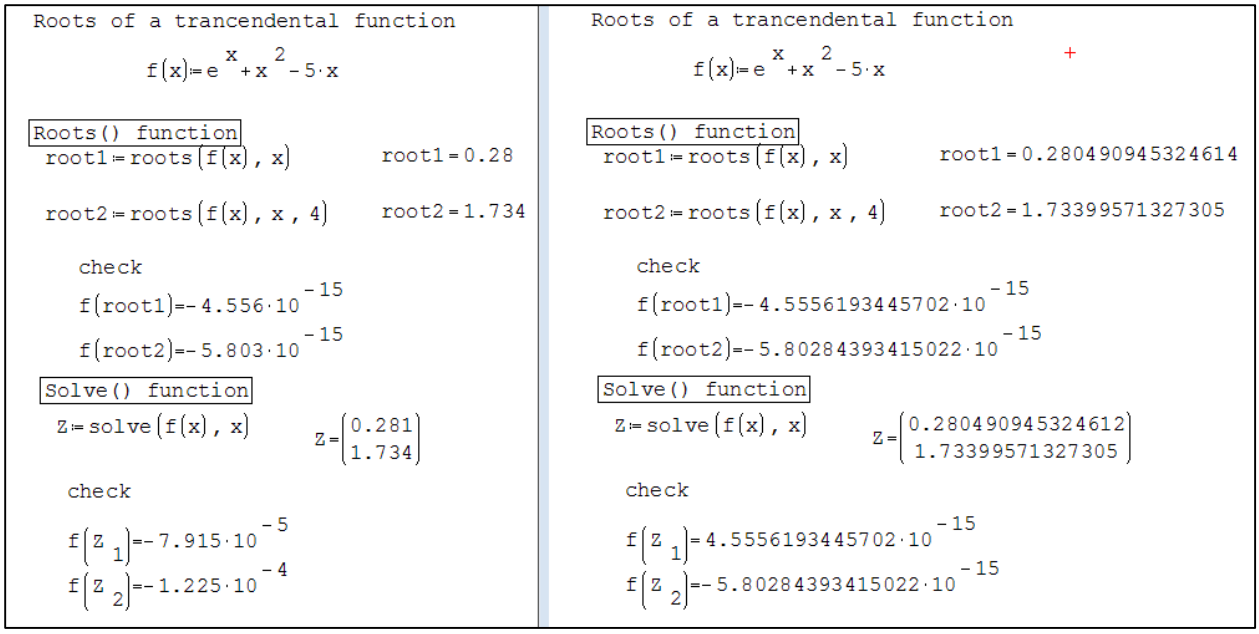
Unlike *roots*(), the *polyroots*() functions has the merit of returning all root and it can return complex roots. However, it works only with polynomial problems. It takes on argument: a vector containing the polynomial coefficients for $x^0, x^1, x^2, ...x^n$. In a figure below we solve the cubic equation $6x^3 + 7x^2 - 94x + 105$ again.





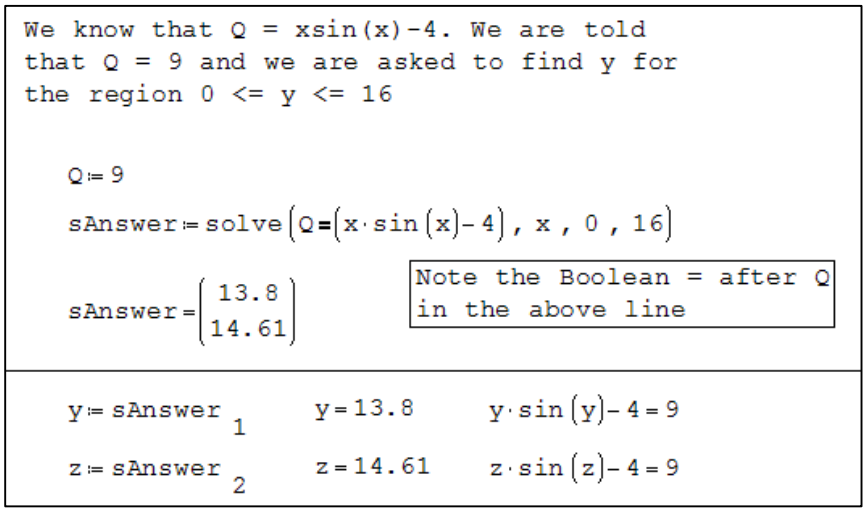
Effect of changing Tools | Options | Decimals

In the figure below we find the roots of a transcendental function. Note how the evaluation of *f(root1)* and *f(root2)* yield much smaller values than $f(Z_1)$ and $f(Z_2)$ making it appear that *roots()* is a more precise function than *solve()*. The 'check' line for roots() is at the limit of SMath's precision (10^{-15}). However, this worksheet was made with Tools | Options | Decimals set to 2. If it is increased to 15, then the precision of the two functions become the same and the 'check' values for *solve()* are identical with those for *roots().* This is one of the few times when the statement "*The decimal setting affects only the display not the accuracy of the calculation*" is false!



