Short & Sweet Math Challenge #21: Powers that be Threaded Mode | Linear Mode

11-02-2016, 02:13 AM (This post was last modified: 11-02-2016 10:48 PM by Valentin Albillo.) **Post: #1**

Valentin Albillo
Senior Member

Posts: 636
Joined: Feb 2015
Warning Level: 0%

Short & Sweet Math Challenge #21: Powers that be

Short & Sweet Math Challenge #21: Powers that be

Hi all,

At the end of April 2008 and after 20 releases I concluded my **"Short & Sweet Math Challenges"** series which allowed you to dust off and show off both your favorite programmable **HP calcs** and your math/HP-programming skills while introducing and discussing interesting and unusual mathematical topics.

The series was well received and now, some 8 years later, I'm releasing *"season 2"*, so to say, beginning with this brand-new **S&SMC #21**, in the tradition and style of the preceding ones. Iif this second installment gets a similarly good reception, I intend to eventually post another full 20-strong batch.

Now as for this present new challenge, first a short prelude. Many of you will surely have heard about **Phi**, the ubiquitous so-called **Golden Ratio** constant which can be promptly evaluated as *(1+Sqrt(5))/2*, namely:

```
Phi = 1.61803+
```

Phi is further characterized by being *the largest root in absolute value* of the polynomial *x^2-x-1* (i.e.: Phi^2-Phi-1 = 0), which happens to be the *lowest*-degree polynomial for which Phi is a root and thus it gets called the *"minimal polynomial"* of Phi.

As it happens, Phi has a plethora of very interesting, sometimes unique properties, one of them being that *its increasing powers are ever nearer to integer values*, for instance:

```
Phi^20   =   15126.99993+
Phi^50   =   28143753122.99999999996+
Phi^100  =   792[...]126.99999999999999999998+
Phi^200  =   627[...]126.999999999999999999999999999999999999999998+
```

Alas, while certainly interesting, this *quasi-integer* property of the powers of Phi is far from unique, there are some other constants that boast this very property, as for instance (let's call it "**Hpi**" for the time being):

```
Hpi = 1.32471+
```

which happens to be the largest root in absolute value of the polynomial *x^3-x-1* (i.e.: Hpi^3-Hpi-1 = 0), which is its *minimal polynomial*. Indeed, we do have:

```
Hpi^20   =   276.992+
Hpi^50   =   1276942.001+
Hpi^100  =   163[...]001.9999995+
Hpi^200  =   265[...]250.000000000001+
Hpi^500  =   115[...]876.99999999999999999999999999994+
```

demonstrating that Hpi's powers also get nearer and nearer to integer values, like Phi's do.

That said, now go and get a programmable **HP calc** of your choice (say an HP-10C or better, hardware/software based HP-calc emulators also welcomed) and use that calc's language of your choice (say RPN, FOCAL, RPL, BASIC, FORTH, etc.) to try and meet:

The Challenge:

Write a program to find and output the constants that have the aforementioned *quasi-integer*-powers property, subject to the requirements that your program must:

- find **ALL** such constants in the range *1 < constant < 2*, for minimal polynomials *up to degree 8* having *all* their coefficients equal to either **+1**, **0**, or **-1**, the leading coefficient in particular being always **+1**.

- output both each constant **AND** the minimal polynomial for which it is a root, *sorted by increasing numerical value*, with a final tally of how many constants were found (*hint: more than 30*) and, optionally, timing.

Of course, the faster your program runs the better, smaller program sizes being important but secondary. If your chosen HP calc runs too slowly you might consider, in order:

- using a better algorithm and/or optimizing your solution for speed,
- coding in a faster language (say FORTH or assembler, if available),
- using a faster HP calc,
- using an HP-calc emulator running on a faster platform, or
- just be patient and have a meal (or two) while it merrily runs.

That's all. Within a few days I'll post and comment my **12-line** (60-statement, 585-byte) solution for the **HP-71B**, which runs like this (no spoilers):

```
>RUN
       ... some lines of output ...

       1.57367896839    x^8-x^7-x^6+x^2-1
       1.59000537390    x^7-x^5-x^4-x^3-x^2-x-1
       1.60134733379    x^7-x^6-x^4-x^2-1
       1.61803398875    x^2-x-1

       ... more lines of output ...

       (number of constants found)
```

I'll also comment on the underlying theory and algorithm used as well as trivia and problems I found. Now for the small print:

- you can use any ROMs, libraries or binary files for your calc as long as they are readily available, preferably for free, as well as extra RAM if needed. For instance, for the HP41 and emulators you can use the Math ROM, Advantage ROM, card reader ROM, printer ROM and many others. My HP-71B solutions typically may use the Math ROM, HP-IL ROM, JPCROM, STRNGLEX binary and additional RAM modules.

- you can use any hard/soft emulator for a given HP calc model but solutions given for non-HP calcs (say written in Excel, Visual Basic, C#, for vintage SHARP or TI machines, etc) **won't** be considered to have met the Challenge and in fact may spoil the fun for all.

- ditto for using the web to search for solutions. You will ruin the fun for yourself and other people attempting the challenge so you'd better try and solve it by your own efforts first, not mercilessly scavenging the web.

Enough said, now **let's see your solutions !** (and please *do **not** use CODE sections in your posts*, they don't print well).

Best regards.
V.
.

Find All My HP-related Materials here: **Valentin Albillo's HP Collection**

PM WWW FIND EDIT QUOTE REPORT

11-02-2016, 06:47 AM **Post: #2**

Gerson W. Barbosa
Senior Member

Posts: 1,361
Joined: Dec 2013

RE: Short & Sweet Math Challenge #21: Powers that be

HP 50g, 181.5 bytes, 14 seconds. Only 6 constants, though.

Best regards,

Gerson.

```
%%HP: T(3)A(R)F(.);
\<< 1. 6.
  FOR i i X SWAP ^ [ 1. -1. -1. ] X PEVAL * [ 1. 0. -1. ] X PEVAL + EVAL DUP 'X' ZEROS DUP TYPE
    IF
    THEN OBJ\-> DROP NIP
    END
  NEXT
\>>

TEVAL

'X^3.+0.*X^2.+-1.*X-1.'
                   1.32471795724
'X^4.+-1.*X^3.+0.*X^2.-1.'
                    1.3802775691
'X^5.+-1.*X^4.+-1.*X^3.+X^2.-1.'
                   1.44326879127
'X^6.+-1.*X^5.+-1.*X^4.+X^2.-1.'
                   1.50159480354
'X^7.+-1.*X^6.+-1.*X^5.+X^2.-1.'
                   1.54521564973
'X^8.+-1.*X^7.+-1.*X^6.+X^2.-1.'
                   1.57367896839
                       s:14.0446
```

EMAIL PM FIND QUOTE REPORT

11-02-2016, 08:16 AM **Post: #3**

Massimo Gnerucci
Senior Member

Posts: 2,152
Joined: Dec 2013

RE: Short & Sweet Math Challenge #21: Powers that be

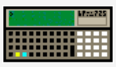
Welcome back Valentin and S&SMC!

Greetings,
Massimo

-+×÷ ↔ *left is right and right is wrong*

PM WWW FIND QUOTE REPORT

11-02-2016, 02:48 PM (This post was last modified: 11-02-2016 02:55 PM by J-F Garnier.) **Post: #4**



J-F Garnier
Senior Member

Posts: 461
Joined: Dec 2013

RE: Short & Sweet Math Challenge #21: Powers that be

Hi Valentin,

I'm very happy to read you again in this forum !

One question:
From the output examples here:

> **Valentin Albillo Wrote:** (11-02-2016 02:13 AM)
>
> ```
> 1.57367896839 x^8-x^7-x^6+x^2-1
> 1.59000537390 x^7-x^5-x^4-x^3-x^2-x-1
> 1.60134733379 x^7-x^6-x^4-x^2-1
> ```

it seems that null polynomial coefficients are allowed, not only **+1** or **-1**,
Is it correct?

