How to Calculate Efficiently

Jean-Charles Bagneris

v2018.08.1

Abstract

This document should help you make sure that you master the bare minimum in using a calculator or a spreadsheet to do various (financial) calculations efficiently.

Keywords: calculation, calculator, spreadsheet, rounding, notation, infix, postfix

Contents

Learning objectives

By the end of this module, students should be able to:

- Perform calculations on common formulas and expressions in finance
- Write readable formulas and expressions in pure text for medias like internet forums, email etc.
- Check the accuracy of their calculations by various methods
- Round properly
- Chain calculations on an electronic calculator without having to write down any intermediary results
- Use a spreadsheet efficiently, presenting tables in a clear manner, separating data from calculations, taking advantage of the numerous facilities available

Welcome!

So you finally decided to have a look at this document, despite your initial reaction, probably something in the lines of "What! Are you kidding me? I know how to calculate, I've been to school too!". Congratulations for setting your pride away for a few minutes, I think you won't regret it.

What follows is a set of useful tips and common sense rules. It is *not* a calculus course. Some of these tips and rules I learned the hard way, so I thought they might help you avoid stupid mistakes, which means save you time, headaches, and a few hair.

Before we start, let's make a quick test. Can you do the calculations below without any errors on your first try? You might also want to rewrite the numbers and formulas in a better way (with thousands separators, for example). Answers are on the next page. You can use a calculator, but not a spreadsheet (yet).

Test calculations

1. 2314 + 4312 - 3142 / 2 + 1233 / 3 * 4
2. 0.05 / (1 - 1.05^-3) * 230000
   (provide answer with 2 decimal places)
3. 240000 * (1 - 1/3)
4. 0.037 + 1.1344 * (12.8% - 3.7%)
   (provide answer in % with 2 decimal places)
5. Beta / (1 + D/E * (1-t))
   with Beta = 1.8256, D = 230, E = 394, t = 1/3
   (provide answer with 4 decimal places)

> **Answers to the test calculations**
>
> 1. 2,314 + 4,312 - 3,142 / 2 + 1,233 / 3 * 4 = 6,699
> 2. 0.05 / (1 - 1.05^-3) * 230,000 = 84,457.97
> 3. 240,000 * (1 - 1/3) = 160,000
> 4. 3.7% + 1.1344 * (12.8% - 3.7%) = 14.02%
> 5. 1.8256 / (1 + 230/394 * (1 - 1/3)) = 1.3142

If your answers were (even slightly) different, double check your calculations. Did you make a typo in inputting the numbers (quite easy with the first one)? Or maybe you had a rounding error? Or you did not chain the long calculations properly?

In this document we will try to help you address and prevent these common errors, and also help you organize a bit when you have numerical problems to solve.

1 Common rules

1.1 Notation and writing formulas

You probably could read and interpret without any effort the test calculations I proposed in the introduction to this document. But if you look carefully, they were actually not written in the purest academic way. Compare for example 2×3 with 2 x 3 and 2 * 3.

The most correct way to typeset this calculation is obviously the first one (2×3), but it is not necessary the most common one these days. The reason is that we now use computers (or tablets, smartphones, …) a lot, and that typesetting mathematics correctly on those system has been very painful for a very long time (though excellent systems exist nowadays). In addition, typesetting properly is quite often not your main objective: you might instead want to input the formula in your calculator, in a spreadsheet or even in a computer program[1] because you are interested in its result, not in beautifully printing it. You might also want to ask a question on a forum about a formula, and the forum does not allow to typeset mathematics in any special way.

Whatever the reason, the fact is that most of the time, we use a mix of common notation and computer syntax to express calculations, and this document will be no exception. To make things clear, I detail these notations below[2].

Note that here we simply follow roughly a set of commonly accepted rules. If you are interested in a formal version, see markup languages such as AsciiMath[3].

Spacing

Always use one space to separate operands (the numbers) and operations (the signs), for the sake of clarity. When using parentheses, there should be a space before an opening parenthesis or bracket, but not after, and conversely, there should be a space after a closing parenthesis or bracket, but not before.

---

[1]Programming is fun, you should give it a try one day if you never did!

[2]The computer style notation of mathematic expressions is far from perfect. It is especially not very helpful for understanding the expression. For an interesting discussion of this problem, see http://glench.com/LegibleMathematics/

[3]http://asciimath.org/

You might prefer, though, not to put any space after the closing parenthesis if the next operation is an exponentiation (^).

Example 1

- Ugly `2+3 -( 4 *5) ^ 2 + 6`
- Better `2 + 3 - (4 * 5) ^ 2 + 6`
- Best `2 + 3 - (4 * 5)^2 + 6`

Another really important point for clarity is the way to express really long numbers, either very big or very small, such as 2500000000 or 0.000000045. The use of scientific notation, like 5.34E09 for 5,340,000,000 is not really common in finance. Most people prefer to use thousands or decimal separators. The problem is then of course to make an agreement about which separator should be used. I think that it does not really matter, provided we stay consistent (don't change the separator you use in the middle of your demonstration!) and that we of course don't use the same separator for thousands and the decimal point[4].

Example 2

On May 18, 2012, Facebook went public in an IPO which valued the company over $104 billion, that is $104,000,000,000[a] in the first hours of quotation. On that same day, the JPY exchange rate was 0.012 638 729 JPY for one USD[b].

---

[a]Source: https://en.wikipedia.org/w/index.php?title=Initial_public_offering_of_Facebook&oldid=771860366
[b]Source: http://www.xe.com/currencytables/?from=USD&date=2012-05-18

Grouping, brackets and parentheses

It is often useful, and sometimes compulsory (see "The precedence order of operations" below) to group parts of long calculations, using parentheses () or square brackets []. Usually, parentheses are enough, but some people prefer to alternate parentheses and square brackets for the sake of clarity.

Example 3

Compare these two expressions:
`2 + 3 * 4 = 14`
`2 + (3 * 4) = 14`
Here the parentheses are not necessary, strictly speaking, but they add clarity.

