



A Robust Database Model for Specific Number Types and an Implementation of a Huge Database for First 1 Trillion Prime Numbers

Neeraj Anant Pande

Professor, Department of Mathematics and Statistics, Yeshwant Mahavidyalaya (College), Nanded-431602, Maharashtra, INDIA

ARTICLE INFO	ABSTRACT
Published Online: 20 October 2026 Corresponding Author: Neeraj Anant Pande KEYWORDS: Number Types, Databases, Model, Accessibility Features, Prime Numbers	Numbers of different specific types are of importance in study of mathematics. For many types of numbers, a general formula for n^{th} number either doesn't exist or although it exists, it becomes impracticable to apply in view of large resource overheads required in its computation. For such numbers, there is need of database of these types of numbers which stores these numbers in succession. This work suggest a robust model database for storing such numbers and highlights various desired features it incorporates. In fact, one such implementation of this database on a huge scale is very recently implemented which is going to be of immense use for researchers in respective domain.

I. INTRODUCTION

Arithmetic and Number Theory are amongst the fundamental branches of mathematics. Both deal with numbers. More detail study of numbers along with functions and other entities associated with them occur in branches like Real Analysis, Complex Analysis etc.

Different types of numbers have been identified and their deep study has been undertaken. In fact, there is a very large and systematically classified online database for various types of integers, viz., "The On-Line Encyclopedia of Integer Sequences (OEIS)" available on URL : <https://oeis.org/>, the initiative of which has roots as back as in 1960's [1]. Although this gigantic collection - encompassing as many as over 3,90,000 types - is for types of integer only, there are similar varieties for rational, irrational, transcendental, real, complex and hyper number types. Each number type has its own importance and exhibits peculiar properties.

II. QUANTUM OF NUMBER TYPES

When any specific number type is considered, there are 3 possibilities regarding the quantity of numbers that occur in that type :

A. Number Types Having Finite Number of Members

Certain number types contain finite number of members, that is, there exist only a finite number of these

types of numbers : sometimes a finite few, sometimes just one, or even sometimes none!

1. There are only 15 supersingular primes (moonshine theory) [2], viz.,
2, 3, 5, 7, 11, 13, 17, 19, 23, 29, 31, 41, 47, 59, 71
2. There is only 1 even prime, viz.,
2
3. There is no square prime. (There just cannot be!)

B. Number Types Having Infinite Number of Members

Certain number types contain infinite number of members, that is, there exist infinitely many numbers of these types.

1. Some number types obviously contain infinitely many members : like positive integers or natural numbers, viz.,
1, 2, 3, ...
2. Some number types contain infinitely many members, which is theoretically proved : like, prime numbers, viz.,
2, 3, 5, ...

C. Number Types Having Uncertain Number of Members

Certain number types contain finite number of members that are known and may contain more members that are hereto unknown. So, whether they contain only finite number of members or infinite number of members is uncertain. Two sub-cases arise here.

1. Number types whose first few finite number of members are known, next are not known and are conjectured not to

“A Robust Database Model for Specific Number Types and An Implementation of a Huge Database for First 1 Trillion Prime Numbers”

exist : like Fermat primes [3], only first 5 of which are known till date, viz.,

3, 5, 17, 257, 65537

It conjectured that there may not exist any other Fermat primes.

2. Number types whose first few finite number of members are known, next are not known and are conjectured to be infinitely many : like repunit primes, particularly in base 10 [4], only first 5 of which are proven to be prime till date, viz.,

11 (2 digits)

11111111111111111111 (19 digits),
11111111111111111111111111 (23 digits)

and next 2 members are 317 and 1031 digit numbers having 1 in all places. The next members are proposed, but not verified to be certainly belonging to this type. Also it is conjectured that there may exist infinitely many of these.

III. THE n^{th} MEMBER

For any of the categories of number types that have been discussed, for any positive integral value of n , it is desirable to have knowledge of n^{th} member.

If a number type has only finitely many, say m , members, belonging to it : if $m < n$, there clearly doesn't exist n^{th} member and if $m \geq n$, they can be exhaustively determined and listed to get n^{th} member.

Challenge arises for those number types whose number of members are either unknown or known to be infinite.

There are 3 possibilities encountered in reaching n^{th} member of any number type.

1. There may exist a systematic and proven formula for determining n^{th} member of a number type.

There are many examples fitting in this category. Like :

n^{th} even positive integer is given by formula : $2n$.

n^{th} odd positive integer is given by formula : $2n - 1$.

n^{th} square number is given by formula : n^2 .