J-F

EMAIL PM WWW FIND QUOTE REPORT

11-02-2016, 03:00 PM Post: #5

Bill (Smithville NJ)
Senior Member

Posts: 415
Joined: Dec 2013

RE: Short & Sweet Math Challenge #21: Powers that be

Hello Valentin,

Welcome back and thanks for another very interesting challenge.

Some of the newer forum members may not be familiar with your past challenges. Back in 2005, I assembled the first eleven of these into a single PDF file. I never did update it. You are up to 21 now. I guess I need to spend a little time getting it updated to 21.

Please see the following forum post for a link to the PDF file of the first 11:

SSMC PDF

Bill
Smithville, NJ

EMAIL PM FIND QUOTE REPORT

11-02-2016, 11:08 PM Post: #6

Valentin Albillo
Senior Member

Posts: 636
Joined: Feb 2015
Warning Level: 0%

RE: Short & Sweet Math Challenge #21: Powers that be

.
Hi, **Gerson** !

I'm truly glad to get your always valuable contributions to one of my S&SMC's, as in the good old times. A few comments:

> **Gerson W. Barbosa Wrote:** ⇒ (11-02-2016 06:47 AM)
> HP 50g, 181.5 bytes, 14 seconds. Only 6 constants, though.

Why only **6** ? Not being versed in RPL I don't fully understand your code but I also don't see any reference to the *maximum degree* for the polynomials, which should be **8**.

> **Quote:**
```
'X^3.+0.*X^2.+-1.*X-1.'              1.32471795724
'X^4.+-1.*X^3.+0.*X^2.-1.'           1.3802775691
'X^5.+-1.*X^4.+-1.*X^3.+X^2.-1.'     1.44326879127
'X^6.+-1.*X^5.+-1.*X^4.+X^2.-1.'     1.50159480354
'X^7.+-1.*X^6.+-1.*X^5.+X^2.-1.'     1.54521564973
'X^8.+-1.*X^7.+-1.*X^6.+X^2.-1.'     1.57367896839
```

Also, even within this limited range, your program is *missing* **two** constants, one between 1.50.. and 1.54.. and another between 1.54.. and 1.57.., perhaps due to the typo that **J-F** pointed out (which I duly corrected) about the coefficients being **-1**, **0**, or **+1** (the '**0**' was missing in my OP).

Thnaks for your continued interest and best regards.
V.
.

Find All My HP-related Materials here: **Valentin Albillo's HP Collection**

PM WWW FIND EDIT QUOTE REPORT

11-02-2016, 11:14 PM Post: #7

Valentin Albillo
Senior Member

Posts: 636
Joined: Feb 2015
Warning Level: 0%

RE: Short & Sweet Math Challenge #21: Powers that be

Hi, **Massimo** !:

> **Massimo Gnerucci Wrote:** ⇒ (11-02-2016 08:16 AM)
> Welcome back Valentin and S&SMC!

Thank you very much for your kind welcome, much appreciated.

Perhaps you'd consider contributing your very own solution to the present challenge ? It's just a warmer, there are many interesting ones ahead !

Best regards.
V.

Find All My HP-related Materials here: **Valentin Albillo's HP Collection**

PM WWW FIND EDIT QUOTE REPORT

11-02-2016, 11:22 PM **Post: #8**

Valentin Albillo
Senior Member

Posts: 636
Joined: Feb 2015
Warning Level: 0%

RE: Short & Sweet Math Challenge #21: Powers that be

Hi, **Jean-François** !:

Thanks a lot for your kind comment and continued interest in my posts, much appreciated.

> **J-F Garnier Wrote:** ⇒ (11-02-2016 02:48 PM)
> One question:
> From the output examples here:
>
> > **Valentin Albillo Wrote:** ⇒ (11-02-2016 02:13 AM)
> >
> > ```
> > 1.57367896839 x^8-x^7-x^6+x^2-1
> > 1.59000537390 x^7-x^5-x^4-x^3-x^2-x-1
> > 1.60134733379 x^7-x^6-x^4-x^2-1
> > ```
>
> it seems that null polynomial coefficients are allowed, not only **+1** or **-1**,
> Is it correct?

Yes, of course, my bad, thanks for pointing it out to me, I've already corrected it in my original post.

I was going to write that the absolute value of the integer coefficients had to be up to and including **1** but decided instead to just enumerate them and the '**0**' was simply left out.

Thanks and here's hoping for your own solution to this challenge, it would be an amazing way to start the *"second season"* ! ... 8-D

Best regards.
V.

Find All My HP-related Materials here: **Valentin Albillo's HP Collection**

PM WWW FIND EDIT QUOTE REPORT

11-02-2016, 11:39 PM **Post: #9**

Valentin Albillo
Senior Member

Posts: 636
Joined: Feb 2015
Warning Level: 0%

RE: Short & Sweet Math Challenge #21: Powers that be

Hi, **Bill** !

Thank for your kind welcome and appreciation, I'm glad you like my S&SMC's

> **Bill (Smithville NJ) Wrote:** ⇒ (11-02-2016 03:00 PM)
> Back in 2005, I assembled the first eleven of these into a single PDF file. I never did update it. You are up to 21 now. I guess I need to spend a little time getting it updated to 21.

It would be great if you'd eventually find the time to update your PDF file with the whole collection but you might consider waiting till the *"second season"* is completed instead of doing incremental updates. Whatever suits you.

> **Quote:**
> Please see the following forum post for a link to the PDF file of the first 11: SSMC PDF

I already downloaded your PDF file back then and found it extremely useful to me because I was *missing* some of the earliest challenges and your PDF put them back in my hands in a most convenient format so I was very obliged to you for it.

If you'd like to try your hand at providing your very own solution (or any comments) to the present challenge, it would be my pleasure to have a look at it.

Best regards.
V.

Find All My HP-related Materials here: **Valentin Albillo's HP Collection**

PM WWW FIND EDIT QUOTE REPORT

11-03-2016, 12:19 AM (This post was last modified: 11-03-2016 12:34 AM by Gerson W. Barbosa.) **Post: #10**

Gerson W. Barbosa
Senior Member

Posts: 1,361
Joined: Dec 2013

RE: Short & Sweet Math Challenge #21: Powers that be

> **Valentin Albillo Wrote:** ⇒ (11-02-2016 11:08 PM)
> .
> Hi, **Gerson** !
>
> I'm truly glad to get your always valuable contributions to one of my S&SMC's, as in the good old times. A few comments:
>
> > **Gerson W. Barbosa Wrote:** ⇒ (11-02-2016 06:47 AM)
> > HP 50g, 181.5 bytes, 14 seconds. Only 6 constants, though.
>
> Why only **6** ? Not being versed in RPL I don't fully understand your code but I also don't see any reference to the *maximum degree* for the polynomials, which should be **8**.
>
> **Quote:**
> ```
> 'X^3.+0.*X^2.+-1.*X-1.'              1.32471795724
> 'X^4.+-1.*X^3.+0.*X^2.-1.'           1.3802775691
> 'X^5.+-1.*X^4.+-1.*X^3.+X^2.-1.'     1.44326879127
> 'X^6.+-1.*X^5.+-1.*X^4.+X^2.-1.'     1.50159480354
> 'X^7.+-1.*X^6.+-1.*X^5.+X^2.-1.'     1.54521564973
> 'X^8.+-1.*X^7.+-1.*X^6.+X^2.-1.'     1.57367896839
> ```

Hello, Valentin!

Thanks for starting Season 2. I'm looking forward for the next episodes. I've always appreciated your insightful and well-thought S&SMC series, even though most of the time I was able to solve only the easier items.

Regarding this particular problem, I remember Phi belongs to a special set of numbers named after a French mathematician which produce near-integers when raised to high powers. I misspelled his name, but Google pointed me to the right reference wherein I found three polynomials that generate such numbers. The RPL program uses only the first generating polynomial.

