

Using VBA in Excel

You may never have done any computer programming, but when you're searching for solutions to an equation, a program can be of enormous benefit. It really is (honestly) easy to get started with coding, and once you've started, there is no telling where you might finish.

I'm going to assume you have access to the Microsoft Excel program. Lots of people use Excel without knowing that it has the programming language Visual Basic built into it. I'll show you how it works.

Open a spreadsheet, and go to the Developer tab (if you don't have that showing, try Microsoft Office Button > Excel Options > Popular > Show Developer Tab). Now click on Insert, and in the ActiveX controls section, click on the Command Button (if you hover over the options, it will tell you which it is).

Now click somewhere on the spreadsheet, and the command button should appear. Drag it into place. Double click on the Command Button, and you get a page where you can enter your Visual Basic code.

Private Sub CommandButton1_Click()

Space for code

End Sub

A simple program might say:

s=0

'puts the initial value of s to 0

For a = 1 to 10

'starts with a = 1, increasing a by 1 each time we reach 'Next a', and finishing when a gets to 10

s = s + a^2

'the new value for s is the old value for s + a squared.

Next a

'move on to the next value of a, going back to the 'For' statement.

Cells(1,1) = s

'insert the final value for s into Row 1, Column 1.

The green lines starting with an apostrophe are simply comments, and explain what each line does without having any effect on the program. Can you tell what the program does?

Try typing out the program (you don't need to put in the comments) into the space provided, then click on 'Design Mode' to come out of this, and run the program by clicking on your Command Button. This program adds together the first 10 square numbers, so it should end by giving 385 in Cell A1.

You may not need much more than For-Next loops, the Cells(1, 1) notation, and the If-then statement to build some handy programs.

A nested pair of For-Next loops might look like this:

```
s = 0
For a = 1 to 10
For b = 1 to 10
s = s + a*b
Next b
Next a
Cells(1, 1) = s
```

For each of the ten values for a, b runs through 1 to 10, which makes 100 calculations of s in total. This program adds together all the numbers in the white area in the grid below. The answer 3025 will hopefully be in Cell A1.

x	1	2	3	4	5	6	7	8	9	10
1	1	2	3	4	5	6	7	8	9	10
2	2	4	6	8	10	12	14	16	18	20
3	3	6	9	12	15	18	21	24	27	30
4	4	8	12	16	20	24	28	32	36	40
5	5	10	15	20	25	30	35	40	45	50
6	6	12	18	24	30	36	42	48	54	60
7	7	14	21	28	35	42	49	56	63	70
8	8	16	24	32	40	48	56	64	72	80
9	9	18	27	36	45	54	63	72	81	90
10	10	20	30	40	50	60	70	80	90	100

Warning; remember to save the changes to any program before running it! You don't want Excel to give up, only for you to then lose your program.

Programs can help us with searching for an answer to a problem. I've chosen the Fermat-Catalan conjecture to show this, the result of merging Fermat's Last Theorem with Catalan's Theorem.

Fermat's Last Theorem says $x^n + y^n = z^n$ has no whole number solutions for x, y and z if n is greater than 2 (proved by Andrew Wiles in 1997).

Catalan's Theorem (suggested in 1844) states that $x^a - y^b = 1$ has only one solution in whole numbers,

$$3^2 - 2^3 = 1 \text{ (this was proved by Preda Mihailescu in 2002).}$$

So what might the love child of these two great theorems look like? With almost any conjecture, it's possible to make it more general, and that creates something harder to prove.

