Quick Challenge: 16-09-2013 , polynomials

From brilliant.org:

> There is a unique choice of numbers a and b such that the polynomial $y^4 + 2y^3 + ay^2 + 2y + b$ can be written as $f(y)^2$, where $f(y)$ is also a polynomial. Find $2a + 9b$.

Brilliant.org (as many other sites, but this one is *ad hoc*) provides a lot of smart and quick mathematical challenges, useful to learn or revise mathematical concepts and so forth. One can use it even to learn more about the **mathematical framework** of the Hp50g, that is very huge. This is the objective.

Note: we will try to avoid CAS manipulations as much as possible, plus we will not be so rigorous.

I think that there are many solutions to the problem[1], let's try some of them to see if they work.

Ideas

Note: we suppose (we are trying after all), that the coefficients are integer numbers.

1. We known that a polynomial that is the square root of another has its degree equal to an half of the original polynomial degree. So given $f(y) = c_o + c_1 \cdot y + c_2 \cdot y^2 + c_3 \cdot y^3 + c_4 \cdot y^4$ we have that the square root of this polynomial can be written in the following way for some coefficents d_i : $\sqrt{f(y)} = d_0 + d_1 \cdot y^1 + d_2 \cdot y^2$. We can try to find these coefficients, for the given case, how? We are not rigorous here, so we can do a program that compares the value for $(\sqrt{f(y)})^2$ and $f(y)$, for a fixed set of y values, trying values of $a, b, d_{0,} d_{1,} d_2$. That needs a lot of time but should solve the problem[2].
2. The $f(y)$ polynomial has four roots, while $\sqrt{f(y)}$ has two of

1 Of course between these there is a quick and elegant solution like: if we can write the given polynomial as $f(y)^2$ and we know that $f(y)$ is a polynomial, then we have that $f(y) = d_0 + d_1 \cdot y^1 + d_2 \cdot y^2$. Thus $f(y)^2 = (d_0 + d_1 \cdot y^1 + d_2 \cdot y^2)^2$, we equating coefficients with the given polynomial and so we have the solution. But this solution is achieved without much number crunching, is “too human”. We want, instead, to let the calculator find the solution using its math library.

2 As I always said: in the worst case a programmer is a bad mathematician, finding solutions by brute force.

them. Nevertheless, with the right coefficients, both roots of $\sqrt{f(y)}$ are included in the roots of $f(y)$ [3]. Moreover if $f(y)$ is the square of a second degree polynomial (then with two roots), we have that $f(y)=g(y)^2=(y-r_1)^2\cdot(y-r_2)^2$; so we need to test only a,b until we find two couples of equal roots (or even a single root repeated four times). With this idea we will explore some commands to find roots of polynomials.

Implementations

Note: the actual code is attached to the pdf version of this document.

The idea no1 is almost unfeasible, it took too much time if we try, in a trivial way, all the values in a fixed interval for $a,b,d_{0,}d_{1,}d_2$. Then we discard this approach.

The idea no2 is, eventually, similar to idea no1, because we try values only for a,b instead of $a,b,d_{0,}d_{1,}d_2$, but this approach can take a lot of time too. For example if we find roots of $f(y)$ for all ordered couples (a,b) , where $a,b\in[-100,100]\subset\mathbb{N}$, in the worst case the calculator need to find roots of around 40'000 polynomials.

So, these two ideas are not satisfactory, we need to explore more the math library of hp50g to find more feasible solutions.

3 Roots can be in pair of complex numbers, remember that exist a statement which assert: if a polynomial with real coefficients have a complex root, there it has also another complex root that is the conjugate of the former. (Even the hp50g user guide, that with 800+ page, reports it: page 6-6,6-7.)

Appendix

Tools used

- Staroffice 8 (that is a pain but latex won't apply easily the opentype/truetype font. Word 2007+ is the best[4] actually :/ , but I can't use it now)
- Codecogs, an online equation editor.
- Offside as font. Cambria as font for formulas.
- Paint.net, to edit some images.
- MathBin to save the formulas done with codecogs.
- Hp 50g graphing calculator to test the ideas.
- Debug4x to write RPL code for the 50g.

Versions

- 16.09.2013 first draft with two ideas
- 17.09.2013-18.09.2013 small changes

4 typographically