PEVAL creates a symbolic polynomial expression from a variable name and a coefficents vector. For instance,

[1 1 1] 'X' PEVAL --> '1+(1+X)*x' FACTOR --> 'X²+X+1'

PROOT might be a better alternative to ZEROS, but I couldn't find a built-in inverse to PEVAL so PROOT could be used. Anyway, I guess this is not the kind of solution your looking for, so I won't proceed with this approach any longer.

> **Valentin Albillo Wrote:** ⇒ (11-02-2016 11:08 PM)
>
> Also, even within this limited range, your program is *missing* **two** constants, one between 1.50.. and 1.54.. and another between 1.54.. and 1.57.., perhaps due to the typo that **J-F** pointed out (which I duly corrected) about the coefficients being **-1**, **0**, or **+1** (the '**0**' was missing in my OP).

Yes, I was aware of the missing constants. By using the third generating polynomial in the aforementioned reference, a few more can be found:

```
%%HP: T(3)A(R)F(.);
\<< 3. 8.
  FOR n X n ^ X n 1. + ^ 1. - X 2. ^ 1. - / - DUP 'X' ZEROS DUP SIZE GET
  NEXT
\>>

TEVAL

'X^3.-(X^4.-1.)/(X^2.-1.)'
             1.46557123188
'X^4.-(X^5.-1.)/(X^2.-1.)'
             1.53415774491
'X^5.-(X^6.-1.)/(X^2.-1.)'
              1.5701473122
'X^6.-(X^7.-1.)/(X^2.-1.)'
              1.5900053739
'X^7.-(X^8.-1.)/(X^2.-1.)'
             1.60134733379
'X^8.-(X^9.-1.)/(X^2.-1.)'
             1.60798272793
               :s: 18.1289
```

Not in the required format, though. Also, the polynomial have yet to be simplified and checked whether the degrees are no greater than 8.

Best regards,

Gerson.

Edited to fix a typo

EMAIL PM FIND QUOTE REPORT

11-06-2016, 06:21 PM (This post was last modified: 11-06-2016 06:22 PM by J-F Garnier.) **Post: #11**

J-F Garnier
Senior Member

Posts: 461
Joined: Dec 2013

RE: Short & Sweet Math Challenge #21: Powers that be

> **Valentin Albillo Wrote:** ⇒ (11-02-2016 11:22 PM)
>
> ... hoping for your own solution to this challenge, it would be an amazing way to start the *"second season"* ! ... 8-D

Well, I tried first on Emu71, then switched to Free42 to have higher computing accuracy, but I'm afraid I wasn't able to build a proper solution with either tool.
With Emu71, I was able to find the roots, but wasn't able to identify **all** the constants with the desired property due to the limited accuracy.
And with Free42, I had the right accuracy, but had difficulties to find the roots of the polynomials.

Waiting for your solution...

J-F

EMAIL PM WWW FIND QUOTE REPORT

11-07-2016, 11:23 PM (This post was last modified: 11-07-2016 11:23 PM by Valentin Albillo.) **Post: #12**

Valentin Albillo
Senior Member

Posts: 636
Joined: Feb 2015
Warning Level: 0%

RE: Short & Sweet Math Challenge #21: Powers that be

Hi, **J-F**:

> **J-F Garnier Wrote:** (11-06-2016 06:21 PM)
>
> Well, I tried first on Emu71, then switched to Free42 to have higher computing accuracy, but I'm afraid I wasn't able to build a proper solution with either tool.
>
> With Emu71, I was able to find the roots, but wasn't able to identify **all** the constants with the desired property due to the limited accuracy.

How many did you identify ? How do you know they aren't **all** ?

For the particular limits of this challenge, i.e.: minimal polynomials up to *degree 8* (or less) and with coefficients **-1,0,+1**, I found no accuracy problems at all (though perhaps there *are* and I just didn't find them ... 8-D)

> **Quote:**
>
> And with Free42, I had the right accuracy, but had difficulties to find the roots of the polynomials.

How so ? Details ?

> **Quote:**
>
> Waiting for your solution...

I'll post it within two days, give or take a day. It does take a significant amount of time to write down the solution post and regrettably it seems there wasn't much interest at all, no one but you and Gerson made any attempt at a solution or posted comments, let alone post actual code.

Thanks for your interest.

Best regards.
V.

Find All My HP-related Materials here: **Valentin Albillo's HP Collection**

PM WWW FIND EDIT QUOTE REPORT

11-08-2016, 09:28 AM (This post was last modified: 11-08-2016 11:11 AM by J-F Garnier.) **Post: #13**

J-F Garnier
Senior Member

Posts: 461
Joined: Dec 2013

RE: Short & Sweet Math Challenge #21: Powers that be

> **Valentin Albillo Wrote:** (11-07-2016 11:23 PM)
>
> How many did you identify ? How do you know they aren't **all** ?

I didn't attempt to count them, but I know that I didn't identify all constants with Emu71, because I missed at least one: the 1.3802775691 value that Gerson reported.

When trying to check if the powers of this value get closer and closer to an integer, I got:

```
I 1.3802775691^I
51 13749532.9553
52 18978171.9238
53 26195145.0089
54 36156571.0751
55 49906104.0305
56 68884275.9545
57 95079420.9637
```

```
58 131235992.039
59 181142096.07
60 250026372.026
61 345105792.99
62 476341785.031
63 657483881.103
64 907510253.132
65 1252616046.13
66 1728957831.16
67 2386441712.27
68 3293951965.42
69 4546568011.56
70 6275525842.74
...
```

The closest values are around power 60, then the lack of accuracy makes the fractional part no more significant.
My criteria was that the value must be close to an integer with 0.01 tolerance for 3 successive power values, to have a good confidence.
I could have relax my criteria, but what if other values had an even worst behaviour?

With Free42, I had the right accuracy, with 35 digits.
I first solved the equation x^4-x^3-1=0 with the solver to have the root X with full accuracy, then the powers (frac part) are:

```
FP(X^75) = 0.98147851
FP(X^76) = 0.98907003
FP(X^77) = 0.01158426
FP(X^78) = 0.01475045
FP(X^79) = 0.99622896
FP(X^80) = 0.98529899
FP(X^81) = 0.99688326
FP(X^82) = 0.01163371
FP(X^83) = 0.00786267
FP(X^84) = 0.99316166
FP(X^85) = 0.99004492
FP(X^86) = 0.00167862
FP(X^87) = 0.00954129
FP(X^88) = 0.00270295
FP(X^89) = 0.99274787
FP(X^90) = 0.99442649
FP(X^91) = 0.00396778
FP(X^92) = 0.00667073
FP(X^93) = 0.99941859
FP(X^94) = 0.99384508
FP(X^95) = 0.99781286
```

Here my criteria is met around power 85.
But since there is no polynomial root solver on the HP-42S (and I didn't want to write one), I had to use the equation solver, and I wasn't sure that the reported root was the correct, largest one.

Regarding the participation to the challenge, maybe others met the same difficulties.
Unless there is another, better way to solve the challenge :-)

J-F

11-08-2016, 02:38 PM (This post was last modified: 11-08-2016 02:40 PM by Paul Dale.) **Post: #14**

Paul Dale
Senior Member

Posts: 1,662
Joined: Dec 2013

RE: Short & Sweet Math Challenge #21: Powers that be

I'd figured out an approach to this problem. I've not implemented it on any hardware.

It is essentially a brute force approach to the problem:

For each length of polynomial (1 .. 8):

- Iterate over all polynomials for the given length that satisfy the specified constraints. That the leading coefficient is 1, the constant term is ±1 and the remaining term coefficients are from {-1, 0, 1}.
- Over all lengths there are 6560 such polynomials. If the constant term can also be 0, the count is 9840.
- Find the roots of each polynomial.
- If there is one real root > 1 and all of the remaining (possibly complex) roots have |root| < 1, then the polynomial is of the required form.

