RA

stichting
mathematisch
centrum

MC

REKENAFDELING

RA

MR 133/72 JULY

JACK ALANEN
EMPIRICAL STUDY OF ALIQUOT SERIES

2e boerhaavestraat 49 amsterdam

Printed at the Mathematical Centre, 49, 2e Boerhaavestraat, Amsterdam.

The Mathematical Centre, founded the 11-th of February 1946, is a non-profit institution aiming at the promotion of pure mathematics and its applications. It is sponsored by the Netherlands Government through the Netherlands Organization for the Advancement of Pure Research (Z.W.O), by the Municipality of Amsterdam, by the University of Amsterdam, by the Free University at Amsterdam, and by industries.

Summary

Techniques from algorithmic analysis and empirical statistics are used to efficiently analyze the computational problem of aliquot series in number theory.

After introducing notation, definitions, and the history of aliquot series, the methodology and main findings in this thesis research are summarized.

Several properties of the function s (the sum of the aliquot divisors) are next given. These include: recurrence relations for evaluating s ; upper and lower bounds on x in $s(x) = n$; conditions for determining the parity of $s(x)$ relative to x ; upper bounds, an asymptotic formula, and the mean value for $s(x)/x$.

Then the oriented graph generated by s is investigated. This leads to concepts such as untouchable number (an n with no solutions to $s(x) = n$), clan (a finite generalized cycle), and Goldbach solutions, and it provides graph theoretic interpretations to perfect numbers, unbounded aliquot series, and other number theory notions associated with s .

Algorithms for solving $s(x) = n$ and searching for sociable numbers are next specified and analyzed. Finally, the results of programming these algorithms on a digital computer are presented as empirical statistics, and interpreted in the form of computed results and conjectures.

Acknowledgements

This report consists of my dissertation presented for the degree of Doctor of Philosophy at Yale University.

I want to thank five people for their help in the preparation of this thesis. First, there is the late Professor Oystein Ore who suggested the problem and guided the early work. I am very grateful to the late Professor L.J. Savage, my thesis advisor and finest teacher, who contributed much of his time and many valuable suggestions. I am also grateful to Professors F.J. Anscombe, D.E. Knuth, and A. van Wijngaarden, who reviewed my work at several stages and suggested significant improvements in the written thesis.

Acknowledgement is made to Yale University and the Mathematical Centre, Amsterdam, for the use of digital computers.

Table of Contents

Index of Tables

Index of Tables (continued)

1. Introduction

Problems in computational number theory are deceptive. On the one hand, they are often simply stated so that even an amateur can have success in computing a solution. Consider the example of amicable numbers: The smallest amicable pair is 220 and 284 because each is equal to the sum of the proper divisors of the other. Although amicable numbers were known to the Pythagoreans, it was 16-year-old B. Nicolo I. Paganini who in 1867 startled mathematicians by discovering, probably by trial and error, the second lowest amicable pair, 1184 and 1210 . (Ore 1948)*

On the other hand, such computational problems can easily lead to analyses which require the most advanced techniques in mathematics, probability, and statistics. For example, the existence of an odd perfect number (an odd number that equals the sum of its proper divisors) remains one of the celebrated unsolved problems in number theory. See (McCarthy 1957) for a summary of the many requirements an odd perfect must satisfy.

Now that digital computers are readily available for solving number theory problems, there is this same deceptiveness. In the one case, a brute force application of the machine by an amateur can produce significant output. The straight-forward tabulation of all the amicable pairs below a million is such a case. See (Alanen, Ore, and Stemple 1967) for details; all that is required is a simple factorization subroutine and about one hour of IBM 7090 machine time.

---

* The flexible system of making references to the Bibliography by such expressions as "(Ore 1948)", "by Borho (1968)" or "(Knuth 1968, p.316)" is familiar and self-explanatory.

At the other extreme, only the most "efficient" (well-planned, elegant, optimal, ingenious, etc.) analysis of the problem will allow a carefully programmed computer to make progress toward solutions. Using a computer in such a fashion, Muskat (1966) has proved that any odd perfect number must be divisible by a prime power greater than 10^{12}. He enlisted computers both to obtain prime factorizations and to check the accuracy and completeness of the lengthy proof.