The rule here is simply to perform first the calculations in the most inner groupings, and to carry on with the upper levels, until all groups have been calculated. This is what your electronic calculator would do automatically.

Note that electronic calculators and spreadsheets don't accept the square brackets [ ] as input, so it is probably better to avoid using those.

Finally, try to avoid using too many parentheses levels, as this might make the whole expression unreadable, and this is really error prone. You will seldom need more than 2-3 levels of parentheses, and you should really avoid using these when they are not necessary.

---

[4]I once received a quotations file where all numbers such as 234,567.45 were written 234.456.45!

Addition and subtraction

Nothing special here, we use the plus + and minus - signs for these operations.

Example 4

2 + 3 - 4 = 1

Multiplication, division, fractions and remainder

About multiplication and division, despite the usual notation for the former is the $\times$ sign, it is usually not written as x or X because to *perform* a multiplication on a spreadsheet and in most computer languages, we use the * (asterisk or star) sign. So we would write 2 * 3 rather than 2 x 3.

Regarding division, we simply use a / (slash) to express it, as well as fractions. For fractions, despite an obvious lack of clarity, we have no choice as it would be a nightmare to align properly numerator and denominator in pure text[5]. Moreover, this is again the way to express it in a spreadsheet.

Example 5

To write the fraction "100 over the sum of 1 and 0.05", we do:

$$100/(1 + 0.05) = 95.238095$$

as we certainly don't want to spend ten minutes to get this (note that I had to use a fixed-width font to make it align properly):

```
    100
--------- = 95.238 095
(1 + 0.05)
```

A seldom used operator is the remainder: it is used to express the remainder (or modulo) of the integer division between two numbers. For example, the remainder of 5 / 3 is 2, the remainder of 50 / 7 is 1. It is commonly expressed with the word "mod" or with a percent sign.

Example 6

5 mod 3 = 2   because   5 = (1 * 3) + 2
50 % 7 = 1   because   50 = (7 * 7) + 1
233 % 5 = 3   because   233 = (46 * 5) + 3

Note that the use of the percent sign is not ambiguous here, as if you want to write "five percent", you would write 5% with the percent sign attached to the five, and no number following it.

---

[5]Actually it might even be impossible, as most fonts nowadays are proportional: the value of a space is not always the same and depends on the context, which would ruin your careful alignment.

Exponentiation

Exponentiation (or "put x to the power of y") is often used in finance, as it is the basis of compound interest calculation. The formal notation is to put the exponent as a superscript, as in 2^3, but again, in pure text, this is impossible. The most common notation, including on many electronic calculators and in spreadsheets, is the caret or circumflex. We would thus write 2^3 to express "2 to the power of 3".

You might sometimes see another notation, because it is used in some computer languages (namely in Python[6], which is widely used in quantitative finance): instead of the caret, we would use ** (two consecutive asterisks or stars). The previous example would then be written 2**3.

> **Example 7**
>
> 10^6 = 10**6 = 1,000,000

1.2   More advanced notation

Below I detail more advanced notation, for things like usual mathematical functions (logarithm, trigonometric functions) or expressions involving Greek letters (sums, products and others).

Feel free to skip this part, especially on a first read. You probably don't need to learn those for a basic finance course anyway. It's here for the sake of completeness only.

Functions, logarithm, trigonometry

Function names are written in lowercase later, with the usual abbreviations, and followed by an opening parenthesis (without any space before it), the list of the function arguments separated by a comma and a space, and finally a closing parenthesis. For the logarithm, ln is the e based logarithm, for other bases, use log and postfix the base to it: log10 for example.

> **Example 8**
>
> ln(exp(1)) = 1
> 2 + log10(10,000) = 6
> cos(x)^2 + sin(x)^2 = 1
> \beta = cov(Ri, Rm) / var(Rm)

Sum, product, Greek letters

Like for the fractions, it is rather cumbersome and difficult to align properly in pure text the subscripts and superscripts for sum and product operations like the one below:

$$P_0 = \sum_{t=1}^{n} \frac{CF_t}{(1+r)^t}$$

---

[6] https://www.python.org/

For sums like this one, and products as well, one trick is to avoid completely the Greek letter by using ellipsis (...) instead. It is usually possible, and has the advantage of being less scary and more readable for people who are not confident with mathematics notation. Thus, the expression above could be written:

P0 = (CF1 / (1 + r)^1) + (CF2 / (1 + r)^2) + ... + (CFn / (1 + r)^n)

Some people prefer to separate an index or subscript with a _, thus they would write P_0 instead of P0 and CF_1 instead of CF1. It is really a matter of preference, chose whatever version you feel is more readable.

How do we input Greek letters, like the famous β in finance? I think the best way is probably to use the Latex[7] or AsciiMath[8] typesetting system convention: use the Greek letter full name, prefixed with a backslash and capitalized for uppercase letters. Thus we would write \delta for δ and \Delta for Δ.

Example 9

So the famous CAPM formula (The Capital Asset Pricing Model, see "Fundamentals of risk and return in Finance"[a]) could be written in pure text:
E(ri) = rf + \beta * (E(rm) - rf)

---

[a] https://files/bagneris.net/

1.3   The precedence order of operations

This is a very important topic, and you should master it unless you want to experience endless nightmares trying to understand or reproduce long calculations. Fortunately it is rather simple, you just have to know about it.

It happened to me many times in class that a student, waving her calculator at me, said "I can't find the same result as you". The reason is always a bad understanding of the order in which operations are performed, or misplaced parentheses – which is the same actually.

Let's start with a simple example.

Example 10

I would like you try this one *without* the help of you calculator:
2 + 3 * 6
The result of this calculation is 20, not 30. If you were wrong, or unsure, carry on, it is not that complicated.

When a calculator, or a spreadsheet, reads (we usually say "parses") such a calculation, it does it following precise rules. You should know and use the same rules, otherwise of course you will not be able to communicate properly with these devices, and your results will often be wrong.