These types of numbers do not have a need to be explicitly stored in succession in a database for getting n^{th} member, as formula readily comes to help for this purpose.

2. There may exist a formula for determining n^{th} member of a number type, but it may not be practicable to apply it, which in turn demand a procedure or algorithm for determining it.

There are many examples fitting in this category also. Like :

n^{th} Fibonacci number [5], which has formula

$$F_n = \frac{\Phi^n - \Psi^n}{\sqrt{5}}, \text{ but it is not much useful in actual}$$

calculations as all numbers involved in it, except the power n , are irrational numbers and dealing with irrational numbers and that too their powers has different

computational challenges. Rather the recurring generating formula $F_n = F_{n-1} + F_{n-2}$ is preferred with first two Fibonacci numbers $F_1 = 0$ and $F_2 = 1$ ready in hand. But too does have an overhead of generating all Fibonacci numbers till F_n , but is doable piece of work to find this n^{th} member.

These types of numbers may or may not have a need to be explicitly stored in succession in a database for getting n^{th} member, depending upon how quick or slow and resource-wise efficient or inefficient procedure is there for reaching the n^{th} member.

3. There may not exist either a formula or a convenient procedure for determining n^{th} member of a number type. There are examples fitting in this category also. Like : n^{th} digit occurring after the decimal point of an irrational number, that too particularly of a non-computable irrational number like the Chaitin's constant (Ω) [6].

These types of numbers have a great need to be explicitly stored in succession in a database for getting n^{th} member as one cannot reach it out easily.

IV. DATABASES FOR PRIME NUMBERS

Prime numbers lack a simple formula for n^{th} member, thereby creating the necessity for creation of its database. There are few databases which host all successive prime numbers till specific limit. Two of these online databases are frequently referred, namely,

1. Database on the site with URL “http://compoasso.free.fr/primelistweb/page/prime/liste_online_en.php” [7]. It gives list of all successive prime numbers till 1,000,000,016,531 (just crossing 1 trillion). There is an choice of listing either of 100, 200, 300 or 600 successive prime numbers per page at a time. Page-wise navigation is possible. The generated list is in plain text format, where data values are separated by specific blank text.

2. The website “The PrimePages” has a webpage entitled “The Nth Prime Page” located at URL “<https://t5k.org/nthprime/>” by Andrew Booker [8] which has two functionalities :

a) generating n^{th} prime for given value of n

b) generating the value of number of primes till a given number n , technically called $\pi(n)$.

This handles n^{th} primes till 1 trillion, which covers positive integers till as many as around 30 trillion. But technically speaking, this is not a database, as this site declares that it generates required prime numbers on the fly in bunches in efficient way. There is no storage of these numbers. Nor the prime numbers can be generated in any specified range. Only one number at a time can be handled.

V. ACTUAL WORKOUT

The author has actually implemented newly proposed database model at a very huge scale for prime numbers. The author has independently & rigorously determined first 1 trillion prime numbers, spanning over the range of about 30 trillion, and stored these in the proposed model of database. In fact, the database has been made available in public domain with full accessibility in read, fetch and download mode. This has become a largest, most handy database of prime numbers across all internet till date.

VI. PROPOSED DATABASE MODEL

The new idea of this work consists of proposal of a database model for storing members of any number type in succession.

The database model mainly consist of two types of components :

1. Main data files (Files) & folders containing them
2. Meta data files (Block Files) & folders containing them

A. Main Data Files (Files) & Folders Containing Them

Main data files are in any one of the human readable format like :

1. CSV (Comma Separated Values)
 2. TSV (Tab Separated Values)
 3. PSV (Pipe Separated Values)
 4. Spreadsheet format (like Microsoft Excel, Google Sheets)
 5. JSON (JavaScript Object Notation)
 6. XML (Extensible Markup Language)
- or any such other formats.

Each of these storage formats have their well-known merits and limitations.

These main data files are level-0 data files, wherein the actual data of number types along with their serial number is stored.

The detail structure of each level-0 main data file is as follows :

The number of data items in each level-0 main data file is fixed, say m : suitably, a power of number system base 10 - like $10^3 = 1,000$ or $10^6 = 1,000,000$ etc. Author's Prime Number Database has this value as 1,000,000.