I assert that an "efficient" analysis must employ techniques in algorithmic analysis (Knuth 1971) and empirical statistics. When a computer algorithm for solving a problem has been proposed, an analysis of the algorithm investigates the two questions: 1. Does the algorithm work? 2. Is the algorithm any good? A correctness proof is used to answer the first question. A program correctness proof does <u>not</u> consist of testing the program with representative input data and checking the resultant output. <u>Nor</u> is it reading a program closely and then announcing that it works. As Dijkstra says (in Buxton and Randall 1970): "Testing shows the presence, not the absence of bugs." By correctness proof we mean a rigorous mathematical proof which verifies that a program is in fact correct.

To answer the second question, a definition of what constitutes optimal performance must be decided upon. If computer memory is scarce, the algorithm will be good when a storage analysis shows that the program and intermediate results fit into memory. If running time is limited, the algorithm will be good when a frequency analysis shows that each computational step is performed a reasonable number of times. Other measures of performance, such as minimizing factorizations or maintaining a desired accuracy, can be explored.

Analysis of an algorithm will often lead to the construction of an improved algorithm. Such analyses are, in general, very difficult. But to demand, seek, and prefer correctness and computational efficiency in an algorithm can yield significant savings in both computer and programming time. Moreover the solution of a problem may actually be impossible before development of an optimal algorithm. For example, to determine that a thirty-digit integer n is prime by successively dividing it by $2, 3, \ldots, \sqrt{n}$ is impractical on a contemporary computer; yet efficient algorithms for proving the primality of such an n in a few seconds of computer time do exist (Knuth 1969).

Descriptive statistics is the second source of techniques for efficient analysis of problems in computational number theory. This sometimes discredited statistical activity helps to arrange and condense complicated sets of numbers in ways that allow you to form opinions and reach decisions. For getting insight or understanding or bright ideas, Savage (1968) encouraged the once cardinal sin of fooling around with the data. There should be increased interest in, and respect for, looking upon the data with affection and curiosity, or as Savage said, "really fooling around with the data to see whether, looked at this way, or the other way, it seems to spell 'Merry Christmas'."

The author undertook to study aliquot series in order to support his assertions that algorithmic analysis should be used to be careful and to lay theoretical groundwork before computer experiments are attempted; and that empirical statistics should be used to form extrapolatory conjectures, empirical theorems, and other inferences from the computed results. Some notation, definitions, and history of aliquot series follow.

Notation and Definitions. By the aliquot divisors* of a number are meant the divisors, including unity, which are less than the number. Let $s(n)$ denote the sum of the aliquot divisors of the nonnegative integer n. Define $s(0) = s(1) = 0$. A series of numbers $n, s(n), s^2(n), \ldots$, where the exponent denotes iteration, is called an aliquot series with leader n. Writing $n_0 = n$ and $n_k = s^k(n)$ for the terms in this series, such series can be typed in one of three ways:

(1.1) the series is purely periodic with proper period k, that is

$$n, n_1, n_2, \ldots, n_{k-1} \text{ are distinct and } n_k = n .$$

Perfect numbers correspond to the case $k = 1$ and hence satisfy $s(n) = n$. For example, $6 = 1 + 2 + 3$ is perfect and we consider 0 perfect according to the definition $s(0) = 0$. Amicable pairs (n, n_1) and crowds (n, n_1, n_2) correspond to $k = 2$ and $k = 3$, respectively. In general, the k distinct members of the series (1.1) are called sociable numbers of index k.

(1.2) the series is ultimately, but not initially, periodic. For example, the series with leader 562 leads to the smallest amicable pair (220, 284) since $s(562) = 284$. Furthermore, a series like 14, 10, 8, 7, 1, 0, 0, ... which contains a prime p always ends with zeroes since $s(p) = 1$,

* Terms that may be new to the reader are italicized (underlined) while terms introduced here but not of general use appear in bold face (wavy underline).

$s(1) = 0$, and $s(0) = 0$.

(1.3) the series is unbounded $(\lim_{k\to\infty} n_k = \infty)$.

It is not known whether this possibility is realized. The smallest n which could be the leader of an unbounded series is 276 , and then n_{348} has 31 digits as calculated by the D.H. Lehmers (personal note, February 1972).