PM FIND QUOTE REPORT

11-08-2016, 03:01 PM **Post: #15**

J-F Garnier
Senior Member

Posts: 461
Joined: Dec 2013

RE: Short & Sweet Math Challenge #21: Powers that be

> **Mike (Stgt) Wrote:** (11-08-2016 01:31 PM)
>
> ... and - regarding the polynom as Sum from i=0 to <=8 of a(i)*x^i - the coefficient a(0) always -1.

I'm suspecting that a(0) is always -1 (for solutions of the given challenge), but I don't understand your argument.
J-F

EMAIL PM WWW FIND QUOTE REPORT

11-08-2016, 03:03 PM **Post: #16**

J-F Garnier
Senior Member

Posts: 461
Joined: Dec 2013

RE: Short & Sweet Math Challenge #21: Powers that be

> **Paul Dale Wrote:** (11-08-2016 02:38 PM)
>
> If there is one real root > 1 and all of the remaining (possibly complex) roots have |root| < 1, then the polynomial is of the required form.

It may be the element I was missing, but can you explain or justify this statement?
J-F

EMAIL PM WWW FIND QUOTE REPORT

11-08-2016, 04:15 PM **Post: #17**

J-F Garnier
Senior Member

Posts: 461
Joined: Dec 2013

RE: Short & Sweet Math Challenge #21: Powers that be

Here are the results of my search on HP71/Emu71:

```
Order 2
1.61803398875 ok x^2-x-1
Order 3
1.83928675521 ok x^3-x^2-x-1
1.46557123188 ok x^3-x^2-1
1.32471795724 ok x^3-x-1
Order 4
1.92756197548 ok x^4-x^3-x^2-x-1
1.75487766625 ok x^4-x^3-x^2-1
1.61803398875 ok x^4-x^3-x-1
1.46557123188 ok x^4-x^2-x-1
Order 5
1.61803398875 ok x^5-x^4-x^3+x^2-x-1
1.32471795724 ok x^5-x^4-1
1.32471795724 ok x^5-x^2-x-1
Order 6
1.83928675521 ok x^6-x^5-x^4-x^2-x-1
1.61803398875 ok x^6-x^5-x^4+x^2-x-1
1.46557123188 ok x^6-x^5-x^4+x^3-x^2+x-1
1.61803398875 ok x^6-x^5-x^3-x-1
1.46557123188 ok x^6-x^5-x^2-1
1.46557123188 ok x^6-x^5+x^4-x^3-x^2-x-1
1.32471795724 ok x^6-x^4-x-1
Order 7
1.83928675521 ok x^7-x^6-x^5-x^4+x^3-x^2-x-1
1.61803398875 ok x^7-x^6-x^5+x^2-x-1
...
1.32471795724 ok x^8-x^6-x^2-x-1
1.32471795724 ok x^8-x^5-x^4-x-1
884 candidates (1<root<2)
59 candidates (quasi-integer powers)
```

Quite disappointing since I got only ... 7 unique constants.

Here is my HP71 working program:

```
10 ! --- SSMC21 ---
20 OPTION BASE 0 @ DIM A(10)
30 C=0 @ C2=0
40 FOR D=2 TO 8
50 DISP "Order";D
60 DIM A(D) @ COMPLEX R(D-1)
70 A(0)=1
80 A(D)=-1 ! assumed...
90 K=3^(D-1) ! numbers of coefficient combinaisons
100 FOR J=0 TO K-1
110 L=J
120 ! build the coefficients
130 FOR I=D-1 TO 1 STEP -1
140 A(I)=MOD(L,3)-1 @ L=L DIV 3
150 NEXT I
160 ! find roots of polynomia A
170 MAT R=PROOT(A)
180 ! DISP "Polynomia";J
190 ! MAT DISP A
200 ! DISP "Roots"
210 ! MAT DISP R
220 X=REPT(R(D-1))
230 IF IMPT(R(D-1))=0 AND X>1.000001 AND X<2 THEN GOSUB 300
240 ! PAUSE
250 NEXT J ! next polynomia of order D
260 NEXT D ! next order polynomiae
270 DISP C;"candidates (1<root<2)"
280 DISP C2;"candidates (quasi-integer powers)"
290 END
300 ! evaluate candidate
310 'T':
320 C=C+1
330 ! DISP "Candidate x=";X
340 ! MAT DISP A
350 F=0 ! flag candidate found
360 N=20
370 Y=X^N
380 IF ABS(FP(Y)-.5)>.49 THEN F=F+1 ELSE F=0
390 N=N+1
400 IF Y<1E10 AND N<80 AND F<3 THEN 370 ! no need to go beyong 1E10 or power 80
410 ! IF X=1.38027756910 THEN PAUSE
420 IF F<3 THEN 530
430 C2=C2+1
440 FIX 11 @ DISP X;"ok"; @ STD
450 DISP " x^";STR$(D);
460 FOR I=1 TO D-1
470 IF A(I)=1 THEN DISP "+";
480 IF A(I)=-1 THEN DISP "-";
490 IF A(I)<>0 THEN DISP "x";
500 IF A(I)<>0 AND D-I<>1 THEN DISP "^";STR$(D-I);
510 NEXT I
520 DISP STR$(A(D))
530 ! PAUSE
540 RETURN
```

EMAIL PM WWW FIND QUOTE REPORT

11-08-2016, 06:24 PM **Post: #18**

J-F Garnier
Senior Member

Posts: 461
Joined: Dec 2013

RE: Short & Sweet Math Challenge #21: Powers that be

> **Mike (Stgt) Wrote:** ⇒ (11-08-2016 04:29 PM)
> And for a(0) = +1, well... then the restriction of the leading coefficient to +1 is thwarted as we look for the maximum _value_ of the root in the range]1..2[. No?

Well, not necessarily. For instance, 1.32471795724 (one of the constants) is a solution of x^4-x^3-x^2+1.

Anyway, in my code above I assumed a(0)=-1:

```
80 A(D)=-1 ! assumed...
```

because all examples posted by Valentin and Gerson are so :-)

J-F

EMAIL PM WWW FIND QUOTE REPORT

11-08-2016, 07:52 PM **Post: #19**

Gerson W. Barbosa
Senior Member

Posts: 1,361
Joined: Dec 2013

RE: Short & Sweet Math Challenge #21: Powers that be

> **J-F Garnier Wrote:** (11-08-2016 03:03 PM)
>
> > **Paul Dale Wrote:** (11-08-2016 02:38 PM)
> >
> > If there is one real root > 1 and all of the remaining (possibly complex) roots have |root| < 1, then the polynomial is of the required form.
>
> It may be the element I was missing, but can you explain or justify this statement?
> J-F

The following is a test for 4-th order polynomials, using Pauli's and Mike's conditions. I have assumed the solutions given by PROOT are ordered by magnitude, but I am not sure about that. But this implementation is probably wrong since it gives only two solutions (three when the program is modified for 3-rd order polynomials).