Actually, the reason for all this trouble is the way we write the expressions, with the operator in the middle of the numbers. This is called infix notation because the operator is *in* (the middle of) the operands. As

---

[7] http://www.latex-project.org/
[8] http://asciimath.org/

it happens, infix notation is not the only possible one, and, as it is ambiguous without the help of prece-
dence rules and it needs parentheses to express some calculations, it is not the best one. This might
sound surprising as for most of us it is straightforward because we think "2 plus 3" and write 2 + 3. But
what we really mean is "add 2 and 3", thus we could as well write + 2 3. Or "take the numbers 2 and 3
and add them", yielding 2 3 +. The first notation is called prefix notation, the second one postfix notation.
Prefix and postfix notations are *not* ambiguous, thus never need any parentheses, whatever the complex-
ity of the expression[9]. Unfortunately, the vast majority of people in the world were (and still are) taught
infix notation early at school, thus would regard prefix or postfix as unreadable (or overly complicated at
best). Note though that the postfix notation is the one used in "RPN calculators". But OK, back to the infix
notation now.

So the calculator, when parsing the infix expression, will do the calculation in this order:

- first, calculate what is in the (inner most) parentheses
- then, perform exponentiation
- then multiplications and divisions
- and finally additions and subtractions.

When two operations are at the same level, the machine simply parses from left to right, as we read
in English. In the USA, they use the acronym PEMDAS (Parentheses, Exponentiation, Multiplication,
Division, Addition, Subtraction) to remember the "order of operations".

Example, continued

We now understand why
2 + 3 * 6 = 20
The parser will "see" the * and do the 3 * 6 multiplication first, yielding 18. Then it will add 2, to get
20 as a final result.

So far, so good, but how would you write the expression "add 2 to 3 and multiply the result by 6", yielding
30 instead? Parentheses to the rescue, of course. The parentheses are not only used for clarity, but also
when we want to force the order in which some operations should be performed. So, we now know that
to get 30, we should write (2 + 3) * 6.

Example 11

What do you think the expression below would yield?
(((2 + (3 * 6) - (5 + 3)) / 4) - 2))
Did you notice that parentheses don't balance? (There is an additional closing parenthesis). And
even now that you know it, isn't it a pain to check? Compare with this version of the same calcula-
tion:
(2 + (3 * 6) - (5 + 3)) / 4 - 2
Here the parentheses around 3 * 6 are not needed and were added for clarity. I think this version
makes the calculation straightforward:
- 2 + (3 * 6) is 2 + 18 = 20
- 20 - (5 + 3) = 20 - 8 = 12
- 12 / 4 = 3
- 3 - 2 = 1

---

[9] More fun with infix, prefix and postfix notations here: http://interactivepython.org/runestone/static/pythonds/BasicDS/
InfixPrefixandPostfixExpressions.html

A final note: I said it already, but you should try to avoid parentheses when adding more of those will make the expression more difficult to read instead of improving its clarity.

Two or three levels of parentheses should be a maximum. I you need more, break your calculation in (logical) pieces. It will make it easier to debug (check) in case you have a problem. Note that when you chain the calculation of the pieces together, you should never input again (that is, write) the intermediary results. More on this in "Chaining and complex calculations" below.

1.4   Magnitude

When I was in college, we had a physics professor who always required us to calculate "by head" the magnitude of any expression before using the calculator. Of course, she was very good at this, often telling us "this should be around 2.3 10^-4" before we could even finish to enter the (long) formula in our calculators.

Many years after I still remember this, and I thank her for it. She was actually right to insist on it, and it is not only a game: having an idea of the order of magnitude of a calculation, or better, being able to roughly tell what the result should "look like" before performing the actual calculation can (and will) save you stupid errors.

We all sometimes make typo, inputting the wrong number (798 instead of 987), too many zeros (was it 100000 or 1000000?), parentheses at the wrong place etc. Being able to notice that a given result is probably wrong, does not look good, or the like, is a good safety belt.

Fortunately, it is far easier in finance than in physics (for example). Most calculations in finance yield prices or returns, and we usually have a good idea of what these prices or returns "look like" without even doing any preliminary calculations. For example, we know that a bond price is never "too far" from its face value. We also know that the earnings of a company are a fraction of its turnover, that you cannot repay more than the initial amount of a loan, etc. If in addition you train a bit to be able to estimate the result of (reasonably) complex calculations, you will never again let a stupid error ruin a correct reasoning.

Finally, an additional benefit of pre-estimating by head the result of a calculation is that it forces you to read the formula again: this can help spot mistakes such as a typo in a number, or even a logical or reasoning problem.

Example 12

Try to find the magnitude of this expression without using any calculator or spreadsheet:
2,000,000 * 0.05 / (1 - 1.05^-4)
There is no absolute way to answer this question. It all depends on the tricks you know to calculate quickly, and on the knowledge you have of the underlying formula. I will use two different methods.
First, for small values of x (say, less than 10%), and reasonable values of n (less than 6 or 7) we can approximate $(1 + x)^n$ with $1 + nx$. Note that this will underestimate the true answer. So 1.05^-4 is a bit greater than 0.80, thus the denominator is slightly smaller than 0.20. Note that 2,000,000 / 0.20 is 2E06 / 2E-1, which is 1E07. Finally, we have 1E07 * 0.05 which is 1E07 x 5E-2, thus 5E5. The final result should be greater than, but reasonably close to 500,000.
The second method assumes that you recognized the expression above as the calculation of the annuity on a 2,000,000 loan, repaid by 4 constant annuities at a rate of 5%. If you are new to fi-

nance, don't worry, this formula will soon look familiar to you, we use it all the time. Knowing that, we can deduce that the annuity would be 500,000 if the interest rate was zero : 4 payments of 500,000 would indeed yield 2,000,000. Be because we have to pay an interest at 5% per year in addition to the capital repayment, the annuity is expected to be greater than 500,000, which covers the capital only. Note that this is the same conclusion as the previous paragraph calculations. Now, try to calculate the real result for yourself. The answer is 564,023.67.

