Gerald's Column
by Gerald Fitton

Professor W W (Bill) Mapleson wrote to me to say that he has enjoyed my "Column" in Archive since it first started (when was that?). Of course, flattery will get you anywhere; how could I resist when he asked me for help with a PipeDream/Fireworkz problem!

So this month we'll take a break from media files to provide a solution to Bill's problem.

The Problem

Bill wants to solve an equation using an iterative numerical method.

Aphrodite's Technique

I'm sure that if I asked you, well most of you, to find the length of the side of a square piece of paper which has an area of 10 square metres you would use some O Level mathematics (or GCSE mathematics if you are much younger than I am) to write down an equation such as x^2=10. Then, by inverting this equation (do you remember my "mathematicians do it backwards" series?) you would deduce that the value of x is the square root of 10. This is sometimes written as x=10^(0.5).

These days your final step would not be to use a Brunsviga mechanical calculating machine which I described in my ancient article about Aphrodite (the Greek Goddess of Love). No! You would whip out a calculator and use its square root function to discover that the square root of 10 is about 3.1623.

Why do I call it Aphrodite's Technique? Because, through it, by teaching her how to do this using a mechanical calculating machine, I came to befriend Jill, who became my wife. This technique is a numerical method of extracting square roots to which I referred in my Gerald's Column of March 1996 together with anecdotes from a Professor John Greening, a PhD, some Teachers and Susan. I shall quote the paragraph which I wrote about Susan.

Susan

"Finally, of the many letters I've had from practising 'computors' I must select one. A lady called Susan Jones has sent me a disc file in PipeDream format which gives a practical algorithm based on the (x + y)^2 = x^2 + 2*x*y + y^2 formula I quoted for extracting square roots using an electronic spreadsheet. She says that it was taught to her by her grandmother (a former Royal Aeronautical Establishment computor from Farnborough) at the tender age of eight. I must congratulate the grandmother; it seems obvious to me that, of the many facts of life, she was acutely aware of the one which would make her young grand daughter singularly eligible and at the same time enable her to indicate that she had selected her lifetime partner. I must add that Susan Jones signs herself as a Miss so, to quote her, she 'has not yet come across the Mr who is worthy to receive this secret aphrodisiac algorithm inherited from my grandmother'. I had considered putting Susan's grandmother's algorithm on the Archive monthly disc but, after careful consideration, I decided that it would be a breach of confidence!"

I have often wondered whether Miss Susan Jones is now a Mrs and whether her ability to extract square roots using her grandmother's algorithm served her well?

Bill's Problem

Bill's Problem is not about extracting square roots but something slightly more complex.

Bill has a mathematical equation of the type s=f(p) where the function, f(p) contains nine terms. If you know p, the partial pressure of oxygen in the blood then, using this equation, you can calculate the saturation, s, of the haemoglobin. Unfortunately the quantity which is measured is s (the saturation of the haemoglobin) and not the partial pressure p.

Bill wants to find p when given s. He needs to 'invert' the equation.

I wonder if they teach the solution of quadratics at school these days with its discriminator ("b squared minus 4 ac") and stuff like that? I remember my grandfather bemoaning the fact that I wasn't taught how to extract cubics and quartics! He had to learn these techniques at night school at the end of the 19th century (about 1890). There is no analytical solution possible for the general fifth power equation and certainly solving a ninth degree polynomial is not the sort of thing to tackle analytically. Solving equations which contain trigonometrical or exponential functions is generally harder still and there is no general solution. For example, can you solve an equation such as y=x*sin(x)?

So we need a numerical rather than an analytical method.

Numerical methods usually need you to use an iterative process. Bill had solved his problem using a fairly simple BASIC program and he sent me a copy of the iterative sub routine so that I could see what he was trying to do.

Bill's BASIC method in words

Rather than give you Bill's BASIC routine I shall describe his method in words. Also, so that we don't get into evaluating nine term expressions, I'll take as my example solving the equation y=x^2 when we know that y=10 and we want to find x. To put it simply, we'll use Bill's method to find the square root of 10, the length of the edge of a 10 m^2 square.

We are pretty certain that the answer will be between 0 and 10 so our first guess at the answer, x, is half way between the two at (0+10)/2 which is 5.

Next we calculate y for x=5 using the formula y=x^2. We find that for x=5, y=25.

This is too large so we have to make another guess. We don't do that by guessing 4 and trying again; we do something better than that. For our next guess we try the average of 0, the lower of our two limiting values, and the 5 we just tried. So our next guess at the value of x is (0+5)/2 which is 2.5.