Each level-0 main data file is has data table or matrix of order $(m + 3) \times 2$, having $m + 3$ rows and 2 columns. The first 2 rows are general description rows and the third row is header row, with following structure :

Row 1 : Courtesy: VALUE, Generator: VALUE

Row 2 : Up Level Block FileID, hyperlink

Row 3 : Serial Number (n), number Type (Notation)

Each next m number of k^{th} row has following structure :

Row k : Sr. No. of number Type Number Type Number

The Courtesy goes to an establishment which has been pivotal in generating or storing or maintaining this database. Like in author's database model for prime numbers, since the database is hosted on Google clouds, the Courtesy field is set to value Google.

The Generator is the person or team of persons which has/have generated this database. This field is necessary to make the database credible. Like in author's Prime Numbers Database, since the database is generated by author himself, the Generator field is set to value of author name, namely, Prof. Neeraj Anant Pande.

The hyperlink of Up Level Block FileID takes to level-1 meta data file which holds down level links to level-0 main data files in its range and up level link to level-2 meta data file. The description of detail structure of high level meta data files follows in coming subsection.

Few lines of the very first level-0 main data file of author's database for prime numbers looks like follows :

A	B
Courtesy: Google	Generator: Prof. Neeraj Anant Pande
Up Level Block FileID	1A6gvaxrnpG_NbaI_VZMpvXQJIK1bTmUhlIZc_PN_0Q2Is
Serial Number (n)	Prime Number (p_n)
1	2
2	3
3	5
4	7
5	11
6	13
7	17
8	19
9	23
10	29

Figure 1 : Model Main Data File

This file is hosted on Google cloud on URL : ["https://docs.google.com/spreadsheets/d/1SNjkC4o5TEh0fZ6owWHwiYcP3W_sAX3xNotjHBqxxDw"](https://docs.google.com/spreadsheets/d/1SNjkC4o5TEh0fZ6owWHwiYcP3W_sAX3xNotjHBqxxDw).

Level-0 main data files, whose range is covered by their level-1 metadata file, are present in common folder having the name format as “Number Type - Serial Number Range”.

B. Meta Data Files (Block Files) & Folders Containing Them

Meta data files are also in any one of the human readable format like :

1. CSV (Comma Separated Values)
 2. TSV (Tab Separated Values)
 3. PSV (Pipe Separated Values)
 4. Spreadsheet format (like Microsoft Excel, Google Sheets)
 5. JSON (JavaScript Object Notation)
 6. XML (Extensible Markup Language)
- or any such other formats.

These meta data files are high level-1 ($l > 0$) data files, wherein information of lower level main data or meta data files is stored.

“A Robust Database Model for Specific Number Types and An Implementation of a Huge Database for First 1 Trillion Prime Numbers”

The detail structure of any level- l ($l > 0$) meta data file is as follows :

The number of lower level main data file items or meta data file items in each higher level- l meta data file is fixed, called its block-size b : suitably, a power of number system base 10 - like $10^2 = 100$ or $10^3 = 1,000$ etc. Author's Prime Number Database has this value as 1,000.

Each level- l meta data file is has data table or matrix of order $(b + 3) \times 5$, having $b + 3$ rows and 5 columns. The first 2 rows are general description rows and the third row is header row, with following structure :

Row 1 : It has 5 columns :

- Column 1 : Unit File Range (for level- l file, $l = 1$)
or Unit Block Range (for level- l file, $l > 1$)
- Column 2 : Files Generated (for level- l file, $l = 1$)
or Blocks Generated (for level- l file, $l > 1$)
- Column 3 : First Number Type
- Column 4 : Last Number Type
- Column 5 : Up Level Block FileID by #Generator# & Courtesy #Courtesy#

In column 5, #Generator# is the name of Generator and #Courtesy# is the name of firm whose courtesy is to be acknowledge, as explained in main data files subsection.

Row 2 : Its 5 columns contains the values of the fields the headers of which are described in respective columns of preceding Row 1. Column 5, preferably has hyperlink to Up Level Block FileID corresponding to level- $(l+1)$.

Row 3 : It has 5 columns :

- Column 1 : No. of First Number Type
- Column 2 : No. of Last Number Type
- Column 3 : First Number Type
- Column 4 : Last Number Type
- Column 5 : FileID by #Generator# & #Courtesy# (for level- l file, $l = 1$)
BlockFileID by #Generator# & #Courtesy# (for level- l file, $l > 1$)

In column 5, #Generator# is the name of Generator and #Courtesy# is the name of firm whose courtesy is to be acknowledge, as explained in main data files subsection.