Hopefully no typing errors since the EMU71 version I have doesn't work on Windows 10 64-bit and I don't know how to copy and paste the listing in DosBox.

```
------------------
3 DESTROY ALL
5 INTEGER A,B,C,D,E,N,T
7 A=1 @ E=-1
10 OPTION BASE 1
15 INTEGER P(5)
20 COMPLEX R(4)
25 FOR B=-1 TO 1
30 FOR C=-1 TO 1
35 FOR D=-1 TO 1
40 P(1)=A @ P(2)=B @ P(3)=C @ P(4)=D @ P(5)=E
45 MAT R=PROOT(P)
50 IF IMPT(R(4))=0 THEN GOSUB 1000
55 NEXT D
60 NEXT C
70 NEXT B
75 END
1000 IF REPT(R(4))>1 THEN GOSUB 2000
1005 RETURN
2000 T=0
2005 FOR N=1 TO 3
2010 IF ABS(R(N))<1 THEN T=T+1
2015 NEXT N
2020 IF T=3 THEN PRINT REPT(R(4));A;B;C;D;E
2025 RETURN

>RUN

1.92756197548 1 -1 -1 -1 -1
1.3802775691 1 -1 0 0 -1
--------------

3-rd order polynomial solutions:

1.83928675521 1 -1 -1 -1
1.46557123188 1 -1 0 -1
1.32471795721 1 0 -1 -1

-------------
```

EMAIL PM FIND QUOTE REPORT

11-08-2016, 11:24 PM **Post: #20**

Dwight Sturrock
Member

Posts: 134
Joined: Dec 2013

RE: Short & Sweet Math Challenge #21: Powers that be

> **Paul Dale Wrote:** (11-08-2016 02:38 PM)
>
> I'd figured out an approach to this problem. I've not implemented it on any hardware.
>
> It is essentially a brute force approach to the problem:

For each length of polynomial (1 .. 8):

- Iterate over all polynomials for the given length that satisfy the specified constraints. That the leading coefficient is 1, the constant term is ±1 and the remaining term coefficients are from {-1, 0, 1}.
- Over all lengths there are 6560 such polynomials. If the constant term can also be 0, the count is 9840.
- Find the roots of each polynomial.
- If there is one real root > 1 and all of the remaining (possibly complex) roots have |root| < 1, then the polynomial is of the required form.

My approach is similar to Paul's. One subroutine sequentially cycles through all coefficients - 1, 0, -1, starting with 1 1 1 1 1 1 1 1 1. The other routine uses the solver to check for roots in the proscribed range. Coded for the 15C, the routines appear to work independently, but not sure how they will behave together across the whole range.

EMAIL PM FIND QUOTE REPORT

11-08-2016, 11:48 PM **Post: #21**

Paul Dale
Senior Member

Posts: 1,662
Joined: Dec 2013

RE: Short & Sweet Math Challenge #21: Powers that be

> **J-F Garnier Wrote:** (11-08-2016 03:03 PM)
>
> > **Paul Dale Wrote:** (11-08-2016 02:38 PM)
> >
> > If there is one real root > 1 and all of the remaining (possibly complex) roots have |root| < 1, then the polynomial is of the required form.
>
> It may be the element I was missing, but can you explain or justify this statement?
> J-F

I'll try to justify the statement.

Newton's identities allow one to calculate the sum of a power of the roots of a polynomial from its coefficients. The constraints on the polynomials given here (monic with integral coefficients) mean that the sum of the roots to any power will be an integer.

For one root to approximate an integer as the power is raised, it must be the single dominant root. i.e. all other roots must approach zero as they are raise to higher and higher powers. Thus, their absolute value must be strictly less than unity. The dominant root must be one or greater.

Pauli

PM FIND QUOTE REPORT

11-09-2016, 06:45 PM **Post: #22**

J-F Garnier
Senior Member

Posts: 461
Joined: Dec 2013

RE: Short & Sweet Math Challenge #21: Powers that be

Following Paul's approach, I modified my Emu71 program to explore these solutions.
I kept the possibility for the least parameter a0 to be +/-1.

Results:

```
Order 2
1.61803398875 ok x^2-x-1
Order 3
1.83928675521 ok x^3-x^2-x-1
1.46557123188 ok x^3-x^2-1
1.32471795724 ok x^3-x-1
Order 4
1.92756197548 ok x^4-x^3-x^2-x-1
1.38027756910 ok x^4-x^3-1
Order 5
1.96594823665 ok x^5-x^4-x^3-x^2-x-1
1.88851884548 ok x^5-x^4-x^3-x^2-1
1.77847961614 ok x^5-x^4-x^3-x^2+1
1.81240361927 ok x^5-x^4-x^3-x-1
1.70490277604 ok x^5-x^4-x^3-1
```

```
1.44326879127 ok x^5-x^4-x^3+x^2-1
1.57014731220 ok x^5-x^4-x^2-1
1.53415774491 ok x^5-x^3-x^2-x-1
Order 6
1.98358284342 ok x^6-x^5-x^4-x^3-x^2-x-1
1.91118343669 ok x^6-x^5-x^4-x^3-x-1
1.80750202302 ok x^6-x^5-x^4-x^3+1
1.71428532914 ok x^6-x^5-x^4-x^2+1
1.74370016590 ok x^6-x^5-x^4-x-1
1.50159480354 ok x^6-x^5-x^4+x^2-1
1.66040772247 ok x^6-x^5-x^3-x^2-1
Order 7
1.99196419661 ok x^7-x^6-x^5-x^4-x^3-x^2-x-1
1.97504243425 ok x^7-x^6-x^5-x^4-x^3-x^2-1
1.92212800436 ok x^7-x^6-x^5-x^4-x^2-x-1
1.88004410997 ok x^7-x^6-x^5-x^4-x-1
1.85454747658 ok x^7-x^6-x^5-x^4-1
1.74745742449 ok x^7-x^6-x^5-x^4+x^3-1
1.77452005864 ok x^7-x^6-x^5-x^3-1
1.67752174784 ok x^7-x^6-x^5-x^2+1
1.65363458677 ok x^7-x^6-x^5-1
1.54521564973 ok x^7-x^6-x^5+x^2-1
1.60134733379 ok x^7-x^6-x^4-x^2-1
1.59000537390 ok x^7-x^5-x^4-x^3-x^2-x-1
Order 8
1.99603117974 ok x^8-x^7-x^6-x^5-x^4-x^3-x^2-x-1
1.97061782036 ok x^8-x^7-x^6-x^5-x^4-x^3-1
1.96113576083 ok x^8-x^7-x^6-x^5-x^4-x^3+1
1.92172206590 ok x^8-x^7-x^6-x^5-x^4+1
1.90988759678 ok x^8-x^7-x^6-x^5-x^4+x+1
1.88166942009 ok x^8-x^7-x^6-x^5-x^3+1
1.88670294847 ok x^8-x^7-x^6-x^5-x^2-x-1
1.86221396917 ok x^8-x^7-x^6-x^5-x-1
1.84771602509 ok x^8-x^7-x^6-x^5-1
1.83032136835 ok x^8-x^7-x^6-x^5+1
1.83524591514 ok x^8-x^7-x^6-x^4-x^3-x-1
1.74243322086 ok x^8-x^7-x^6-x^4+1
1.65524582449 ok x^8-x^7-x^6-x^2+1
1.57367896839 ok x^8-x^7-x^6+x^2-1
1.72778030821 ok x^8-x^7-x^5-x^4-x^3-x^2-1
1971 candidates (1<root<2)
48 candidates (dominant root)
```

An interesting point is that solutions with the last parameter a0=+1 are found.
If solutions are restricted to a0=-1, there are only 37 candidates.
Another open point is that some solutions found by my pure numerical method are not there, for instance:

```
1.75487766625 ok x^4-x^3-x^2-1
```

My HP71 program (actually close to Gerson's program, but I explored all roots):

```
10 ! --- SSMC21 ---
20 OPTION BASE 0 @ DIM A(10)
30 C=0 @ C2=0
40 FOR D=2 TO 8
50 DISP "Order";D
60 DIM A(D) @ COMPLEX R(D-1)
70 A(0)=1
80 ! A(D)=-1 ! a0=-1 assumed...
90 K=3^(D-1) ! numbers of coefficient combinaisons
100 K=K*2 ! for a0=+/-1
110 FOR J=0 TO K-1
120 L=J
130 ! build the coefficients
140 IF MOD(L,2) THEN A(D)=1 ELSE A(D)=-1 ! for a0=+/-1
150 L=L DIV 2 ! for a0=+/-1
160 FOR I=D-1 TO 1 STEP -1
170 A(I)=MOD(L,3)-1 @ L=L DIV 3
180 NEXT I
190 ! find roots of polynomial A
200 MAT R=PROOT(A)
210 FOR K=0 TO D-1
220 X=REPT(R(K))
230 IF IMPT(R(K))=0 AND X>1.000001 AND X<2 THEN GOSUB 300
```

```
240 NEXT K
250 NEXT J ! next polynomial of order D
260 NEXT D ! next order polynomials
270 DISP C;"candidates (1<root<2)"
280 DISP C2;"candidates (dominant root)"
290 END
300 'T': ! evaluate candidate
310 C=C+1
320 F=1
330 FOR I=0 TO D-1
340 IF I<>K AND ABS(R(I))>=1 THEN F=0
350 NEXT I
360 IF F=0 THEN 480
370 C2=C2+1
380 FIX 11 @ DISP X;"ok"; @ STD
390 DISP " x^";STR$(D);
400 FOR I=1 TO D-1
410 IF A(I)=1 THEN DISP "+";
420 IF A(I)=-1 THEN DISP "-";
430 IF A(I)<>0 THEN DISP "x";
440 IF A(I)<>0 AND D-I<>1 THEN DISP "^";STR$(D-I);
450 NEXT I
460 IF A(D)>0 THEN DISP "+";
470 DISP STR$(A(D))
480 RETURN
```