We find the corresponding value of y=x^2=2.5^2=6.25. This time the answer, 6.25, is less than 10 so we know that x must be larger than 2.5. Also, we know that the value of x is less than 5 so, for our next guess we choose (2.5+5)/2=3.75.

We go on like this getting ever closer to the square root of 10 until we are near enough.

The Binary Chop

The mathematicians amongst my readers will recognise this technique as one often called "The Binary Chop". There are other much more sophisticated techniques which converge much more rapidly such as Newton-Raphson which you'll be able to find explained on the Internet in much more detail than is possible here. Let's stick with the Binary Chop.

Iteration

My article this month is not about different techniques, their advantages and disadvantages but, to quote Bill, how to execute "Iterative calculations in PipeDream or Fireworkz". Bill doesn't think I've covered iterative calculations in Archive but I do recall doing so in 1996. Actually I covered iteration using an extended spreadsheet. Spreadsheets of that type do look very messy; it is better to use a Custom Functions to do this sort of thing.

Custom Functions

I did cover custom functions in Archive articles back in 1992 so I'm not going to go back over that ground again. I believe that those articles are still available on the Archive Annual CD. However, if you can't find it then have a look at the many tutorial examples of custom functions in both PipeDream and Fireworkz format on our GoldLine CD.

If the function you want to use is not part of the PipeDream or Fireworkz standard set of functions then you can write your own function. Such functions which you write yourself are called custom functions.

For example, PipeDream and Fireworkz do include the inverse of the function y=tan(x) as x=atan(y). What you will not find in the standard set of functions is an inverse function for the quintic function y=x^5+x. If you are given x you can find y from the equation y=x^5+x but you will have trouble finding x when you are given y.

This is the nature of Bill's problem.

Using PipeDream or Fireworkz, Bill's problem can be solved by writing a suitable custom function which will use a numerical (non analytical) method to find x when given y.

What I have done is to write a simple custom function which can be used to invert many equations of the type y=f(x) provided that f(x) can be evaluated. By "can be evaluated" I mean that you can find a numerical value for y if you are given a numerical value for x.

Iteration

One big advantage of custom functions over the functions which are included in the standard set is that conditional statements and loops can be included within the function. Thes features of custom functions allow you, the writer, to use iterative procedures such as the Binary Chop as part of the solution to your numerical and string problems.

The set of control commands which can be used within a custom function includes not only IF-ELSE-ELSEIF, FOR-NEXT, REPEAT-UNTIL, WHILE-ENDWHILE but also the totally abhorrent and definitely not recommended GOTO!

The custom function which solves Bill's problem and many others of the same type contains a repeat-until loop and a conditional if-else-endif branching point within the loop.

One of the parameters specified in the evaluation of the inverse of Bill's nine term equation is the degree of accuracy required. When the two numbers which bracket the correct answer are near enough the same number then the custom function exits the repeat-until loop and returns an answer.

The if-else-endif branching point is used to decide whether to replace the current upper or lower bound with the latest guess.

The custom function, just like a BASIC program, keeps running around the loop between the repeat and until lines as many times as is necessary to home in on the result.

Nested Custom Functions

Those of you who decide to look at the PipeDream and Fireworkz custom functions which I have sent by email to Bill will see that the file [c_iterate] contains two custom functions.

The first is the function I have just described; it does the iteration. The second custom function consists of four lines the last one of which is: line 46 "...result(x^2)".

The specific problem I chose as my example is to solve inverse of y=x^2 using iteration. The "x^2" in line 46 is the right hand side of this equation. You can replace the "x^2" by any function which can be evaluated (to a number) when x is known. For example, if you wish to find the solution to y=x*sin(x) all that you need to do is to replace the "x^2" with "x*sin(x)" and my custom function will solve your equation.

The first function calls the second function just like a BASIC procedure or subroutine.

Communication

Write to me about Archive things to archive@abacusline.co.uk. Indeed, write to me if you would like me to send you a copy of this custom function in either PipeDream or Fireworkz format and I will send it to you.

You might like to try http://archive.abacusline.co.uk (with Username "archive" and Password "amusement" both without the inverted commas) if you wish to download this and other articles I have written for Archive. There is a link from that site to our AVLine website which I am using to store media files (pictures, audio and video files).

I have completed the conversion of most of my recent Gerald's Column articles to PDF format. They contain Type 1 fonts such as Homerton and Trinity (or their Windows equivalent). You will be able to download them from our Archive website and, I hope, be able to read them on whatever platform you use.