1.5   Rounding and precision

Rounding should never be a problem if you follow a very simple but **absolute** rule: you should only round the **final result only**. NEVER EVER round (or worse, truncate) any intermediary result. Which means that when you have a long calculation to perform, you should get rid of the terrible habit of splitting it into pieces, noting the result of each piece on a draft paper, and then (hopefully) putting it all together, by inputting again the various pieces (that of course you mixed a bit) in the calculator. This method is a sure way to get errors. Yes, errors, the plural is intended because there will likely be more than one.

To avoid this, you should chain the calculations properly on your calculator, or on the spreadsheet. This is not too difficult and is explained below, in the "Chaining and complex calculations" section for the calculators, and in the "Cells References" one for the spreadsheet.

So, let's assume that you chained a long calculation properly. You must now chose the precision to which you have to round the final result. Either you have a precise request about this in the question ("round to 2 decimal places"), or you don't have any. If you don't have any, just follow these simple rules:

- prices and monetary values should be displayed according to the common precision of the currency: 2 decimal places for euros or dollars, for an example, no decimal places for JPY (Japanese Yen). But see "big numbers" below for really big amounts (millions and above), and remember that in the currency exchanges context, the precision is usually greater.
- percentages should be displayed with 2 decimal places: 3.45%
- big numbers and amounts (millions and above) should usually use 5 significant numbers, no more, when they are estimated, or are forecasts. But you should use the exact numbers in bookkeeping (balance sheet, income statement, etc.) Thus, write 1,456,300 instead of 1,456,298.34 if this is next year sales forecast.
- other numbers should follow the general rule of 5 significant digits. For an example, use 4 decimal places for a beta, as this will make a total of 5 significant digits (one before the decimal separator, four after), like in 1.4563

Finally, about the rounding itself. If you are curious about this, you might find different methods by searching a bit on the internet[10], but in finance there is actually only one: round to the nearest unit, rounding up when the last digit is 5 or greater than 5. Note that this is exactly what a spreadsheet would do when rounding a number implicitly for presentation purposes. This method is known as "round half up".

Example 13

2.345 rounded to 2 decimal places is 2.35
2.344987 rounded to 2 decimal places is 2.34, to 3 decimal places is 2.345, to 4 decimal places is 2.3450

---

[10]See https://en.wikipedia.org/w/index.php?title=IEEE_floating_point&oldid=781687541 for details

1,243,870.43 rounded with 5 significant digits is 1,243,900

2 Calculations with an electronic calculator

2.1 Which calculator?

Now is time for me to tell you the truth: yes, you will need an electronic calculator to follow a finance course. We deal with numbers in finance. Many numbers, all the time. And we go a bit beyond adding, subtracting or multiplying these numbers. So, the first question you might ask is "what kind of calculator will I need?". Well, anything beyond what is called a "basic calculator" will do, which means that, in addition to the four basic operations (addition, subtraction, multiplication, division), you will need exponentiation (x^y) as we use it all the time in discounting. Some memory registers will help as well, but don't worry, those are available in all reasonably advanced models. That means that so-called "accounting and tax calculators" (like the one in Fig.1), despite their names, are *not* suitable for financial calculations in most cases.
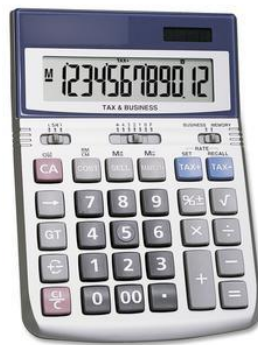

Figure 1: A basic calculator

So a suitable calculator is what is usually called "scientific calculator" by most retailers, online or brick-and-mortar ones. Note that you don't need anything fancy: graphics are not needed, one line of display is enough, and it does not need to be programmable. As of June, 2018, you probably can find one for less than $15 in the nearest supermarket. But if you went to school, chances are you have one somewhere already, just find it (difficult), find or download the manual (simple), buy new batteries (easy) and you are ready.

About the manual, just go to the calculator's manufacturer website, find the "support" or "download" section, and get the manual in your favorite language. Alternatively, just enter your calculator brand and model followed by the world `manual` in any search engine.

To end with this introduction about calculators, allow me a few words about smartphones. Although they are indeed very convenient, versatile, and always at the reach of your hands, **smartphones are not calculators**. First, I seriously doubt you will be allowed to use a smartphone in an exam, at least in the near future. Second, the default calculators applications on these phones are generally rather limited – they lack memory registers, for example. They roughly are similar to what I called a basic calculator above. Thus, make yourself a favor, and learn how to use a real calculator. Practice in order to be confident that even under the stress of an exam, you can get the results you need with certainty.

Figure 2: A suitable calculator

2.2   A quick review of less intuitive operations

One of the reason most people never opened their calculator manual is that using it is fairly intuitive for common calculations. You just input the expression as it is written and press the [=] key, and there is your result. That's good, because that means that you can even borrow someone else's calculator, a different brand and model, and manage to use it anyway. Actually we even know now that the reason for that is that expressions are commonly written using the infix notation, and (most) calculators expect you to enter infix expressions, just pressing [=] (or sometimes [EXE]) in the end to tell it to perform the evaluation and display the result.

There are some operations that nonetheless are less commonly used and might be less intuitive, even a bit different depending on the brand and model of the calculator. You should learn how to perform some of these operations, because they are time-savers, and will prove useful with many financial calculations.

One such not so intuitive fact is the difference between the - sign in the expressions -3 and 3 - 2. The first - is actually part of the negative number, so there is not really any operation in the first expression. The second one is indeed an operator, and means "subtract the number on the right from the number on the left". Most people are not really aware of the difference, but most calculators are, and some are a bit touchy about that. If you try to input 2^-3, for example, using the [-] key and expecting to calculate "two to the power of minus three", some calculators may complain. They don't understand as there is an operator on the left of the minus sign, and they expect a number. The correct way is to use the "change sign" key on the calculator to express the negative numbers. This key is usually labeled with a minus sign between parentheses [(-)] or sometimes [CHS] for "change sign", or even [+/-]. Try to find the key on the three keyboards on Fig.3.