History. Perfect and amicable numbers have been studied for centuries, so a history of their exploration using digital computers will be emphasized here. Euclid proved that the formula $2^{n-1}(2^n-1)$ always gives an even perfect number if the parenthetical expression is a prime. Two thousand years later, Euler proved that this formula gives all the even perfects. Primes of the form 2^n-1 are called Mersenne primes and the twelfth Mersenne prime, $2^{127}-1$, discovered by E. Lucas in 1876 is the largest to have been found without the aid of modern computers (Gardner 1968). The 23 known perfects* and their corresponding Mersenne primes are listed by Gardner. The last perfect - which has 6,751 digits - was discovered in 1963 when a computer at the University of Illinois determined the 23rd Mersenne prime, $2^{11213}-1$.

Amicable numbers were known to the Pythagoreans and numerous rules for constructing certain types of amicable pairs have been published (see Lee's 1969 history) . Exhaustive computer searches have recently enumerated all the amicable pairs less than 100000000 as follows:

---

*) The 24th even perfect number, $2^{19936}(2^{19937}-1)$, was recently computed (Tuckerman 1971).

Interval	Year	All amicable numbers in this interval published by
$(0, 10^5]$	1967	Rolf
$(10^5, 10^6]$	1967	Alanen, Ore, and Stemple
$(10^6, 10^7]$	1968	Bratley and McKay
$(10^7, 10^8]$	1970	Cohen

Sociable numbers and aliquot series are obvious generalizations of perfects and amicables. Two sociable series, one of index 5 with leader 12496 and the other of index 28 with leader 14316, were announced by Poulet (1918). While systematically enumerating the amicable pairs, the above authors conducted limited computer searches for crowds and other sociables of higher index. In the interval $(10^6, 6.10^7)$, Cohen's program outputted nine sociable series of index 4. Borho (1969) had published one of these series, $s^4(28158165) = 28158165$, but lacked machine time to fully implement his theoretical requirements on sociables of index 3 and 4. A condensed summary of the additional sociable series, with certain lesser numbers and indices, whose existence has been denied by computer trials follows:

(Alanen, Ore, and Stemple) Crowds with leader $n < 10^6$. Odd-even amicable pairs with the odd number < 3469563409.

(Borho) Sociables $n, n_1, \ldots, n_{k-1}$ with $n < 10^5$ and $2 < k < 10$.

(Cohen) Sociable series of index 10 or less, of which the lesser number is smaller than 6.10^7.

Computer experiments for seeking new sociable numbers and for tabulating aliquot series are currently being conducted by many scientists, so that it is difficult to keep up with very recent discoveries and results. For example, the D.H. Lehmers (personal note, February 1972) are daily pushing the series with leader 276 forward to determine if it is ultimately periodic. Also, R. David has recently reported (personal note, January 1972) his discovery of two new sociable series of index 4 . Their leaders are 209524210 and 330003580 , but details of their computation are unknown to me. For reference, Table 1.1 lists the thirteen known sociable series and their factorizations.

A tabulation of the s function was given by Dickson (1913), and Poulet (1929) computed several long aliquot series until a term increased beyond his practical power of calculation. Both of these authors comitted numerous errors and were limited by the necessity of performing calculations by hand. The most recent work appears to be a table computed on Olivetti - Underwood Programma 101 machines, of all aliquot series with leader $n < 10000$ (Guy and Selfridge 1971).

For fixed n , consider solutions to the equation $s(x) = n$ and denote the total number of these solutions by $d(n)$. Clearly $d(0) = 2$, $d(1) = \infty$, and $d(2) = 0$ because $s(x) = 0$ has only the solutions $x = 0$ and $x = 1$; $s(p) = 1$ for every prime p ; and $s(x) = 2$ is impossible. When $d(n) = 0$, I call n *untouchable*. If $x = n$ is the only solution to the equation $s(x) = n$, then n is a *hermit*; 28 is a hermit. Every hermit is a perfect number, but not conversely since, for instance, $s(25) = s(6) = 6$.

The first few untouchables were given by Dickson (1913), and Poulet (1929) further listed a few small solutions to $d(x) = n$

for $0 \leq n \leq 3$.

Next, a summary of the method of analysis and results, described in detail throughout Section 2-7, will be given.

Section 2 derives several properties of the function s . First recurrence relations for evaluating s are presented. Given a factor of n , these relations permit calculation of $s(n)$ in terms of this factor. These recurrence relations are later used heavily in proofs and in the construction of efficient algorithms.

Complete conditions for determining the parity of $s(x)$ relative to x are next specified. For example, an odd number has even s value only when it is a perfect square. The fact is that changes in the parity of aliquot series terms are related to whether or not a perfect square term occurs.

