Generating Sequences in the Style of OEIS

Gloria Liu

May 2, 2024

1 Project Description

The project involves generating sequences (up to thirty terms) from the constant term of polynomials of the form $(a * x + b + \frac{c}{x})^n$, where $igcd(a, b, c) = 1$, for $-A < a, b, c < A$. I also find the linear recurrences of these sequences and check which ones are already in the Online Encyclopedia of Integer Sequences (OEIS).

For example, for $a = 1, b = 2, c = 3$, the sequence generated would be:

$2, 10, 44, 214, 1052, 5284, 26840, 137638, 710828, 3692140, 19266920, 100932220, 530479640 \ldots$

2 Generating the Sequences

An individual sequence can be generated simply using the following line of code in Maple:

```
Seq:= proc(a, b, c, K) local n, x:
  [seq(coeff(expand((a*x + b + c/x)^n), x, 0), n = 1 .. K)]:
end:
```

To generate many sequences at once, I used for loops for different values of a, b, and c, checking their gcd before generating the sequences.

3 OEIS Lookup and Other Functions

I used an existing Python resource that could programatically query the OEIS with a given list of integers. I added on to the original by being able to query many sequences at once, by extracting them from a csv file, and then adding a column that indicates what OEIS sequence it corresponds to (if there exists one). This is also why I exported the sequences into a csv file in Maple. Once querying in the OEIS is done in the Python environment, I can import the csv file back into Maple and do further analysis there.

At one point, there was an OEIS Maple package, that I could not use because it is unfortunately not maintained anymore.

3.1 Lookup

I created two lookup functions. Given a database of sequences, you can look up a sequence by entering a few terms as a list. I manually check over each term in the input and compare it with the sequences in the database. If a database also has the OEIS reference number, it's possible to look up a sequence that way as well.

3.2 Linear Recurrences

I used the Almkvist-Zeilberger algorithm to find linear recurrences, specifically the function $AZd((a + b/x + c * x)^n/x, n, N)$[1].

All of these sequences have linear recurrences.

4 Observations

For a given a, b, and c, there are two sequences that are generated (with terms that are off by a sign) if we take different combinations of $\pm a$, $\pm b$ and $\pm c$.

The sequences generated by (a, b, c), $(a, -b, c)$, $(-a, b, -c)$ are the same, with some terms off by a sign.

The sequences generated by $(-a, b, c)$, $(a, b, -c)$, $(-a, -b, c)$ and $(a, -b, -c)$ are the same with some terms off by a sign.

Also, (a, b, c) and (c, b, a) will result in the same sequence.

When looking up a sequence on OEIS, a result sequence with the same terms but different signs are sometimes shown. The sequences are different but for the sake of simplicity I tried to only record them once.

The biggest A I ran is 10. It did not take long to generate the sequences, but it took a couple minutes to get the results back from the OEIS. Generally, it took more time to query the OEIS compared to generatiing the functions. Also, the greater the absolute value of a, b, and c, even below 10 I noticed that there were less associated sequences on the OEIS. So, it should be possible to generate many, many, new sequences this way!

5 References

Find recurrence relations: `https://sites.math.rutgers.edu/~zeilberg/tokhniot/EKHAD.txt`

Almkvist-Zeilberger algorithm: `https://sites.math.rutgers.edu/~zeilberg/mamarim/mamarimPDF/duis.pdf`

OEIS: `https://oeis.org/`

Original OEIS Python request: `https://paddy3118.blogspot.com/2023/08/oeisify.html`

Maple

Python