To make things easier, some calculators use it as a prefix, thus to input "minus 4" you would use the change sign key *followed* by the [4] key, but some expect a postfix notation, which means on the latter you would have to use the [4] key *first*, then the "change sign" one. Sigh. So, make sure you check your calculator manual about this.

> **Example 14**
>
> Once you found the correct way to use negative numbers on your calculator, practice with the expressions below and make sure you get the correct results.
> 2^(-3) = 1 / 2^3 = 0.125
> 3 * 10^(-2) = 0.03
> 1.056^(-5) = 0.761 518
> 230,000 * (1 + 0.045)^(-10) = 148,103.37

Another incredibly useful feature of calculators most people never use is the availability of memory registers. Even better, it is possible to perform arbitrary calculations directly on the register content, that is, to use the register content in a calculation, and put the result in that same register. And it happens that to be able to perform complex calculations, you should know how to use your calculator memory registers, and to use those beyond simply storing and retrieving values.

So the idea is that the calculator has memory registers (at least one, but often more, 16 is a usual number). You can of course store a number or the result of a calculation in a given register, usually by pressing [STO] followed by the register "name" (probably a number from 0 to 9 or a letter from A to F). You can then recall the content of the register at any time in an expression. The usual way is to press [RCL] followed by the register name. Again, try to find the [STO] and [RCL] keys on the keyboards displayed on Fig.3.

Note that the ability to store and recall to and from registers already saves you the use of pen and paper: no need to write any intermediary result, just store it somewhere and recall it when you need it. But wait, there's more.

Figure 3: Electronic calculators keyboards

In addition to being able to store to and recall from registers, you may also perform calculations directly with the register content. That means that you can do something like "calculate this and put the result in register A". And then, "calculate that and *add* the result to register A". Finally recall the content of register A, and voilà, you have the sum of this and that! This seems simple but actually proves incredibly useful for many financial calculations that involve sums of various terms, like the NPV of an investment, the price of bonds, etc. Of course you can subtract also, and, with some scientific calculators, perform more complex calculations. But "add to register X" alone is really something you should learn to master. You won't regret it, I promise.

The "add to register content" trick is usually done by pressing the keys `[STO]`, then `[+]`, then the register name.

Now is a good time to check your calculator's manual and learn how to perform these operations:

- Store a number in register X
- Recall the content of register X
- Add the displayed number to register X
- Perform an arbitrary operation involving the current displayed number and the content of register X, storing the result in register X

Note that the last one (arbitrary operation) is not necessarily available on all calculators, but it does not matter. Adding to the register content is usually all we need in finance.

Example 15

Use the "add to register" trick to calculate the result of this expression:

$$\frac{2}{3} + \frac{4}{5} + \frac{3}{31} + \frac{67}{211}$$

Calculate each fraction in turn, then add the result to the same register content (for the first one, don't add, store it). Then recall the register content. You should find 1.880 976 … as a result. Note that you can do the calculation directly easily, just input:

2 / 3 + 4 / 5 + 3 / 31 + 67 / 211 =

But the idea here is to practice using the memory registers.

2.3   Chaining and complex calculations

I have been teaching finance for more than twenty years now. One thing I noticed that did not change over the years is that the vast majority of students (and people) don't know how to use an electronic calculator properly. In addition, they never opened (I don't even use "read") the manual, and lost it a few days after buying the calculator.

What do I mean by "use an electronic calculator properly"? Well, let's check with an example.

Example 16

Can you perform the calculation below with your calculator alone? No pen, no paper, not writing anything. Only the calculator. Provide the result with 6 decimal places.

$$-200 + \frac{23}{1 + 0.054} + \frac{12}{(1 + 0.054)^2} + \frac{45}{(1 + 0.054)^3} + \frac{37}{(1 + 0.054)^4}$$

You should find -98.964 147 …. If you were wrong, or it took you more than a few seconds, or you cheated and used pen and paper, read below.

Basically, you have two ways to do that kind of calculations:

- either you are confident enough to write it in the infix notation properly, then just enter everything in the calculator in one shot, pressing the `[=]` key in the end,
- or you are not, then split the calculation in logical pieces, calculating each piece in turn and finally "assembling" the results.

> **Example, continued**
>
> The infix notation for our example above is written below. If you enter this carefully in your calculator, without adding or removing anything, and press the [=] key in the end, you will get the correct result:
>
> -200 + 23 / (1 + 0.054) + 12 / (1 + 0.054)^2 + 45 /(1 + 0.054)^3 + 37 / (1 + 0.054)^4

Let us focus now on the "other way", that is, split the calculation in logical pieces. That means that we have to store the logical pieces results somewhere before assembling those in the end. That's where most people simply write it on a piece of paper – truncating or rounding it for convenience – and then inputting it again in the last step, "assembling" everything.

But we want to avoid the paper system, as it is long, stupid (why input again a number which was in the calculator in the first place?) and error prone (rounding or truncating intermediary result on the way, reading, writing or inputting typos). That should remind you of the memory registers we studied in the previous section.

The golden rule is that you should **never ever** input (type in) a number which is the result of a previous calculation. Instead, we use the "last answer" key (usually labeled [ANS]), or memory registers. And the trick is that we do not "assemble" in the end, but in the course of the calculation: any time we get a new piece, we immediately "assemble" it with the "under construction" final result, by using the "perform calculation on the register content" trick. After we processed all the pieces, the final result is complete and ready.

> **Example, continued**
>
> So with the example above, we decide to split the calculation logically at each + sign, that is, we will calculate each of the fractions in turn, and immediately add the result to the sum of the previous fractions so far.
>
> Let us see the different steps to get the final result. The exact way to implement those depend on your calculator model, but if you did the previous examples – and read the related part of the manual – you should be able to perform all the steps. Try until you find the correct result.
>
> 1. Input 200 in the calculator, change the sign to get -200, store it in a memory register (remember which one!)
> 2. Calculate the first fraction: simply do 23 / (1 + 0.054)
> 3. *Add* the result of this calculation to the memory register content you used at step 1
> 4. Repeat steps 2 and 3 with the remaining fractions
> 5. Recall the memory register content
>
> Check your result against the one I give above. Retry until you get it correct. Practice.