J-F

EMAIL PM WWW FIND QUOTE REPORT

11-10-2016, 12:53 AM **Post: #23**

Valentin Albillo
Senior Member

Posts: 636
Joined: Feb 2015
Warning Level: 0%

RE: Short & Sweet Math Challenge #21: Powers that be

Hi all:

About a week has elapsed since I posted **S&SMC#21** and in that time some of you have contributed a number of interesting ideas and even actual code implementing them, as well as relevant discussion.

Thanks so much for your interest and contributions, now is the time to give my own *original solution* to the challenge, but first I'll present some theory which will help understand the algorithm I used.

The challenge asked for you to write a program which should find and output all the constants in the range >1 and <2 with the property that their increasing integer powers would be nearer and nearer to *integer* values, outputting both each constant and its minimal polynomial, limited to degree **8** or less and coefficients **-1**,**0**,**+1**, with the leading coefficient in particular being always **+1**. The *Golden Ratio* (Phi) was given as an example of such a constant.

J-F Garnier tried the natural approach of constructing all -1,0,+1 polinomials up to degree 8, finding a candidate root, then computing its increasing integer powers and examining the fractional parts to see if they were converging to either 0.0000... or 0.9999.....

While natural, this approach is doomed to fail because the exponentially growing powers quickly exceed the accuracy of any calculator and the fractional parts get inaccurate or even lost, thus the test eventually fails.

The proper way comes by the way of a little theory, which I'll succintly explain with just the minimum amount of detail to make it short. Given the integer coefficients a(n) of a monic polynomial (leading term 1), say:

```
x^n + a(n-1)*x^(n-1) + ... + a(2)*x^2 + a(1)*x + a(0)
```

any *symmetric* function of the roots can be expressed in terms of the coefficients, where symmetric means that the value of the function is the same (invariant) for all permutations of the roots. For example, the function:

```
Sum of the roots = x1 + x2 + x3+ ... + xN
```

is a symmetric function because its value is the same no matter how the roots are permuted, so it can be computed as an integer function of the coefficients, in this case:

```
Sum of the roots = x1 + x2 + ... + xN = -a(n-1)
```

*(Example: Sum of the roots of [x^3-6*x^2+11*x-6 = 0] = -(-6) = +6. The roots are 1,2,3 and 1+2+3 = 6)*

The same goes for the sum of the roots raised to any integer power, such as for example:

```
Sum of the 10-th powers of the roots = x1^10 + x2^10 + ... + xN^10
```

which, being a symmetric funtion, can also be computed as an integer expression of the coefficients, the expression being more complicated but still delivering an integer result, thus we have:

```
x1^10 + x2^10+ ... + xN^10 = integer value
```

no matter what the roots are and no matter if they are real, complex, or a mix. Now consider what happens when one of the roots is real with an absolute value > 1 while all other roots (real or complex) have an absolute value < 1, such as is the case for the *Golden Ratio*, namely:

```
Min. Polynomial : x^2-x-1
root x1 : +1.61803398875
root x2 : -0.61803398875

Sum of their Nth powers S(N) = x1^N + x2^N = 1.61803398875^N + (-0.61803398875)^N
```

If we the compute S(N) for N=10, 20, 30, .. on a 12-digit HP calc we get:

```
1.61803398875^10 + (-0.61803398875)^10 = 122.991869381 + 0.0081306187558 = 123 exactly
1.61803398875^20 + (-0.61803398875)^20 = 15126.9999339 + 0.0000661069613 = 15127 exactly
1.61803398875^30 + (-0.61803398875)^30 = 1860498.00000 + 0.0000005374904 = 1860498 exactly
```

and you see what's happening: the powers of the root with absolute value > 1 grow *larger and larger* while the powers of the root with absolute value < 1 grow *smaller and smaller*. As their sum has to be an *integer* value and the smaller root's powers are contributing less and less to the sum, the powers of the larger root are forced to approach the integer value of the whole sum by themselves and thus be almost-integer.

This generalizes to all monic polynomials of any degree so that the criterium is that the monic polynomial of degree N must have one real root of absolute value > 1 and all its other N-1 roots, whether real or complex, must have absolute values <1. This way the sums of the powers of the smaller roots will tend to 0 and the powers of the one root with absolute value > 1 will tend to the integer sum and so will tend to being integer themselves.

The algorithm is now quite clear:

- iterate through all monic polynomials of coefficients 1,0,+1 and degrees up to 8 (per the challenge)
- for each polynomial find all its roots and check:
    - that there's only one root of absolute value >1
    - that all other roots have absolute values < 1
- if such a root is found, check if its absolute value is < 2 (per the challenge requirement)
    - if it is, record both the root and its minimal polynomial
- after all polynomials have been examined, sort and ouput the recorded roots/polynomials.

and this is my **11-line** generic implementation for the **HP-71B** (39 statements, 587 bytes), readily adaptable to any **RPL** and **RPN** HP models with a polynomial root-finding capability:

```
>CAT

SSMC21 BASIC 587 bytes

>LIST

 1  DESTROY ALL @ OPTION BASE 1 @ INPUT L @ DIM P(L+1),F(L),T$[60] @ COMPLEX R(L)
 2  INPUT "Sort by C,P ? ","C";X$ @ K=16*(X$="P") @ N=0 @ P(1)=1 @ H=0 @ SETTIME 0
 3  H=H+1 @ F(H)=-1 @ REPEAT @ P(H+1)=F(H) @ IF H<L THEN 3 ELSE MAT R=PROOT(P)
 4  X=0 @ FOR I=1 TO L @ IF ABS(R(I))<1 THEN 5 ELSE IF X THEN 6 ELSE X=REPT(R(I))
 5  NEXT I @ IF X>1 AND X<2 THEN N=N+1 @ DIM S$(N)[60] @ Z=L+2 @ GOSUB 9
 6  F(H)=F(H)+1 @ UNTIL F(H)>(H#1 AND H#L) @ H=H-1 @ IF H THEN 6
 7  FOR I=1 TO N @ FOR J=I+1 TO N @ IF S$(I)[K]>S$(J)[K] THEN VARSWAP S$(I),S$(J)
 8  NEXT J @ NEXT I @ FOR I=1 TO N @ DISP S$(I) @ NEXT I @ DISP N;TIME @ END
 9  Z=Z-1 @ IF NOT P(Z) THEN 9 ELSE FIX 11 @ X$=" "&STR$(X)&" " @ STD @ T$=""
10  FOR I=1 TO L+1 @ IF P(I) THEN J=2-(P(I)>0) @ T$=T$&"+-"[J,J]&"x^"&STR$(Z-I)
11  NEXT I @ S$(N)=REPLACE$(REPLACE$(X$&T$[2],"^1",""),"x^0","1") @ RETURN
```

(Lines 1-2 do the *initialization*, lines 3-6 do the *search*, lines 7-8 do the *sorting* and *output*, lines 9-11 *format* the output.)