Upper and lower bounds on x in $s(x) = n$ are deduced. The largest value possible for x equals $(n-1)^2$ when $n > 1$ is fixed; this happens if $n-1$ is prime. The smallest value possible for $s(x)$ equals $x/2$ when x is even; equality happens iff $x = 2$.

Upper bounds for the ratio $s(x)/x$ are established and compared (Table 2.1). An asymptotic formula is also given. These results are all functions of $\omega(x)$, the number of distinct prime factors of x . Since "round" numbers (numbers with a considerable number of comparatively small factors) are rare, the result of computing $s(x)$ will, it turns out, rarely exceed $5x$.

Lastly, the mean value of the ratio $s(x)/x$ is displayed as $\pi^2/6-1$, or about 0.645. Hence n_{k+1} is typically about 65% of n_k . This suggests that, on the average, aliquot series eventually terminate.

These properties of s derived in Section 2 provide interesting and useful results independent of any computer computations.

Moreover, a little bit of theoretical work before using the machine can assist in the construction of more efficient and productive computer programs.

Section 3 applies well-known graph theory (Ore 1965) to characterize the oriented graph generated by s. This leads to such concepts as untouchable number, clan, and Goldbach solution. And it provides graph theoretical interpretations to perfect numbers, unbounded aliquot series, and other number theory notions associated with s. In Figure 1.2 appears part of the generalized cycle which contains the perfect number 8128. Further such graphs have been drawn by (Guy and Selfridge 1971). The results in Section 3 are a theoretical characterization of these graphs rather than a partial empirical tabulation. For example, it is proved (Theorem 7) that there exist an infinite number of both even and odd numbers which have edges leading into them (i.e., they are touchable).

When solutions to the equation $s(x) = n$ (for fixed odd $n > 1$) are investigated, solutions composed of the product of two distinct primes frequently obtain. These are named Goldbach solutions because Goldbach conjectured that every even integer can be written as the sum of two primes. The truth of a slightly strengthened Goldbach conjecture, which seems abundantly true empirically, implies that odd untouchable numbers (excepting 5) do not exist.

No finite generalized cycles (clans) of s appear to be known, besides the singular hermit 28. Because of the result (Theorem 6) on Goldbach solutions, a guide in searching for clans is to eliminate series with odd numbers from consideration.

Section 4 explores problems in solving the equation $s(x) = n$ for fixed $n > 1$. The straightforward procedure (enumerate $s(x)$

for all $x \leq (n-1)^2$, as based on Theorem 4) to solve this equation for n about 5000 requires around 25 million factorizations. Better algorithms are thus required for large n . Several efficient computer algorithms are constructed, proved correct, and further analyzed in this Section. They are based upon building and traversing a certain tree structure, called the aliquot tree for n , which contains all solutions to $s(x) = n$ among its nodes. No numbers are factored by these efficient algorithms and for $n = 5000$ they involve fewer than 250000 "simple" computational steps.

Refinements to the algorithms in Section 4 are possible if Goldbach solutions to $s(x) = n$ are either not required or else are found as a special project using another fast computer method which is described. Theoretical results (Theorems 8 and 9) are generated to support the analyses (especially the correctness proofs) of these algorithms.

Important and interesting features of the algorithms were brought out during their analysis. In particular, the discipline of proof accrued the advantages:

1. Provided a systematic search for errors.
2. Gave sufficient reasons why the algorithm was correct.
3. Led to ways by which the algorithm was spectacularly improved.
4. Made explicit the assumptions on which correctness rested.

Hence an attempt to satisfy yourself as to the correctness of an algorithm should be the first and most basic part of the analysis of any algorithm.

Section 5 looks into algorithms for the exhaustive systematic determination of sociable series. The usual approach to detect sociables is to examine each aliquot series $i, s(i), s^2(i), \ldots$

for $i = 0, 1, 2, \ldots, n$ until a term exceeds some large number N or until a term equals some preceding term (in which case a sociable series has been captured). This is Algorithm E .

A refinement of this straightforward approach is to keep track of the series terms which have already been examined. Thus when $N = n = 284$, the amicable pair (284, 220) would not be detected after (220, 284) is found. This is Algorithm H .