We're going to try to find solutions to the equation $a^m + b^n = c^k$. (*)

How many whole number solutions can we find to (*)? What happens if one or more of m, n and k is 1? What if m, n and k are all 2? What if two of m, n and k are 2? What if m, n and k are all 3? If we insist that $1/m + 1/n + 1/k < 1$, then we guarantee that m, n and k are not all small. Can we find a solution for (a, b, c, m, n, k) with this restriction? How could we find other solutions? Does this equation (*) have infinitely many solutions? Or finitely many? So how could we code a program to tackle the Fermat-Catalan conjecture?

```

Private Sub CommandButton1_Click()
s = Cells(1, 7)
j = 1
For a = 1 To s
For b = a + 1 To s + 1
For c = 1 To 150
For m = 2 To 10
For n = 2 To 10
For k = 2 To 10
If 1 / m + 1 / n + 1 / k < 1 Then
If a ^ m + b ^ n = c ^ k Then
Cells(j, 1) = a
Cells(j, 2) = b
Cells(j, 3) = c
Cells(j, 4) = m
Cells(j, 5) = n
Cells(j, 6) = k
j = j + 1
End If
End If
Cells(1, 8) = a
Next k
Next n
Next m
Next c
Next b
Next a
End Sub

```

A simple search for solutions might look like this. This program employs six nested For-Next loops and nested If-then statements. Clicking out of 'Design mode' and then clicking the Command Button runs the program. Putting s = 20 generates the output below.

This at least is the start of the output, which I cut short (pressing Esc gets you out of the program). You can see that the first four solutions have been found (the highlighted cells).

1	2	3	7	3	2	20	5
1	2	3	8	3	2		
1	2	3	9	3	2		
1	2	3	10	3	2		
2	4	2	4	2	5		
2	4	2	6	3	7		
2	4	2	8	4	9		
2	4	8	8	4	3		
2	4	24	9	3	2		
2	7	3	5	2	4		
2	8	2	6	2	7		
2	8	2	9	3	10		
2	8	4	9	3	5		
2	8	32	9	3	2		
2	16	2	8	2	9		
2	16	8	8	2	3		
2	17	71	7	3	2		
3	6	3	3	3	5		
3	6	45	6	4	2		
3	9	54	7	3	2		
3	11	122	5	4	2		
3	18	3	6	3	8		
3	18	9	6	3	4		
4	8	2	3	2	7		
4	16	2	4	2	9		
4	16	8	10	5	7		
4	16	128	10	5	3		

It takes Excel a while to run this. Every time you add another nested loop, the number of calculations increases exponentially (the idea of cell H1 is to keep track of how far the program has run). Sometimes you need to leave a computer running overnight to arrive at an answer. Once you've experienced the thrill of waking to remember that your machine is waiting for you with an answer, you'll never feel the same

way about maths or computing again (I promise!)

Excel can go up to 16 significant figures; larger numbers than this are going to end in

$$1^m + 2^3 = 3^2$$

$$2^5 + 7^2 = 3^4$$

$$13^2 + 7^3 = 2^9$$

$$2^7 + 17^3 = 71^2$$

$$3^5 + 11^4 = 122^2$$

$$33^8 + 1549034^2 = 15613^3$$

$$1414^3 + 2213459^2 = 65^7$$

$$9262^3 + 15312283^2 = 113^7$$

$$17^7 + 76271^3 = 21063928^2$$

$$43^8 + 96222^3 = 30042907^2$$

approximate results. These ten red solutions below to (*) are the only ones known. Our program found four of these, as far as it was allowed to run. You'll notice that in each of these ten solutions, one of m, n and k is 2. So we can add to the unsolved problem above Beal's Conjecture, which states that if we have a solution to (*), then one of m, n and k has to be 2. Beal is a rich man, and he currently offers a prize of a million US dollars to anyone resolving his conjecture one way or the other. This puzzle could be the wisest investment of time you've ever made...

If you're going to investigate conjectures, using a computer is common practice. It can save you time by finding you a quick counterexample, and stopping you from disappearing up blind alleys. But mathematicians disagree on how helpful they are; Andrew Wiles says he never uses them, and it certainly hasn't troubled his career. We need to beware too of leaping onto a computer to resolve something when a little hard thought would answer the question for us immediately. Sometimes a pencil and paper are all we need.