It uses the **Math ROM** (*PROOT*) and the **JPCROM** for structured statements (*REPEAT*, *UNTIL*) and assorted utility functions (*VARSWAP*, *REPLACE$*). The JPCROM functions are mere conveniences and can be dispensed with but PROOT is mandatory or else the program would be much longer and slower.

My program, as written, is *generic* for monic -1,0,+1 polynomials of *any degree*, not just up to 8, and can display its output *sorted* by either constant or (for degree < 10) by minimal polynomial, as well as providing a count of the constants found and timing.

For example, for polynomials up to **degree 4**, sorting by constant:

```
>RUN

? 4
Sort by C,P ? C

1.32471795724 x^3-x-1
1.38027756910 x^4-x^3-1
1.46557123188 x^3-x^2-1
1.61803398875 x^2-x-1
1.83928675521 x^3-x^2-x-1
1.92756197548 x^4-x^3-x^2-x-1

6 .76
```

i.e., it found **6** constant in 0.76 seconds (**Emu71** on an old 2.4 GHz single-core).
The same run sorted by polynomial (same count and timing):

```
>RUN

? 4
Sort by C,P ? P

1.61803398875 x^2-x-1
1.32471795724 x^3-x-1
1.46557123188 x^3-x^2-1
1.83928675521 x^3-x^2-x-1
1.38027756910 x^4-x^3-1
1.92756197548 x^4-x^3-x^2-x-1
```

For polynomials up to **degree 6**, sorted by constant:

```
>RUN

? 6
Sort by C,P ? C

1.32471795724 x^3-x-1
1.38027756910 x^4-x^3-1
1.44326879127 x^5-x^4-x^3+x^2-1
1.46557123188 x^3-x^2-1
1.50159480354 x^6-x^5-x^4+x^2-1
1.53415774491 x^5-x^3-x^2-x-1
1.57014731220 x^5-x^4-x^2-1
1.61803398875 x^2-x-1
1.66040772247 x^6-x^5-x^3-x^2-1
1.70490277604 x^5-x^4-x^3-1
1.74370016590 x^6-x^5-x^4-x-1
1.77847961614 x^5-x^4-x^3-x^2+1
1.81240361927 x^5-x^4-x^3-x-1
1.83928675521 x^3-x^2-x-1
1.88851884548 x^5-x^4-x^3-x^2-1
1.91118343669 x^6-x^5-x^4-x^3-x-1
1.92756197548 x^4-x^3-x^2-x-1
1.96594823665 x^5-x^4-x^3-x^2-x-1
1.98358284342 x^6-x^5-x^4-x^3-x^2-x-1
```

i.e.: **19** constants found in 15.25 seconds.

Finally, for polynomials up to **degree 8** sorted by constant, as per the challenge requirements:

```
>RUN
```

```
? 8
Sort by C,P ? C

1.32471795724 x^3-x-1
1.38027756910 x^4-x^3-1
1.44326879127 x^5-x^4-x^3+x^2-1
1.46557123188 x^3-x^2-1
1.50159480354 x^6-x^5-x^4+x^2-1
1.53415774491 x^5-x^3-x^2-x-1
1.54521564973 x^7-x^6-x^5+x^2-1
1.57014731220 x^5-x^4-x^2-1
1.57367896839 x^8-x^7-x^6+x^2-1
1.59000537390 x^7-x^5-x^4-x^3-x^2-x-1
1.60134733379 x^7-x^6-x^4-x^2-1
1.61803398875 x^2-x-1
1.65363458677 x^7-x^6-x^5-1
1.66040772247 x^6-x^5-x^3-x^2-1
1.67752174784 x^7-x^6-x^5-x^2+1
1.70490277604 x^5-x^4-x^3-1
1.71428532914 x^6-x^5-x^4-x^2+1
1.72778030821 x^8-x^7-x^5-x^4-x^3-x^2-1
1.74370016590 x^6-x^5-x^4-x-1
1.74745742449 x^7-x^6-x^5-x^4+x^3-1
1.77452005864 x^7-x^6-x^5-x^3-1
1.77847961614 x^5-x^4-x^3-x^2+1
1.80750202302 x^6-x^5-x^4-x^3+1
1.81240361927 x^5-x^4-x^3-x-1
1.83524591514 x^8-x^7-x^6-x^4-x^3-x-1
1.83928675521 x^3-x^2-x-1
1.84771602509 x^8-x^7-x^6-x^5-1
1.85454747658 x^7-x^6-x^5-x^4-1
1.86221396917 x^8-x^7-x^6-x^5-x-1
1.88004410997 x^7-x^6-x^5-x^4-x-1
1.88670294847 x^8-x^7-x^6-x^5-x^2-x-1
1.88851884548 x^5-x^4-x^3-x^2-1
1.91118343669 x^6-x^5-x^4-x^3-x-1
1.92212800436 x^7-x^6-x^5-x^4-x^2-x-1
1.92756197548 x^4-x^3-x^2-x-1
1.96594823665 x^5-x^4-x^3-x^2-x-1
1.97061782036 x^8-x^7-x^6-x^5-x^4-x^3-1
1.97504243425 x^7-x^6-x^5-x^4-x^3-x^2-1
1.98358284342 x^6-x^5-x^4-x^3-x^2-x-1
1.99196419661 x^7-x^6-x^5-x^4-x^3-x^2-x-1
1.99603117974 x^8-x^7-x^6-x^5-x^4-x^3-x^2-x-1
```

i.e.: it found **41 constants** in 239.96 seconds, just shy of 4 min. For fun, a run for polynomials up to **degree 10** finds **85** constants in less than an hour.

That's all. Thanks again for your contributions, I hoped you enjoyed the challenge, see you all in **S&SMC#22**.

Best regards.
V.

Find All My HP-related Materials here: **Valentin Albillo's HP Collection**

PM WWW FIND EDIT QUOTE REPORT

11-10-2016, 07:51 AM (This post was last modified: 11-10-2016 07:52 AM by Paul Dale.) **Post: #24**

Paul Dale
Senior Member

Posts: 1,662
Joined: Dec 2013

RE: Short & Sweet Math Challenge #21: Powers that be

Valentin, your explanation is much better than mine but at least I was on the right path.

An interesting challenege as always.

- Pauli

PM FIND QUOTE REPORT

11-10-2016, 06:47 PM (This post was last modified: 11-10-2016 06:55 PM by J-F Garnier.) **Post: #25**

J-F Garnier
Senior Member

Posts: 461
Joined: Dec 2013

RE: Short & Sweet Math Challenge #21: Powers that be

> **Valentin Albillo Wrote:** (11-10-2016 12:53 AM)
>
> ```
> >RUN
>
> ? 8
> Sort by C,P ? C
>
> 1.32471795724 x^3-x-1
> 1.38027756910 x^4-x^3-1
> ...
> 1.99196419661 x^7-x^6-x^5-x^4-x^3-x^2-x-1
> 1.99603117974 x^8-x^7-x^6-x^5-x^4-x^3-x^2-x-1
> ```
>
> i.e.: it found **41 constants** in 239.96 seconds, just shy of 4 min.

It was an interesting and fun challenge, and the opportunity for me to learn something in math.