Solving (not root finding)

To enable us to compare the three tools *roots*(), *polyroots*(), and *solve*(), we have found the zeros of functions. The next example shows *solver*() used to find the value a variable (here *x*) needs to have to give an expression a certain value. Our expression is *xsin(x) – 4* but we have made no attempt to write this as a function of *x*. We need to know the value of x than makes the expression evaluate to *Q* where *Q* =8. A more technical example follows.



The Manning equation is used in fluid mechanics to solve problems relating to water flow through circular channels; the author thanks Gilberto Urroz for this example. The Manning equation is shown to the left below with supplementary equations to the right.

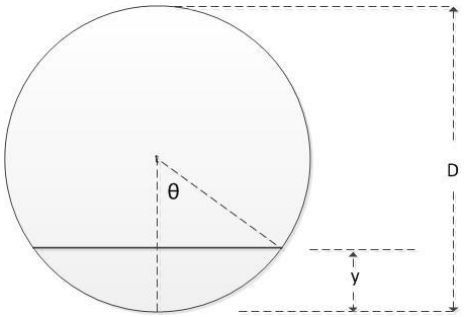
$$Q = \frac{Cu}{n}\frac{A(y)^{5/3}}{P(y)^{2/3}}\sqrt{So}$$

$$\theta(y) = \cos^{-1}\left(1 - 2\cdot\frac{y}{D}\right) \qquad P(y) = \theta(y)\cdot D$$

$$A(y) = \frac{D^2}{4}\cdot\left(\theta(y) - \sin(\theta(y))\cdot\cos(\theta(y))\right)$$

We will work in US customary units here. The variables are

Q	discharge rate (cf/s)
n	Manning coefficient (0.012 for cement)
A	area of the water in the channel (ft²)
D	diameter of channel (ft)
P	wetted perimeter (ft)
So	slope of channel
Cu	a constant (1.486 in customary units; 1 in SI units)



Examining the figure above we develop the three supplementary equations based on the auxiliary variable θ, the half-centre angle.

Open circular channel flow. The Manning Equation

Find what depth (y) results in a flow (Q) of 10 ft³/s through a channel having a diameter (D) of 3 feet. The channel is made of cement (n = 0.012) and has a slope of 0.0005

$Cu := 1.486 \quad D := 3 \quad n := 0.012 \quad So := 0.0005 \quad Q := 10$

$$\theta(y) := \operatorname{acos}\left(1 - 2\cdot\frac{y}{D}\right)$$

$$A(y) := \frac{D}{4}\cdot\left(\theta(y) - \sin(\theta(y))\cdot\cos(\theta(y))\right)$$