Each next b number of k^{th} row has following structure :

Row k : It has 5 columns :

- Column 1 : Value of Sr. No. of First Number Type
- Column 2 : Value of Sr. No. of Last Number Type
- Column 3 : Value of First Number Type
- Column 4 : Value of Last Number Type
- Column 5 : Value (Preferably hyperlinked) of FileID/BlockFileID for lower level- $(l-1)$ for Number Type range given in its Columns 1 and 2.

Few lines of the very first level-1 meta data file of author's database for prime numbers looks like follows :

Figure 2 : Model Meta Data File

This file is hosted on Google cloud on URL : "https://docs.google.com/spreadsheets/d/1A6gyaxrpG-NbaLVZMpvXOJK1bfTmUhiZc_PN_0O2Is".

Level- l meta data files, whose range is covered by their level- $(l+1)$ metadata file, are present in common folder having the name format as “Number Type (File Info & IDs) - Serial Number Range”.

VII. NAVIGATION

The viewer can navigate through all covered number types reaching any desired n^{th} number type in generated range by starting with highest level meta data file and going recursively to lower level meta data files till the main data file link is obtained in respective range.

Simple computer programs or scripts are made available by the author to do this navigation programmatically that allow to directly get n^{th} number type number or numbers in any specified range within the database limit. They are also kept open in public domain for use by anybody interested.

VIII. FEATURES AND FUTURE SCOPE

A new robust database model has been proposed here for serially storing specific number types. Its actual implementation is worked out on a gigantic (or terantic!) scale till 1 trillion prime numbers by the author.

This model has some key features, which make it really robust for employing to store databases for numbers whose much information is not explored yet.

The data is stored in human readable format. This gives the direct manual access to numbers of respective number type.

As all successive numbers of specific number type are available in the database, it allows going through all of them for various purposes like gap analysis, occurrence density etc in desired ranges by just crawling through them.

The database is prescribed to be hosted on internet clouds, making it available on all platforms across the world all the time.

Platform independent API's are available for fetching n^{th} member of respective number type or all members in desired range for this kind of database programmatically.

It is expected that this database model proves a better candidate to be employed for serially storing all such

number types whose n^{th} member cannot be reached with efficient methods and requires serial storage.

IX. ACKNOWLEDGEMENT

The author is thankful to the anonymous referee of this paper. The author wishes to put on record the extensive use of computational resources like computer systems, software, internet facility of host institute, which made the generation and availability of the huge model database for Prime Numbers possible. The author is also thankful to Google LLC for hosting this valuable database spanning into few terabytes.

X. APPEAL

Although initially the author has maintained this database on Google cloud at his personal costs, it is expected, and the author appeals, that Google LLC (the courtesy has already been extended to Google in all places of the Database) maintains this database, which generated and authenticated by the author, free of charge - as it is a huge and precious resource for all mathematicians and computers scientists working in different areas like number theory, a host of related unsolved conjectures in related domains, encryption etc. Any other interested cloud service provider is also welcome to contribute in this research initiative. In fact, Google LLC and other cloud service providing platforms are urged to freely host all such important databases which may be proposed to be generated by any researcher in future, in public domain for the noble purpose of a new term being coined by the author : Corporate Research Responsibility (CRR)!

REFERENCES

1. N.J.A.Sloane, 1964, Webpage “The On-Line Encyclopedia of Integer Sequences”, “<https://oeis.org/>”
2. Conway J. H. and Norton S. P., 1979, “Monstrous Moonshine”, Bull. London Math. Soc. 11, 308-339.
3. Ribenboim, Paulo, 1996, The New Book of Prime Number Records (3rd ed.), New York: Springer.
4. S. Yates, 1982, Repunits and repetends, Star Publishing Co., Inc., Boynton Beach, Florida.
5. Ball, Keith M, 2003, Strange Curves, Counting Rabbits, and Other Mathematical Explorations, Princeton University Press, Princeton, NJ.
6. Finch, S. R., 2003, Mathematical Constants, Cambridge University Press, Cambridge, England.
7. Webpage “Prime I.T.”, “http://compoasso.free.fr/primelistweb/page/prime/liste_online_en.php”
8. Andrew Booker, Webpage “PrimePage Primes: The Nth Prime Page”, “<https://t5k.org/nthprime/>”