Because Algorithm R can be used to generate s values efficiently (that is, without factoring numbers), a faster method for detecting sociables is to store these s values in a table and then traverse the table systematically looking for sociables. This is Algorithm D .

Comparisons are made between Algorithm E , H , and D . Table 5.1 summarizes these storage and factorization frequency comparisons for the "best" and "worst" cases. All three algorithms are lacking when n exceeds a million.

Instead of systematically exhausting leader possibilities and computing all of their series terms up to some large value, restricting conditions can be placed on the leader and/or their series terms, so that the total number of possibilities is reduced while the probability of finding a sociable series is not reduced significantly. Section 6 contains such procedures based upon heuristic arguments and empirical observations on aliquot series.

Section 6 sets forth the results of computer experiments as empirical statistics, and interprets them in the form of computed results and conjectures. It begins with a description of the tables computed and how they were programmed.

Statistics based on the aliquot series with leaders below 1000 revealed seven distinct series which may be unbounded. One of these, the series with leader 276 , is conjectured to extend to over 448 terms. While examining these long series, H. te Riele (1972) observed that perfect numbers can appear as factors of series terms and when they do, they seem to remain as factors in succeeding terms. This suggested examining the series with leader Pq , where P is a perfect number and q is a prime that is relatively prime to P . For $P = 2^6.127$ and $q = 3$ the first 49 terms of this series are displayed in Table 1.3. Using Theorem 1 te Riele was then able to prove that the series with leader $3P$, where P is the 24th perfect number $2^{19936}(2^{19937}-1)$ which has 12003 digits, extends to over 5000 strictly increasing even terms. Hence Table 1.3 gave the insight that leads to a theorem on long series lengths in aliquot series.

Other statistics on series termination are provided by Tables 6.8 and 6.9. These are based upon the series with leaders below 40000 . If we consider a series to be unbounded when a term n_k exceeds 10^{10} , then 14% of these series were unbounded. A majority of 84% lead into prime numbers so that $n_k = 0$. The remaining 2% "bump" into sociable series. Poulet's two sociable series terminated numerous (54 or 0.1%) series considering the scarcity of sociables. All posibilities seem to occur: Large terms only after many terms; termination after many terms; series which remain small for many terms; series which increase rapidly.

A systematic search for new sociable series was conducted by implementing Algorithms H and D . Computed result 2 states that no further sociable series exist whose terms are below 200000 . Conjecture 3 argues that Poulet discovered his two sociable series by a systematic hand-calculation of those 901 aliquot series whose leader is a round number below 10000 . A round number possesses six or more prime factors.

Reasons why sociable numbers usually contain round numbers are given. The known perfect numbers are 87% round. Of the amicables below 10^8, 89% have at least one round number. And 85% of the known sociables with index over two contain round numbers. Based upon these observations, a computer search was conducted, unsuccessfully, for sociables with leader above the 6.10^7 tried by Cohen (1970). See the program in Figure 6.12. It should be noted that David's sociable series with leader $2^2.5.16500179$ (Table 1.1) would have been discovered by this program when larger values of q were taken. Further understanding of this roundness property among sociables and additional computer searching based upon it are called for.

A list of the 570 untouchable numbers below 5000 was computed (Table 6.3). Empirical properties of these untouchables are examined. By extrapolation it appears there are an infinity of untouchable numbers (Conjecture 4). A significance test suggests that among even numbers, being untouchable and being the double of a prime are <u>not</u> independent events.

Related to $d(n)$, the number of solutions to $s(x) = n$, is the number of "Goldbach decompositions" which has been studied by Stein and Stein (1965). This leads to the conjecture that $d(2n+1)$ is unbounded for large n. A related conjecture is that the equation $d(n) = k$, for fixed k, has at least one odd solution n. Refer to Tables 6.1, 6.2, and 6.11 for empirical tabulations of these solutions.

Much data on $d(n)$ can be found in Tables 6.1 and 6.2, which tabulate the solutions of $s(x) = n$ for n up to 500.

<u>Section 7</u> specifies the algorithms mentioned in Sections 4 and 5. An effort is made to prove that each computer procedure is un-

ambiguously specified, does terminate, has well-defined input and output, and can be performed in a reasonable number of steps. Correctness proofs for nontrivial program sections are outlined. A study of the properties of the algorithms is attempted; for example, a frequency analysis (how many times each step of the algorithm is likely to be executed) and a storage analysis (how much memory it is likely to need) are specified.

