The OEIS [1] is an indispensable tool for identifying an integer sequence when one knows a few of its terms. However, there are potential applications of the OEIS that are currently infeasible because of the way data is collected and stored. By better structuring some of its data, the OEIS can become a useful tool for identifying and studying integer sequences in other ways as well.

### Standard lookup

“Which is this sequence 1, 1, 2, 5, 14, 42, 132, 429, … that came up in my research?”

**The On-Line Encyclopedia of Integer Sequences® (OEIS®)**

It’s the sequence of Catalan numbers. And the OEIS knows lots about it:

A(0) = A(1) = 1, A(n) = (2n+1)A(n-1)/(n+1), n = 2, 3, 4, …

In particular, the generating function is algebraic:

\[ G(x) = \frac{1 - \sqrt{1 - 4x}}{2x} \]

There are many classes of integer sequences — periodic sequences, algebraic sequences, holonomic sequences, k-automatic sequences, sequences that count lattice paths, etc. One would like to have immediate access to sequences in each of these classes, along with standardized representations of them to facilitate computation of arbitrary terms, closure properties, and programmatic surveying and analysis.

A new **Classes** field could contain this information. For the Catalan numbers, this field would contain information such as the following.

- **AlgebraicGeneratingFunction**: \( x(4x+1)A'(x) + (2x+4)x - A(x) = 0 \), \( A(0) = 1 \), \( A'(0) = 1 \)
- **HolonomicGeneratingFunction**: \( x(4x+1)A'(x) + (2x+4)x - A(x) = 0 \), \( A(0) = 1 \), \( A'(0) = 1 \)
- **HolonomicRecurrence**: \( (2x+4)x + (2x-2)x - A(x) = 0 \), \( A(0) = 1 \), \( A'(0) = 1 \)

Each line contains the name of a class, along with a specification of the sequence in that class. The consistent notation allows programmatic interpretation.

### Structured data

Data that is represented in a uniform way is immediately accessible to querying and manipulation by computer programs.

OEIS data is already structured into fields. Some fields contain data intended for humans, for example **Name**, **Comments**, and **Example**. Other fields contain precise structured data that is both human-readable and programatically accessible; namely, the **Data** field contains the first few terms of the sequence as a comma-separated list, and this precise data format is what enables the standard lookup and Superseeker.

The **Formula** field and the fields containing code (in Maple, Mathematica, etc.) contain information that is potentially useful both to humans and to programs but is currently only accessible to humans.

To make this information usable by programs, we should represent it consistently and add any missing contextual information. For example, all elements of the Mathematica field should be of the same type, and this type should be as widely applicable as possible.

OEIS editors insure that conventions are respected in new entries and bring existing entries up to date, so implementation of new conventions can be crowd-sourced.

### Potential programmatic uses of OEIS entries

We have seen that the OEIS already contains some data types that are automatically accessible by programmatic means. The consistent notation allows programmatic interpretation.

However, there is a large portion of the OEIS data that is currently inaccessible to computer programs. To identify cases such as these, the OEIS needs better structured data that is both human-readable and programmatically accessible; namely, structured data such as those that are automatically accessible.

By making OEIS data more computable, we will be able to...

- get statistical information about the relative number of sequences of various kinds that have arisen in human mathematics;
- search for sequences in new ways, allowing researchers to perform automated testing for examples or counterexamples to a given conjecture;
- do automated theorem discovery and proof by applying a general theorem to a large number of examples.

### Benefits

### References