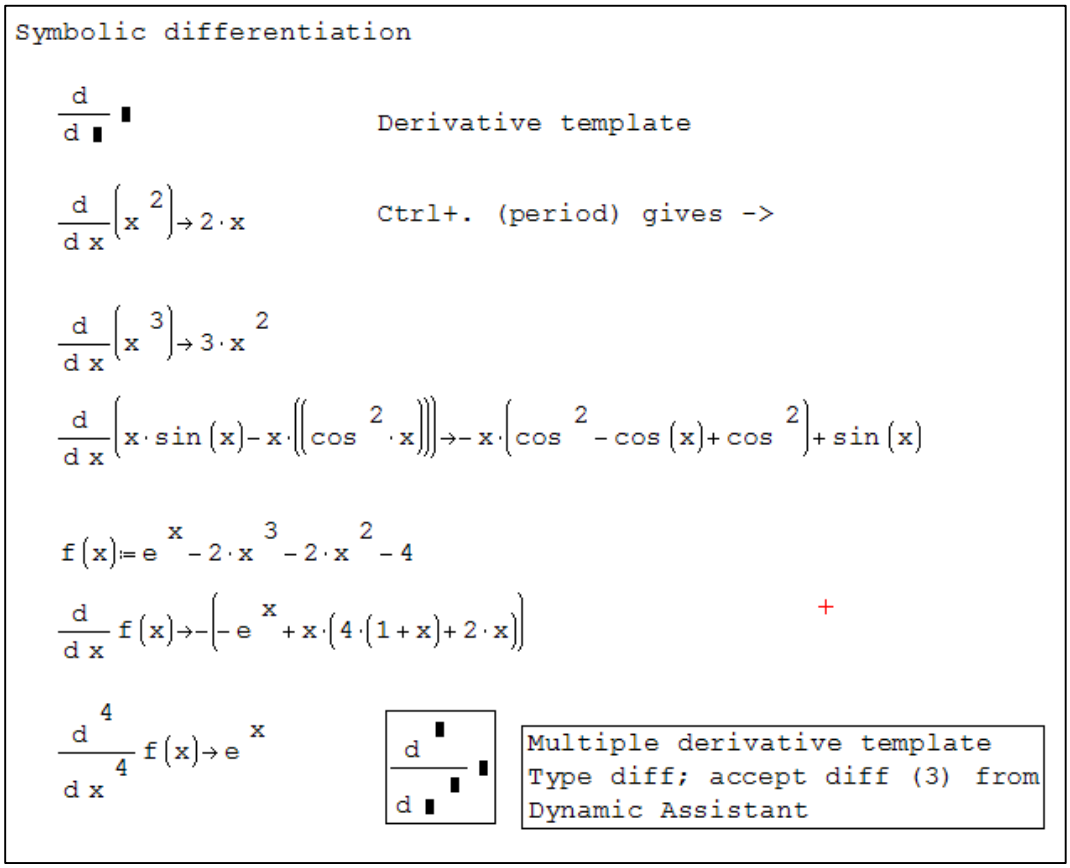
$$P(y) := \theta(y)\cdot D$$

$$\operatorname{solve}\left(Q = \frac{Cu}{n}\cdot\frac{A(y)^{\frac{5}{3}}}{P(y)^{\frac{2}{3}}}\cdot\sqrt{So},\ y,\ 0,\ 3\right) = \begin{pmatrix}1.71\\1.71\end{pmatrix}$$

Type "sol" and accept Solve4. After Q there is a Boolean equals symbol; at the end there is a simple = symbol

Symbolic Differentiation

The figure below shows some examples of symbolic differentiation.



Programming

It will be assumed that the reader has some experience of programming. When you began programming you are taught the major structural parts of a program are: sequential, decision (or branching) and looping (or repetition).

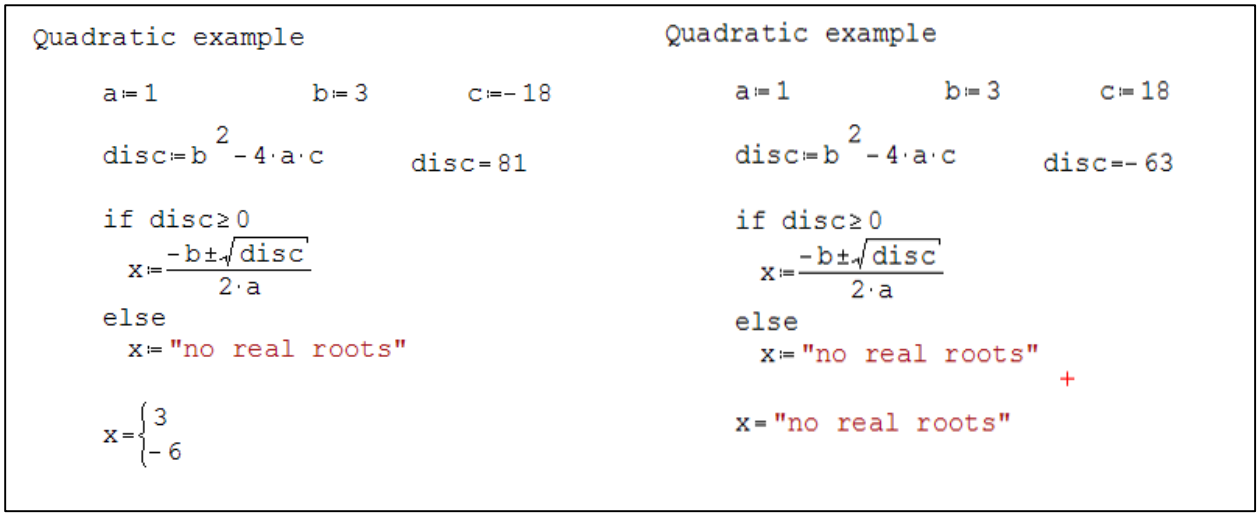
Sequential: In the Manning worksheet above we begin with four statements which are executed left to right; the program then executes the four statements that appear reading from top to bottom. That is all there is to know about the sequential structure. Unlike other languages, SMath needs no punctuation between statements on the same line – indeed there is no *line* in the programming sense. It is best to think of math regions that appear to be in line as being one below the other. SMath works top down.