I have two comments:

1) It seems that your program doesn't explore all the polynomials. It uses basically the same algorithm that I used in my previous thread (based on Paul's idea), and I got not **41** but **48** constants.
I had a look at your program, but had some difficulties to understand the logic of the polynomial scanning. However, I was able to hack your program to display all the tested polynomials, and indeed some are missing:

```
? 4
Sort by C,P ? p
1.00000000000 1
1.00000000000 x-1
1.00000000000 x^2+1
1.00000000000 x^2-1
1.00000000000 x^2-x+1
1.00000000000 x^2-x-1
1.00000000000 x^3+1
1.00000000000 x^3+x+1
1.00000000000 x^3+x-1
1.00000000000 x^3-1
1.00000000000 x^3-x+1
1.00000000000 x^3-x-1
1.00000000000 x^3-x^2+1
1.00000000000 x^3-x^2+x+1
1.00000000000 x^3-x^2+x-1
1.00000000000 x^3-x^2-1
1.00000000000 x^3-x^2-x+1
1.00000000000 x^3-x^2-x-1
1.00000000000 x^4+x-1
1.00000000000 x^4+x^2+x-1
1.00000000000 x^4+x^2-1
1.00000000000 x^4+x^2-x-1
1.00000000000 x^4-1
1.00000000000 x^4-x-1
1.00000000000 x^4-x^2+x-1
1.00000000000 x^4-x^2-1
1.00000000000 x^4-x^2-x-1
1.00000000000 x^4-x^3+x-1
1.00000000000 x^4-x^3+x^2+x-1
1.00000000000 x^4-x^3+x^2-1
1.00000000000 x^4-x^3+x^2-x-1
1.00000000000 x^4-x^3-1
1.00000000000 x^4-x^3-x-1
1.00000000000 x^4-x^3-x^2+x-1
1.00000000000 x^4-x^3-x^2-1
```

```
1.00000000000 x^4-x^3-x^2-x-1
36
```

For instance, the polynomials x^2+x+1 and x^2+x-1 are not tested, as well as the x^3+x^2... polynomials.

2) my second comment is what I already mentioned in my previous message: the value 1.75487766625, root of x^4-x^3-x^2-1, is not found by this algorithm, although its powers clearly tend to an integer (quite quickly actually):

```
x^10 276.992792634
x^11 486.088465506
...
x^21 134643.001528
x^22 236281.996298
...
x^27 3932464.99924
x^28 6900995.00047
```

The roots of the polynomial x^4-x^3-x^2-1 are:

```
(.122561166877,-.74486176662)
(.122561166877,.74486176662)
(-1,0)
(1.75487766625,0)
```

Clearly, the roots 1 or -1 should not cause to reject the polynomial under test, since powers of -1 or +1 will not impact the decimals of the powers.

Thanks again for this challenge!

J-F

EMAIL PM WWW FIND QUOTE REPORT

11-11-2016, 12:13 AM **Post: #26**

Valentin Albillo
Senior Member

Posts: 636
Joined: Feb 2015
Warning Level: 0%

RE: Short & Sweet Math Challenge #21: Powers that be

Hi, **Mike**:

> **Mike (Stgt) Wrote:** (11-10-2016 10:51 AM)
> So the reason for the almost integer effect of increasing powers is just the "symetry" of roots r1 and r2 with abs(r1)<1 and abs(r2)>1 ?

The reason for the *almost-integer i*ncreasing powers is, in short:

- all *symmetric functions* of the roots of any polynomial can be expressed in terms of the coefficients of the polynomial, where *symmetric* means that the function remains the same and has the *same* value for *every* permutation of the roots. For instance, the sum of the Nth-powers of the roots of a quintic polynomial is symmetric because:

```
S = x1^5 + x2^5 + x3^5 + x4^5 + x5^5     (initial permutation)
  = x1^5 + x2^5 + x3^5 + x5^5 + x4^5     (second permutation)
                  ...
  = x5^5 + x4^5 + x3^5 + x2^5 + x1^5     (last permutation)
```

so you see, no matter how you permute the roots the sum stays the same.

- if the polynomial is *monic* (leading coefficient is 1) and has *integer* coefficients, then any symmetric function of the roots is *integer* as well because it can be computed from the integer coefficients using just addition, subtraction and multiplication of their integer values. Thus, in particular, *the sum of the roots raised to any integer power is mandatorily integer* as well, regardless of the absolute values of the roots or whether they're real and/or complex.

-now, if any number is >1 in absolute value, its increasing powers will grow ever *bigger*, while if the number is <1 in absolute value its increasing powers will grow ever *smaller*, tending to 0.

- so, for those monic polynomials which have just the **one** root >1 in absolute value while all other roots are <1 in absolute value, the sum of the powers of the roots will essentially equal the power of the large root alone plus change, and as the sum has to mandatorily be integer, then the power of the larger root tends to be integer, *almost-integer* so to say. The only difference with an integer is the sum of the powers of all other roots, whichs tends to 0 for each root and so for the sum of all of them.

I hope this explanation makes it clear for you, thanks a lot for your interest.

Reegards.
V.

Find All My HP-related Materials here: **Valentin Albillo's HP Collection**

PM WWW FIND EDIT QUOTE REPORT

11-11-2016, 12:15 AM **Post: #27**

Valentin Albillo
Senior Member

Posts: 636
Joined: Feb 2015
Warning Level: 0%

RE: Short & Sweet Math Challenge #21: Powers that be

Hi, **Pauli**:

> **Paul Dale Wrote:** (11-10-2016 07:51 AM)
> Valentin, your explanation is much better than mine but at least I was on the right path.

Absolutely, you nailed it.

> **Quote:**
> An interesting challenege as always.

Thank you very much for your kind comment and interest, hope to see you back in S&SMC#22 due next month or so.

Regards.
V.

Find All My HP-related Materials here: **Valentin Albillo's HP Collection**

PM WWW FIND EDIT QUOTE REPORT

11-11-2016, 01:10 AM **Post: #28**

Valentin Albillo
Senior Member

Posts: 636
Joined: Feb 2015
Warning Level: 0%

RE: Short & Sweet Math Challenge #21: Powers that be

Hi, **J-F**:

> **J-F Garnier Wrote:** (11-10-2016 06:47 PM)
> I have two comments:
>
> 1) It seems that your program doesn't explore all the polynomials. [...] However, I was able to hack your program to display all the tested polynomials, and indeed some are missing: [...] For instance, the polynomials x^2+x+1 and x^2+x-1 are not tested, as well as the x^3+x^2... polynomials.

I made use of mathematical equivalences and polynomial theory to prune the number of polynomials to be tested to the barest minimum necessary. For instance, your polynomial:

```
x^2+x+1
```

doesn't have any real roots, just two conjugate complex roots. For polynomials with real coefficients, all complex roots come out in conjugate pairs which of course have the same absolute value and thus can't be *unique*, i.e., even if their absolute value were >1 there are *two* of them, not just *one* as required.

Your other polynomial, namely:

```
x^2+x-1
```

is equivalent to **x^2-x-1** if you simply change the variable from **x** to **-x**, which of course doesn't alter its absolute value at all, so you only need to generate and check **x^2-x-1**, which I did. In short, I choose the coefficients in such a way that those redundant polynomials aren't generated while searching.

> **Quote:**
> 2) my second comment is what I already mentioned in my previous message: the value 1.75487766625, root of x^4-x^3-x^2-1, is not found by this algorithm, although its powers clearly tend to an integer (quite quickly actually)

Your number **1.75487766625** is indeed a bona-fide **Pisot number** (which is the proper mathematical name for the numbers which have this *almost-integer* powers property) as can be readily checked like you did.

However, although it's indeed a root of the **4th-degree** polynomial you mention, namely:

```
x^4-x^3-x^2-1
```

this is *not* its *minimal polynomial*. Your number is a root of infinite polynomials with integer coefficients but *only one of them has the minimum degree possible*, and in this case it is the **3rd-degree** polynomial:

x^3-2*x^2+x-1

which neither your second program nor my original solution generates or checks because it is *not* a **-1,0,+1** polynomial as per the challenge requirements because it does have a **-2** coefficient. That's why neither your program nor mine should have found it and *that's correct*, the -2 coefficient places it squarely out of the specified requirements.

> **Quote:**
> Thanks again for this challenge!

You're welcome, I'm glad you enjoyed it and thank you very much for your interest, your kind comments and, above all, the time you took to deal with it, much appreciated. Hope to see you back in **S&SMC#22** due next month or so.

Best regards.
V.

Find All My HP-related Materials here: **Valentin Albillo's HP Collection**

User(s) browsing this thread: Valentin Albillo*