Had unlimited time and resources been available, plenty of further interesting things could have been done. Let me outline four topics, in particular, for future research in aliquot series: 1. Find an asymptotic empirical distribution for $s(n)$, suitably rescaled; 2. Develop a theory of untouchable numbers; 3. Conduct heuristic searches for sociable series; 4. Support the conjectures in Section 6 with additional evidence. Some elaborations on these four topics follow:

1. A splendid addition to aliquot series research would be to determine, possibly empirically, the asymptotic distribution of $s(n)$ for large values of n. If $s(n)$ is normalized by translation and scale parameters that are powers of n, then a limiting distribution might be obtainable empirically.

2. One could investigate whether untouchability behaves like Bernoulli trials with respect to even numbers. In particular, are the number of runs of even untouchables of various lengths what is expected under the hypothesis of independent trials? Similarly, half the distance from one untouchable to the next should be distributed in a geometric distribution; what is the observed phenomenon? All kinds of questions suggest themselves and each new answer would doubtless suggest more. Table 6.3 of

untouchables could be extended by suitably altering procedure R in Section 7. Perhaps an odd number deserves to be called almost untouchable if it is touched only by Goldbach solutions. Then comparison of the frequency of almost untouchables with that of truly untouchable evens could be made.

3. I have noted that almost every known sociable series includes at least one round number. Another empirical observation is that terms in a sociable series usually contain the same number of digits. Therefore, it seems desirable to base sociable series searches on such heuristics in order to reduce the search domain. It is again emphasized that a careful analysis should be attempted before time-consuming calculations are performed to find sociable numbers. Program traps should be set to yield something even if not the object of greatest interest. It requires skill and patience to anticipate possibilities so that a program will trap relevant information which seems secondary to the main output. Data analysis is clearly an area where you never know ahead of time everything of interest, and yet you must try to anticipate and accumulate.

4. Additional empirical evidence could be brought to bear on my conjectures in Section 6. For instance, according to the arguments for Conjectures 1 and 2, collapse of an even series occurs at a certain rate. How many situations are there in which collapse was not to be expected and did in fact not take place? Further numerical evidence will naturally suggest further conjectures.

In summary, techniques in both algorithmic analysis and empirical statistics have been applied to efficiently investigate aliquot series computing problems in number theory. Because the computer computations were carefully carried out, the empirical results are asserted to be mathematical facts. Further, they provide valuable data for the empirical side of number theory, which is as indispensable to discovering mathematical theorems as demonstration is to establishing them.

Table 1.1. The thirteen known sociable series, their factorizations, and their discoverers.

(Poulet 1918)

12496	$(2^4.11.71)$	14316	$(2^2.3.1193)$
14288	$(2^4.19.47)$	19116	$(2^2.3^4.59)$
15472	$(2^4.967)$	31704	$(2^3.3.1321)$
14536	$(2^3.23.79)$	47616	$(2^9.3.31)$
14264	$(2^3.1783)$	83328	$(2^7.3.7.31)$
		177792	$(2^7.3.463)$
		295488	$(2^6.3^5.19)$
		629072	$(2^4.39317)$
		589786	(2.294893)
		294896	$(2^4.7.2633)$
		358336	$(2^6.11.509)$
		418904	$(2^3.52363)$
		366556	$(2^2.91639)$
		274924	$(2^2.13.17.311)$
		275444	$(2^2.13.5297)$
		243760	$(2^4.5.11.277)$
		376736	$(2^5.61.193)$
		381028	$(2^2.95257)$
		285778	$(2.43.3323)$
		152990	$(2.5.15299)$
		122410	$(2.5.12241)$
		97946	(2.48973)
		48976	$(2^4.3061)$
		45946	(2.22973)
		22976	$(2^6.359)$
		22744	$(2^3.2843)$
		19916	$(2^2.13.383)$
		17716	$(2^2.43.103)$

Table 1.1. (Continued)

Borho (1969)

28158165	(3^3.5.7.83.359)
29902635	(3^3.5.7.31643)
30853845	(3^3.5.11.79.263)
29971755	(3^3.5.11.20183)

Cohen (1970)