Decision: A decision structure provides the program with alternative paths. The decision structure in SMath is the IF…ELSE structure. This offers two paths; the choice being dependent on the truth-value of a criterion (or a condition). For our example we will again use the well-know quadratic formula.

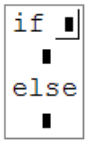
$ax^2 + bx + c = 0$ $\qquad x = \dfrac{-b \pm \sqrt{b^2 - 4ac}}{2a}$

Let $disc = b^2 - 4ac$ $\qquad x = \dfrac{-b \pm \sqrt{disc}}{2a}$

Clearly, we can have no real roots when $b^2 - 4ac$ is negative. We may define the *discriminant* as shown above and then we can say the quadratic has no real roots when disc is negative (i.e. when disc is less than zero). Here is a simple SMath example.



To insert an IF structure, position the red x where you need the structure on the page and in the Programming palette click on If to generate a bare structure as shown to the left below.



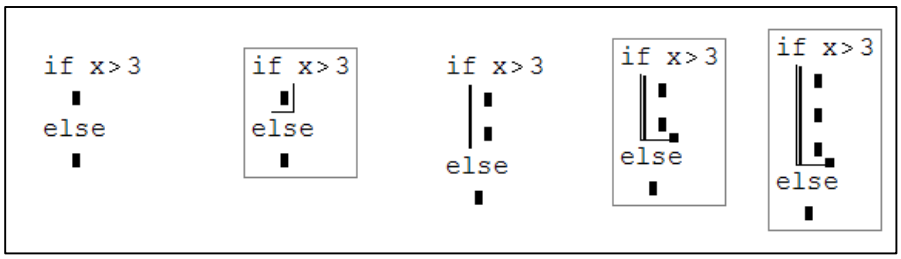
The placeholder next to *if* is where the condition is entered. The condition is also called a *logical structure*; some examples are shown above. All placeholders must be filled but there may be occasions when you just want *IF THIS TEN THAT without ELSE ANOTHER THING*. Just type a zero or a text string in the last placeholder!

SMath does not have a Select Case structure found in most computer languages.

*IMPORTANT: Do not type if; you **must** select if from the Programming palette. Alternatively, use Insert | Functions | Programming, or type if and then accept the offering from the Dynamic Assistant. This also applies to for, while, and line.*

*IMPORTANT: The various symbols (=, >, ≤, ≥, etc.) in a logical structure **must** be entered from the* Boolean palette. Do not attempt to use the keyboard for any of them. Alternatively, use Insert | Operator.

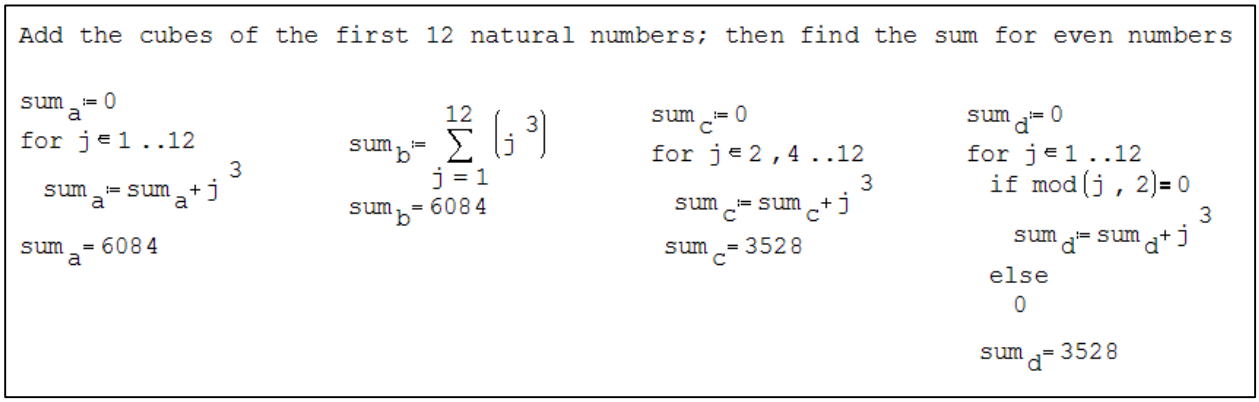
Very often we need more than one statement to be executed after the If test or the Else condition. We *can get this using the line command from the Programming palette. The sequence is shown below:*



1) Start with the basic template
2) Click on the inner placeholder
3) Click on *line* in the Programming palette
4) Click to the left of the line to reveal a pull handle in the lower right area
5) Drag this down as far as needed – preplan since there is no way of decreasing the number of placeholders within the line.