A last advice about long and chained calculations: always double check the result. And to avoid the "muscle memory" problem (your fingers make the same typo twice in a row), the second time, make the calculation backwards: start with the last piece. In the example above, you would start with the last fraction, and finish with the -200.

2.4   RPN calculators

Look, Ma, no parentheses!

You can probably skip this short section, except of course if you have a RPN calculator, or if you are considering using one.

RPN stands for reverse Polish notation, which is another name for the postfix notation. So a RPN calculator expects you use the postfix notation to enter expressions[11] and thus does not need any parentheses.

Explaining how to convert an expression from infix to postfix notation is far beyond the scope of this document. If you are interested, you can find many resources online. I like this one: *Infix, Prefix and Postfix expressions*[12] especially, as it also introduces the related notion of stack, and explains a bit the way things work.

> **Example 17**
>
> As an example, here is the calculation of the previous section example, rewritten in postfix notation:
> -200 23 1 0.054 + / + 12 1 0.054 + 2 ^ / + 45 1 0.054 + 3 ^ / + 37 1 0.054 + 4 ^ / +

Finally, I cannot write about RPN calculators without a word about the famous HP-12C, which has long been a *de facto* standard in finance, and is still produced (in a slightly modernized version) more than 30 years after its introduction in 1981. Sometimes dubbed as "the calculator that would not die", it is probably the most iconic RPN calculator. Here is a recent good (and short) read about it: *HP 12c, Thirty Five Years and Still Going Strong*[13]. Note also that its manual (that you can download easily[14]) is a very good introduction to time value of money and basic financial calculations.

3   Using a spreadsheet properly

Many people have the same problem with spreadsheets as with electronic calculators: because their use is rather intuitive, they never really learn how to use those properly. The result is usually huge, convoluted tables, with formulas and hard coded values everywhere, that nobody – even their creator – can debug, check or even update a few months after they were created. And debugged and checked they should be, as you would be surprised about the mistakes one can find in those "carefully" crafted gigantic tables. I will share a few pieces of advice regarding sheets organization in order to get tools one can use effectively, but also update if or when needed.

Of course you might also use a spreadsheet application to do quick calculations and not even save the result, like some kind of draft paper and calculator combination. Surprisingly, it often happened to me to see some students use a calculator to get some result and then enter this result in a spreadsheet cell! I wonder how they did not realize that *each* cell of a spreadsheet is a calculator, probably easier to use and more powerful than the one which sits besides their laptop.

---

[11] Most modern RPN calculators also propose the infix notation as an option. The infix notation is also known as "algebraic mode".

[12] http://interactivepython.org/runestone/static/pythonds/BasicDS/InfixPrefixandPostfixExpressions.html

[13] https://theictscoop.com/hp-12c-thirty-five-years-and-still-going-strong-c7d7183c2ad3

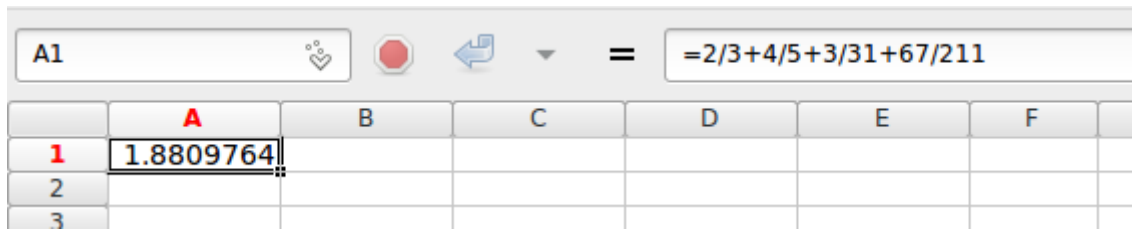[14] http://h10032.www1.hp.com/ctg/Manual/c00363319.pdf

Figure 4: Using a spreadsheet cell as a calculator

Example 18

Fig. 4 shows the calculation of example 15 above done in a spreadsheet. Here we simply use one cell as a calculator. Notice the result in the cell, and the expression in the data entry area. The infix notation is used, and the only difference with a calculator here is that the whole expression *starts* with =, where you would *end* it with the [=] key on a calculator.

3.1   Sheet Organization

The golden rule is simple: a spreadsheet should be clear, easy to check and easy to update.

Clear means that first, the objective of the sheet should be obvious, or clearly stated at the top (not in a hidden part). In addition, the data source should be identified (Where does it come from? When was the last time it changed?), if some input is possible, its location clear and immediate. Of course the results should be easy to identify as well. If the problem you have to solve or model is complex, then break it into (linked) pieces, and use one "tab" (sheet) per piece in your spreadsheet file.

Easy to check means that ideally, we can input a test data set which produces known results, to check that the results are indeed what we expect. This is made easier nowadays with the availability of scenarios or hypotheses: just record your test set as an hypothesis so that you can use it after any change to check you did not break anything.

Lastly, easy to update means that you should anticipate that things change. Tax rates, currency prices, interest rates, even some formulas change over time (because of a new regulation, the way to calculate a particular tax changes, for example). If formulas are spread all over the spreadsheet, with hard coded values interspersed everywhere, updating it will be a nightmare. Never "hard code" (write down) any parameter directly in a formula. Instead, put it in a cell and reference the cell in the formula. Be nice with your future self! Don't forget to comment and explain your formulas, especially the non trivial parts, again, this will save you and your team a lot of time when you come back to it.

Finally, note that, although rather flexible, a spreadsheet is not a one-size-fits-all problem solver. The table presentation with rows and columns is so deeply part of its structure that if your problem or model does not fit with it, do not fall into the "when you have a hammer everything is a nail" trap and use another tool. For example, the recent years have seen the development and rise of "notebooks" such as the Jupyter notebook[15] or the R notebook[16] in data science, and these tools are slowly making their way in finance as well.