1264460	(2^2.5.17.3719)	2115324	(2^2.3^2.67.877)
1547860	(2^2.5.193.401)	3317740	(2^2.5.165887)
1727636	(2^2.521.829)	3649556	(2^2.107.8527)
1305184	(2^5.40787)	2797612	(2^2.331.2113)
2784580	(2^2.5.29.4801)	4938136	(2^3.7.109.809)
3265940	(2^2.5.61.2677)	5753864	(2^3.23.31271)
3707572	(2^2.11.84263)	5504056	(2^3.17.40471)
3370604	(2^2.23.36637)	5423384	(2^3.53.12791)
7169104	(2^4.17.26357)	18048976	(2^4.11.102551)
7538660	(2^2.5.376933)	20100368	(2^4.919.1367)
8292568	(2^3.59.17569)	18914992	(2^4.37.89.359)
7520432	(2^4.127.3701)	19252208	(2^4.1203263)
18656380	(2^2.5.932819)	46722700	(2^2.5^2.47.9941)
20522060	(2^2.5.13.17.4643)	56833172	(2^2.11.52.24371)
28630036	(2^2.19.449.839)	53718220	(2^2.5.2685911)
24289964	(2^2.97.62603)	59090084	(2^2.43.343547)

David (personal note, January 1972)

209524210	(2.5.7.19.263.599)	330003580	(2^2.5.16500179)
246667790	(2.5.17.59.24593)	363003980	(2^2.5.18150199)
231439570	(2.5.19.23.211.251)	399304420	(2^2.5.1163.17167)
230143790	(2.5.17.499.2713)	440004764	(2^2.110001191)

Figure 1.2. A partial drawing of some nodes and edges in the generalized cycle which contains the perfect cycle 8128.

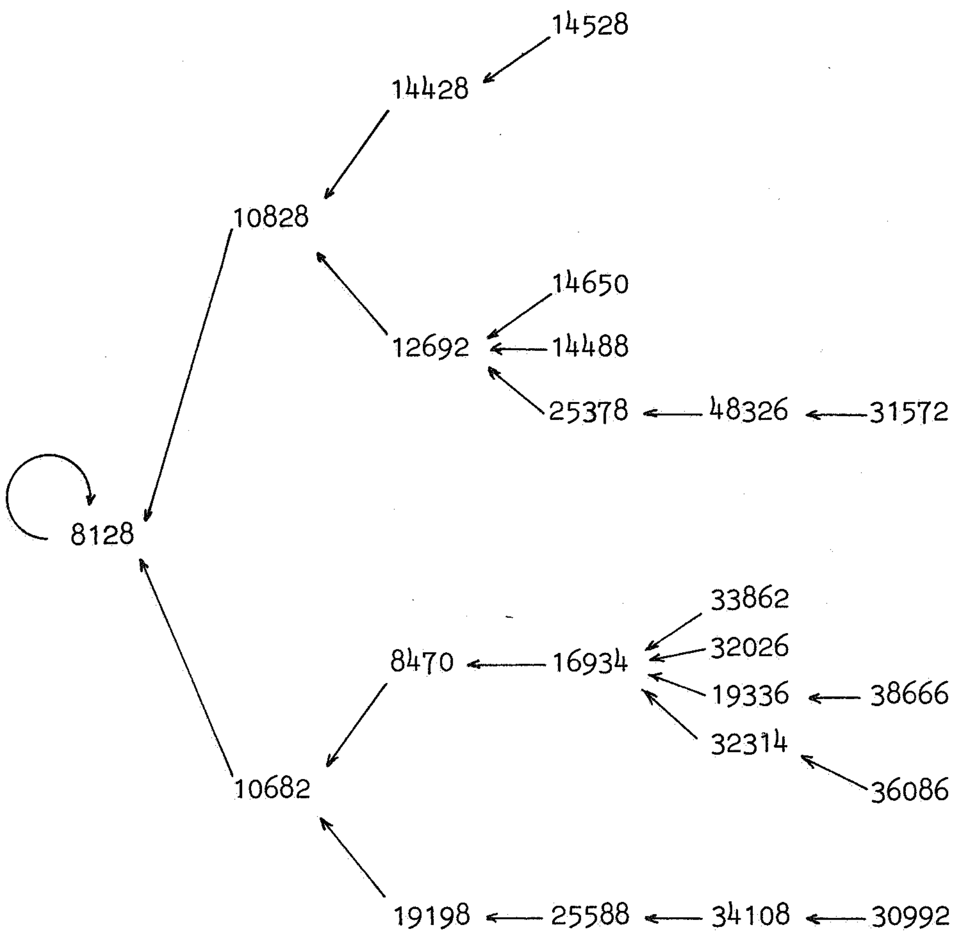


Table 1.3. The aliquot series with leader $n = 3P$, where $P = 2^6.127$ is a perfect number.