Looping: There are two looping structures in SMath: FOR and WHILE. Generally FOR is used when the number of iterations needed is known on entering the loop – *add the cubes of the first 12 natural numbers*. The WHILE loop is normally used then the actual number of iteration is not known but the looping is to occur until a certain variable reaches a preset value – *keep adding the cubes of the natural numbers while the sum does not exceed 1000*.

We find the sum of the cubes of the integers 1 to 12 in the worksheet below. We also see that a FOR loop is not really needed as we can use the summation function Σ. But the FOR loop comes into its own when we want to sum just the even numbers.



To get for j ϵ 1..12 follow these steps:

1) Click on the for item in the Programming palette to get for ■ ϵ ■
2) Type j in the first place holder, and range (or just a few letters of *range*) in the second then accept *range (2)* from the Dynamic Assistant giving get for j ϵ ■ ..■ .
3) Type the numbers into the last two place holders.

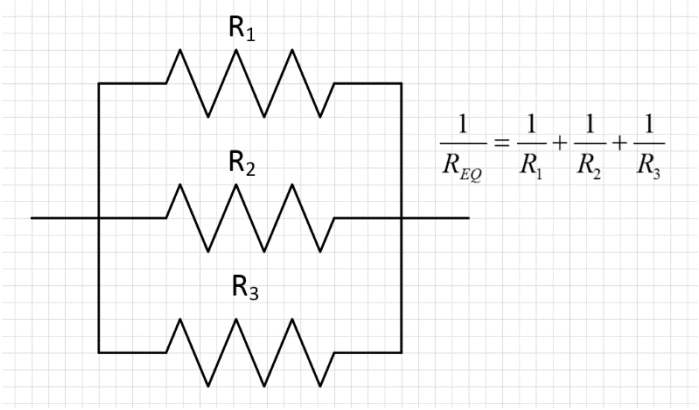
The FOR statement in the third piece of code is made the same way except that we use range (3). The first two numbers generated from 2, 4..12 are 2 and 4. The difference between 4 and2 is 2, so the following numbers increment by 2 giving 6, and 8. The sequence finishes at 8 since this is equal to the third argument. The range 1, 5, 18 would give 1, 5, 9, 13, 17; the next in the sequence would be 21 but this would exceed the third argument, so the sequence stops at 17.

The final piece of code shows an alternative method of summing the cubes of the even numbers. We have an IF structure *nested* within a FOR structure. This is shown as demonstration of nesting not as a recommended way of getting the answer to the problem at hand. Note the zero under the ELSE – we want to do nothing when the value of *j* is not even.

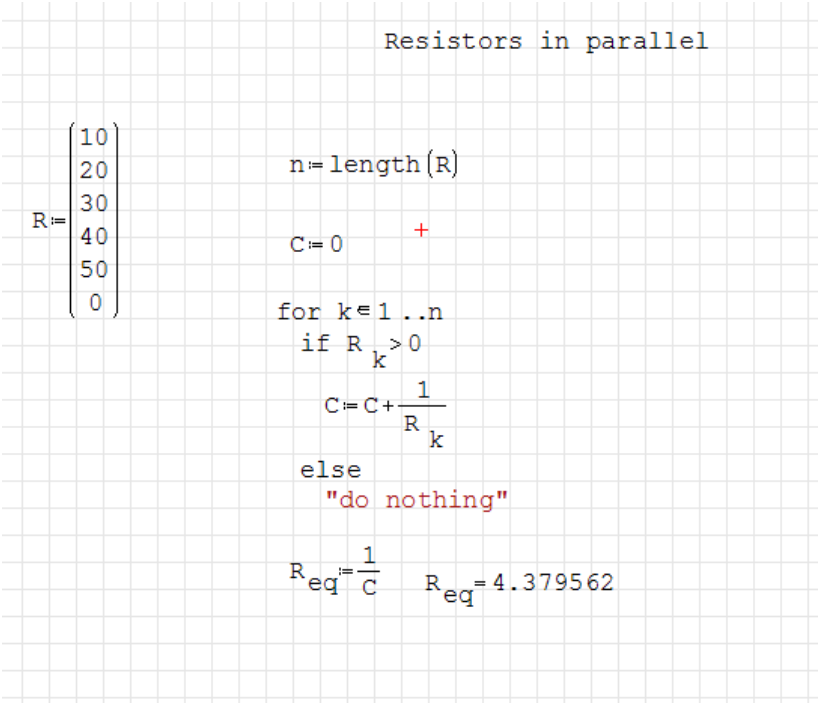
Now we look at the WHILE structure. The code in the first example in the worksheet shown below seems to indicate a bug. But no, it is exactly what we should expect! As we enter the loop fir the first time *Sum* is 0, when we exit it is 1 and j = 1. In succession *Sum* becomes 9, 36. 100 … 441, 784. By this time *j* is 7. What happens next? The WHILE structure performs its test and find Sum is still less than 1000 so it executes once more making Sum equal to 1296 with j having value 8. The next iteration fails the test (Sum is now larger than 1000) so the loop ends. But we have exceeded the program specifications. We can fix this as shown in the second piece of code.

```
Sum the cubes of the natural numbers while sum is less than 1000
   sum_e := 0                          sum_f := 0
   j := 0                              k := 0
   while sum_e ≤ 1000                  while sum_f ≤ 1000
     | j := j + 1                        | k := k + 1
     | sum_e := sum_e + j^3              | sum_f := sum_f + k^3
   sum_e = 1296                        if sum_f > 1000
   j = 8                                 sum_f := sum_f - k^3
                                       else
                                         0
                                       sum_f = 784
```