---

[15]https://jupyter.org/
[16]http://rmarkdown.rstudio.com/r_notebooks.html

3.2 Presentation

A few words about presentation: although it might be a rather personal matter and a question of taste, it happens that if your work is not for personal use only, you should probably follow the simple rules below.

Alignment

This is the most important, and these are **absolute rules**, even for personal work:

- all numbers should be right aligned, and of course all numbers in a given column should have the same precision (decimal places),
- text should be left aligned, except for column headers, which should be aligned with the data in the column,
- centering is probably not a good idea.

Fonts and colors

Please refrain from using flashy colors and fancy fonts everywhere. A spreadsheet is mainly intended to display numbers: use fonts which are appropriate and align numbers correctly. Never use more than one font in a sheet, only use background color for special cases (i.e. most cells should not have any background at all), use the default borders as much as possible, do not use more than two colors for the text and numbers.

Figure 5: Discounted cash flow calculation

Example 19

I am usually very reluctant to use a lot of colors in my tables and I am rather an advocate of simplicity. But one thing I often do is put a light cyan background to the cells that contain the input data. Thus I can see immediately which numbers are inputs and which ones are outputs and result from calculations.

In Fig. 5 you can see a simple calculation for discounting using the compound interest method (this is the basis of the time value of money methods, see "Time Value of Money"[a] for details). Long story short, we should divide the cash flow by a factor which is one plus the discount rate, to the power of the number of years, that is, we do:

$$\frac{\text{Cash Flow}}{(1 + \text{Discount Rate})^{\text{Years}}}$$

So we have cells to input the cash flow, the tax rate and the years, and these have a light cyan background, and a cell with the formula – you can see the formula in the data entry area, which

displays the result: this one does not have any special background. It makes very clear that we
have three inputs, and one output in this table.

---

[a]https://files.bagneris.net/

Finally, the post *Design better data tables*[17] is a good read to summarize what you should remember
about presentation, and a good illustration is provided in *Data looks better naked*[18].

3.3   Cells References

As with calculators, on a spreadsheet, you should never ever input (type in) a number which is the result
of a previous calculation. In addition, you should never input the same number (data) twice. Use the cell
(absolute or relative) reference instead.

Moreover, using absolute or relative references properly, you can write formulas that copy horizontally or
vertically without any modifications, saving time when building tables with many similar rows or columns,
like tables where you have one column per year or month, for example.

So, when we need to use in a calculation a number which is in a given cell of our table, we just "desig-
nate" it, using the mouse pointer or the arrow keys. We never have to enter directly the cell address, this
is not convenient and error prone. Just "show" which cell you mean.

When you "show" the cell, the spreadsheet will calculate and return the cell address for you. By default,
this will be the *relative* address of the cell. A relative address is shown as the coordinates of the cell,
like C3 which means the cell at the intersection of column C and row 3, but internally, what the spread-
sheet remembers is the relative position of this cell to the one in which it is referred to (the one with the
formula). Thus, if you refer to C3 in C4, C3 actually means "same column, one row above". This is very
convenient, as if you copy your formula one cell to the right, thus in D4, the reference "same column,
one row above" will be automatically kept, and you will see D3 in the formula instead of C3. And D3 is
actually in the same column, one row above of D4.

Conversely, an absolute reference to a cell is the address of this cell and will never change upon copying
or moving. To show that they are absolute, the row or/and column references are prefixed with a $ sign,
like in A2 which refers to the cell at the intersection of column A and row 2. Again, you don't have to
input the $ yourself, there is a way to tell the spreadsheet program that the reference is absolute. It is
often the [F4] key on the top row of a PC keyboard, but it is different on Apple computers and tablets,
so you should find how to do it with your own spreadsheet and computer.

With a correct combination of absolute and relative references, you should be able to design formulas
that you input once and only once, and then copy and paste in your table as many times as needed. You
should never find yourself repeating something you did already[19].

Finally, most modern spreadsheets allow to *name* cells or range of cells, which can make your formulas
even more explicit and/or easy to check. Reading `=sales-cogs` is probably easier to understand than
`=B3-B4`, for example. The only drawback of this approach is that it does not always works well when go-
ing from one format to another one (like preparing you spreadsheet with Excel on a Windows computer
and reading it with Numbers on a MacOS one).

---

[17]https://medium.com/mission-log/design-better-data-tables-430a30a00d8c
[18]http://www.darkhorseanalytics.com/portfolio/2016/1/7/data-looks-better-naked-clear-off-the-table
[19]This is known as the "Don't Repeat Yourself" of DRY rule.

Figure 6: Discounted cash flows 1

Example 20

In Fig. 6, the columns B to F have a similar structure: in line 3 we have an input which is the year number, in line 4 another input which is the cash flow, and in line 5 an output which is the discounted cash flow, calculated as in the previous example.

Note that to calculate the discounted cash flow, we need the discount rate in addition to the cash flow and the year number. Here the discount rate is in the cell in B1.

In Fig. 6, we see that in the formula in B5 (which you can see in the data entry area), the reference to the cash flow B4 and the reference to the year B3 are relative, but the reference to the discount rate B1 is absolute. That means that the same formula, copied in all the cells in the C5 to F5 range, will work correctly: the relative references to the year (two cells above) and the cash flow (one cell above) will be changed accordingly, and the absolute reference to the discount rate (B1) will stay the same. You can see the content of the D5 cell in Fig. 7 and check that indeed, the year is taken in D3, the cash flow in D4 and the discount rate in B1.

Figure 7: Discounted cash flows 2

Summary

- To write any arbitrary mathematical expression in text only, use infix notation. Add parentheses for clarity only (i.e. don't use them if they are not needed or clutter the result).
- The precedence order of operations is Parentheses, Exponentiation, Multiplication and Division, Addition and Subtraction. It is fundamental to understand how a calculator or spreadsheet will perform the calculations in a long expression.