We next look at an example in electrical engineering. The diagram shows three resistors in parallel and a formula for computing the equivalent resistance if the network were to be replace by one resistor.



We will expand the problem to allow us to have up to six resistors in parallel. To this end we define a vector R with 6 elements. We will handle cases with fewer resistors by using values of zero for 'missing' resistors. We have to develop a strategy for avoiding 1/R when R = 0 since this is a mathematical no-no. We use an IF structure for this; note the use of text in the ELSE part when we wish nothing to happen.



Linear Regression Example

Suppose we have collected a number of *x, y* data pairs; this could be a chemistry experiment where *x*-values are concentration and the *y*-values the absorbance of the solutions at a particular wavelength. Our aim is to fit the data to an equation in the form y = *mx + c*.

The equations for a least-squares fit can be written in several forms, we shall use those below taken from http://mathworld.wolfram.com/LeastSquaresFitting.html

$$ss_{xx} = \left(\sum_{i=1}^{n} x_i^2\right) - n \cdot \bar{x}^2 \quad \text{where } \bar{x} = \frac{1}{n}\sum_{i=1}^{n} x_i$$

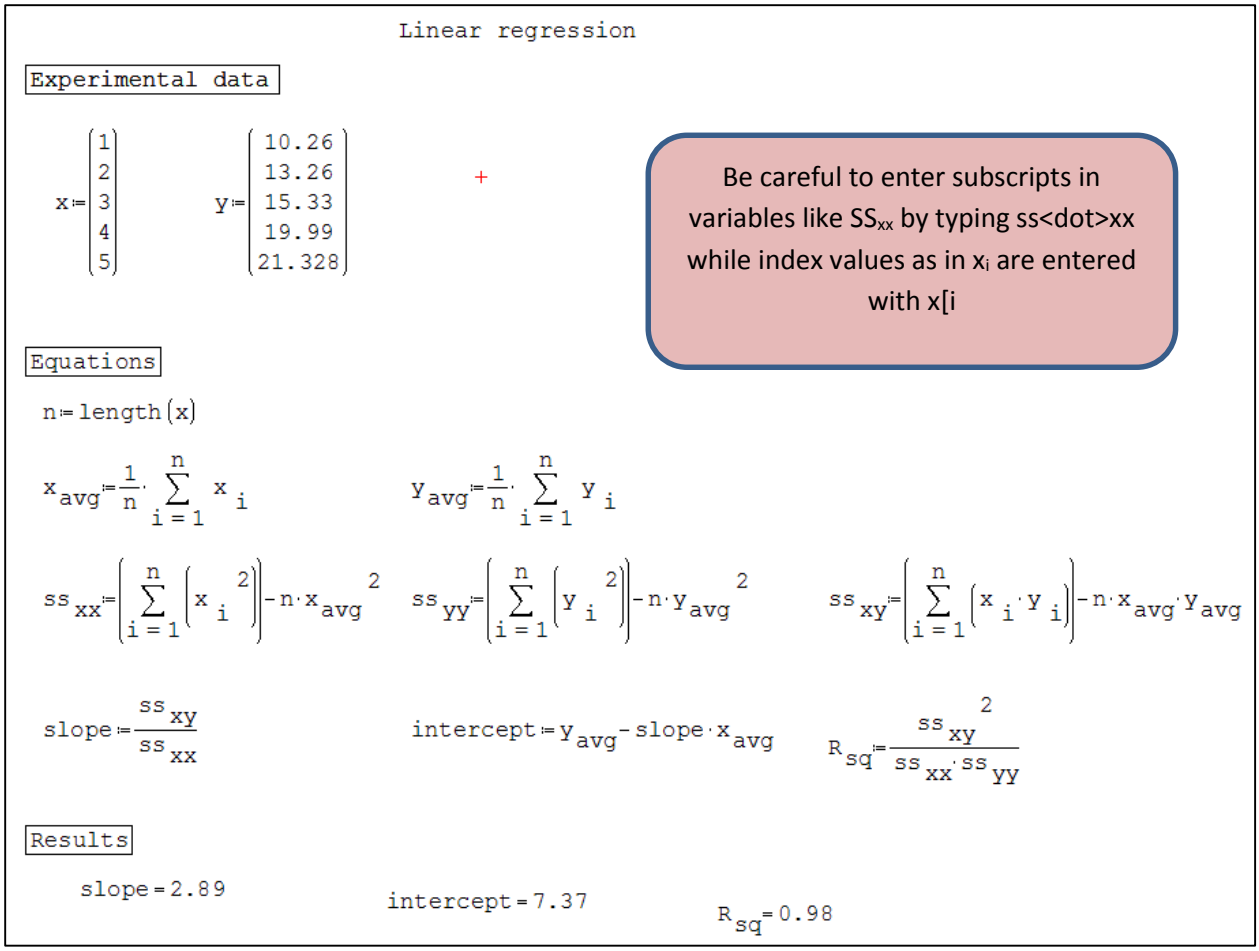
$$ss_{yy} = \left(\sum_{i=1}^{n} y_i^2\right) - n \cdot \bar{y}^2 \quad \text{where } \bar{y} = \frac{1}{n}\sum_{i=1}^{n} y_i$$

$$ss_{xy} = \left(\sum_{i=1}^{n} x_i y_i\right) - n \cdot \bar{x} \cdot \bar{y}$$

$$\text{slope } m = \frac{ss_{xy}}{ss_{xx}} \quad \text{intercept c} = \bar{y} - m\bar{x} \quad R^2 = \frac{ss_{xy}^2}{ss_{xx} \cdot ss_{yy}}$$

This worksheet uses the summation function in several expressions. This function may be entered from the Insert | Function menu by locating *sum*() or from the Functions palette where it appears as an icon similar to the Greek sigma Σ. The worksheet is reusable – just enter new x and y vectors.

At the end of the next topic (Drawing graphs) we show how to graph the results of these calculations. The file used for the above figure is called LinearRegress.sm on the author's webpage http://people.stfx.ca/bliengme/SMath.



Multi-linear Regression

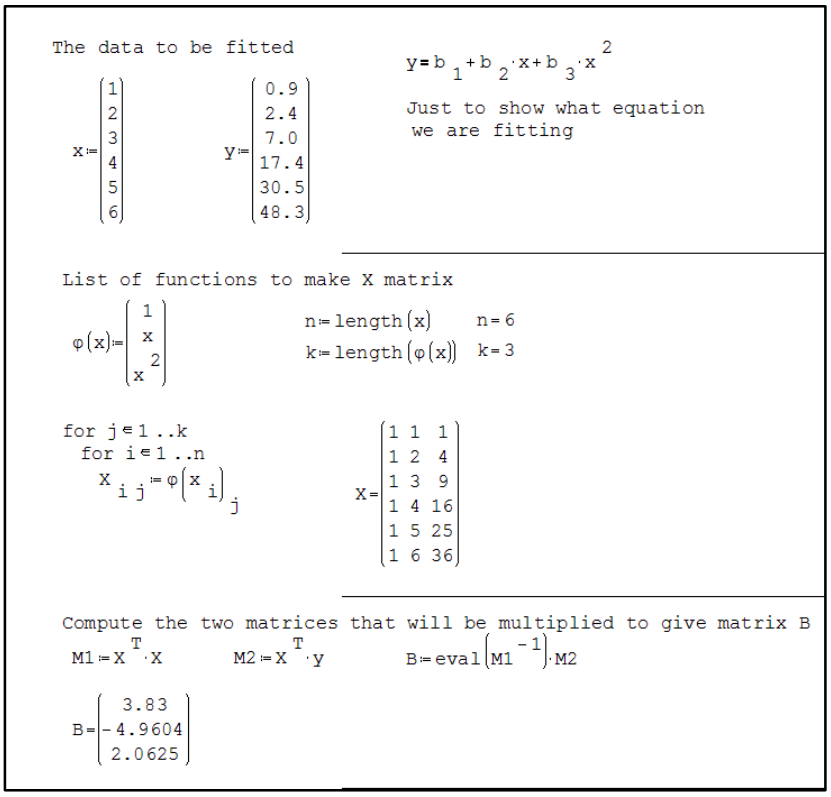
You have n pairs of (x,y) data; your task is to fit this data to the equation:

$$y = b_1 \cdot f_1(x) + b_2 \cdot f_2(x) + b_3 \cdot f_3(x) + \ldots b_k \cdot f_k(x)$$

We do this by finding the values of the coefficients b_1, b_2, …, b_k that give the minimum sum of the squares of the residuals between the given and the computed y-values (i.e. the least-squares fit).