- Never ever round any intermediary result. Rounding should be done on the final result only, using the "round half up" rule.
- Use a powerful enough calculator: it should be able to do exponentiation with arbitrary exponents, and have memory registers
- When performing long calculations with a calculator, never ever write down or input again any intermediary result. Use the calculator memory registers.
- When using a spreadsheet, make sure you can easily tell the input data from the calculated results. Never ever input again a previous result, reference its cell instead.
- Make "movable" formulas, which will produce the expected results when copied or moved along columns (or rows). Never ever hard code any parameter in the formula, put it in a cell and reference the cell instead.

Exercises

1. Check if the expressions below are true or false. Don't use any calculator or anything, just check by looking at it.

    - 2 + 3 * 5 = 2 + (3 * 5)
    - 3 + 2 * 6 - 2 = 28
    - 6 * 5 - 3 = 6 * (5 - 3)
    - 5 * 7 - 2 = 33

2. Give the results of the calculations below without using a calculator or any other tool.

    - 600 / 3 + 45 / 5
    - 1 + 2 / 5
    - 6 * 7 / 3
    - 2^3 + 4 * 5 - 7

3. Check the expressions below by estimating the results. Don't use any calculator or other tool.

    - 3 * (1 + 0.02)^3 < 3.18
    - 2,000,000 / 0.03 > 6E07
    - (2.5 / 5 * 400 / 0.18)^3 > 1,000,000
    - 1.25 / (1 + 0.2 * (1 - 1/3)) > 1.25

4. Use your calculator to perform the calculations below. Don't use any draft paper or anything else than your calculator. Give the results with 2 decimal places.

    - 6.5 * (1 - 1.05^-12) / 0.05 + 100 / 1.05^12
    - 1.18 / (1 + 0.45 * (2 / 3))
    - -2,340 + 1,500 / 1.14 + 2,500 / 1.14^2 + 3,000 / 1.14^3
    - (2,000,000 * 0.18 - 90,000) * (1 - 1/3)

5. The formula below gives the value of the unlevered beta β_{UL}, given the beta β and leverage D/E of a company, and its corporate tax rate t:

    $$\beta_{UL} = \frac{\beta}{1 + \frac{D}{E} \times (1 - t)}$$

    - is the infix transcription below correct? If not, why?
      `\beta_UL = \beta / (1 + D * (1 - t) / E)`
    - calculate β_{UL} for the following data:
      $\beta = 1.4532$, $D/E = 45/76$, $t = 1/3$

Exercises answers

1. Check if the expressions below are true or false. Don't use any calculator or anything, just check by looking at it.

    - 2 + 3 * 5 = 2 + 15 = 17 = 2 + (3 * 5) True
    - 3 + 2 * 6 - 2 = 3 + 12 - 2 = 13 != 28 False
    - 6 * 5 - 3 = 30 - 3 = 27 != 6 * (5 - 3) False
    - 5 * 7 - 2 = 35 - 2 = 33 True

2. Give the results of the calculations below without using a calculator or any other tool.

    - 600 / 3 + 45 / 5 = 200 + 9 = 209
    - 1 + 2 / 5 = 1.4
    - 6 * 7 / 3 = 42 / 3 = 14
    - 2^3 + 4 * 5 - 7 = 8 + 20 - 7 = 21

3. Check the expressions below by estimating the results. Don't use any calculator or other tool.

    - 3 * (1 + 0.02)^3 < 3.18 False
      3 * (1 + 0.02)^3 = 3 * 1.02^3
      3 * 1.02^3 > 3 * 1.06
      3 * 1.02^3 > 3.18

    - 2,000,000 / 0.03 > 6E07 True
      2,000,000 / 0.03 = (2 / 3) * (1E06 / 1E-2)
      (2 / 3) * (1E06 / 1E-2) = (2 / 3) * 1E08
      (2 / 3) * 1E08 > 0.6 * 1E08
      (2 / 3) * 1E08 > 6E07

    - (2.5 / 5 * 400 / 0.18)^3 > 1,000,000 True
      (2.5 / 5 * 400 / 0.18)^3 = (0.5 * 400 / 0.18)^3 = (200 / 0.18)^3
      (200 / 0.18)^3 > (200 / 0.2)^3
      (200 / 0.2)^3 = 1,000^3 = 1,000,000

    - 1.25 / (1 + 0.2 * (1 - 1/3)) > 1.25 False
      1 + 0.2 * (1 - 1/3) > 1
      1.25 divided by something > 1 is lower than 1.25.

4. Use your calculator to perform the calculations below. Don't use any draft paper or anything else than your calculator. Give the results with 2 decimal places.

    - 6.5 * (1 - 1.05^-12) / 0.05 + 100 / 1.05^12 = 113.29
    - 1.18 / (1 + 0.45 * (2 / 3)) = 0.91
    - -2,340 + 1,500 / 1.14 + 2,500 / 1.14^2 + 3,000 / 1.14^3 = 2,924.37
    - (2,000,000 * 0.18 - 90,000) * (1 - 1/3) = 180,000

5. The formula below gives the value of the unlevered beta β_{UL}, given the beta β and leverage D/E of a company, and its corporate tax rate t:

    $$\beta_{UL} = \frac{\beta}{1 + \frac{D}{E} \times (1 - t)}$$

    - is the infix transcription below correct? If not, why?
      ```
      \beta_UL = \beta / (1 + D * (1 - t) / E)
      ```

Yes, the transcription is correct: the parentheses around the denominator ensure that we correctly calculate it before performing the division. Note that multiplying D by `(1 - t)` first and then dividing by E is the same as dividing D by E and then multiplying by `(1 - t)` as multiplication and division are commutative

- calculate β_{UL} for the following data:
  $\beta = 1.4532$, $D/E = 45/76$, $t = 1/3$
  We get `\beta_UL` = 1.4532 / (1 + 45 * (1 - 1/3) / 76) = 1.0419