Assuming your worksheet has had the x- and y-values entered, the next step is to construct the X matrix which, for *n* pairs of (*x, y*) values and *k* coefficients will of size *n* by *k*. The elements of the matrix will be as shown in the table below. If the first term in the equation for y is the intercept then $f_1(x)$ will be unity (the number 1) so this will be the value in each element of the first column.

$f_1(x_1)$	$f_2(x_1)$	$f_3(x_1)$	…	$f_k(x_1)$
$f_1(x_2)$	$f_2(x_2)$	$f_3(x_2)$	…	$f_k(x_2)$
…	…	…	…	…
$f_1(x_n)$	$f_2(x_n)$	$f_3(x_n)$	…	$f_k(x_n)$



← Data to be fitted

← Generate the X matrix

← Compute X^TX and X^Ty

← Compute B as product of X^TX and X^Ty

In the figure above we see an example of fitting 6 data points to a quadratic equation (3 coefficients).

This documents is not the place to offer a proof but it can be shown that one can generate a matrix (here called B) of the coefficients using the formula

$$B = (X^TX)^{-1} \cdot X^Ty.$$

In the top part of figure above we have the (x,y) pairs. In the center part we compute the X matrix as just described. For this simple function we could have computed X manually but why bark when you own a dog! In the lower part we first compute X^TX and then X^Ty. Finally we get B from $B = (X^TX)^{-1} \cdot X^Ty$.

On the author's webpage http://people.stfx.ca/bliengme/SMath the files LinearFit.sm, QuadFit.sm, and NonLinearFit.sm demonstrate this topic fitting data to $y = b_1 + b_2 \cdot x$, $y = b_1 + b_2 \cdot x + b_3 \cdot x^2$, and a $\log_{10}(P) = b_1 + b_2 \cdot (1/T) + b_3 \cdot \log_{10}(T) + b_4 \cdot T$, respectively. The same site also has the Excel file RegressionWithMatix.xlsx for those who wish to compare the Excel and the SMath approach.

Appendix A: SMath Studio Functions

Function name	Matix & Vector	Complex number	Trigonometry	Hyperbolic	Programming	Strings	Files
abs							
acos			X				
acosh				X			
acot			X				
acoth				X			
acsc			X				
ainterp							
alg	X						
arg		X					
asec			X				
asin			X				
asinh				X			
atan			X				
atanh				X			
augment	X						
cinterp							
col	X						
cols	X						
concat							X
cos			X				
cosh				X			
cot			X				
coth				X			
CreateMesh							
csc			X				
csch				X			
csort	X						
det	X						
dfile							

Function name	Matix & Vector	Complex number	Trigonometry	Hyperbolic	Programming	Strings	Files
diag	X						
diff (2)							
diff (3)							
el (2)	X						
el (3)	X						
error						X	
eval							
exp							
expand							
findstr						X	
for (3)					X		
for (4)					X		
Gamma							
identity	X						
if					X		
Im		X					
importData (1)						X	X
importData (9)						X	X
int							
invert	X						
IsDefined							
IsString						X	
Jacob							
length	X						
line					X		
linterp							
ln							
log							
log10							

Function name	Matix & Vector	Complex number	Trigonometry	Hyperbolic	Programming	Strings	Files
mat	X						
matrix	X						
max	X						
min	X						
minor	X						
mod							
norm1	X						
norme	X						
normi	X						
nthroot							
num2str						X	
numden							
perc							
pol2xy		X					
polyroots							
product							
random							
range (2)							
range (3)							
rank	X						
Re		X					
reverse	X						
rfile							X
rk fixed							
roots (2)							
roots (3)							
round							
row	X						
rows	X						

Function name	Matix & Vector	Complex number	Trigonometry	Hyperbolic	Programming	Strings	Files
rsort	X						
sec			X				
sech				X			
sign							
sin			X				
sinh				X			
solve (2)							
solve (4)							
sort	X						
sqrt							
stack	X						
str2num						X	
strlen							
strrep							
submatrix	X						
substr (2)							
substr (3)							
sum							
sys							
tan			X				
tanh				X			
tr	X						
transpose	X						
trunc							
vminor	X						
wfile							X
while					X		
xy2pol		X					