k	n_k	factorization of n_k
0	24384	2.2.2.2.2.2.3.127
1	40640	2.2.2.2.2.2.5.127
2	56896	2.2.2.2.2.2.7.127
3	73152	2.2.2.2.2.2.3.3.127
4	138176	2.2.2.2.2.2.17.127
5	154432	2.2.2.2.2.2.19.127
6	170688	2.2.2.2.2.2.3.7.127
7	349504	2.2.2.2.2.2.43.127
8	365760	2.2.2.2.2.2.3.3.5.127
9	902208	2.2.2.2.2.2.3.37.127
10	1568704	2.2.2.2.2.2.127.193
11	1584960	2.2.2.2.2.2.3.5.13.127
12	3877056	2.2.2.2.2.2.3.3.53.127
13	7534656	2.2.2.2.2.2.3.3.103.127
14	14443456	2.2.2.2.2.2.127.1777
15	14459712	2.2.2.2.2.2.3.127.593
16	24164544	2.2.2.2.2.2.3.127.991
17	40339264	2.2.2.2.2.2.7.127.709
18	51994816	2.2.2.2.2.2.127.6397
19	52011072	2.2.2.2.2.2.3.3.3.3.79.127
20	105347008	2.2.2.2.2.2.13.127.997
21	121781824	2.2.2.2.2.2.127.14983
22	121798080	2.2.2.2.2.2.3.3.3.3.5.37.127
23	326672448	2.2.2.2.2.2.3.127.13397
24	544519104	2.2.2.2.2.2.3.127.137.163
25	927104064	2.2.2.2.2.2.3.127.193.197
26	1570597824	2.2.2.2.2.2.3.41.127.1571
27	2722546752	2.2.2.2.2.2.3.127.111653
28	4537642944	2.2.2.2.2.2.3.71.127.2621
29	7737847872	2.2.2.2.2.2.3.127.317333
30	12896478144	2.2.2.2.2.2.3.3.11.11.31.47.127
31	30275296320	2.2.2.2.2.2.3.5.127.239.1039
32	67104646080	2.2.2.2.2.2.3.5.127.277.1987
33	148513897536	2.2.2.2.2.2.3.59.127.103231
34	254239556544	2.2.2.2.2.2.3.3.3.17.127.68147
35	543386442816	2.2.2.2.2.2.3.3.3.7.19.127.18617
36	1393600488384	2.2.2.2.2.2.3.13.13.31.127.10909
37	2760715246656	2.2.2.2.2.2.3.127.113218309
38	4601192142784	2.2.2.2.2.2.127.566091553
39	4601192159040	2.2.2.2.2.2.3.5.127.37739437
40	10122623140032	2.2.2.2.2.2.3.3.13.127.1231.8647
41	21399210114880	2.2.2.2.2.2.5.13.31.127.1306589
42	35693713768640	2.2.2.2.2.2.5.29.37.127.821.997
43	55522523041600	2.2.2.2.2.2.5.5.127.273240763
44	82173334605504	2.2.2.2.2.2.3.127.14159.238009
45	136971954712896	2.2.2.2.2.2.3.73.127.76949153
46	233290137724608	2.2.2.2.2.2.3.73.101.127.1297619
47	403583253133632	2.2.2.2.2.2.3.11.13.127.743.15577
48	862499296396992	

2. Properties of s

The following notations and conventions will be freely employed:

$\sigma(x) = s(x) + x$ is the sum of the divisors of x.

$\omega(x)$ denotes the total number of distinct prime factors of x.

$\Omega(x)$ equals the total number of prime factors of x.

p_i will be the i-th prime $(p_1 = 2,\ p_3 = 3,\ p_3 = 5,\ldots)$.

$q_1 < q_2 < q_3 < \ldots$ denote distinct primes.

$e, e_1, e_2, e_3, \ldots$ are (usually positive) integral exponents.

$i, i_1, i_2, i_3, \ldots$ are (usually positive) integral subscripts.

$\log y$ is the "natural" logarithm of y.

$\log_2 y$ is the base two logarithm of y.

In this Section are given properties of the s function which will be used later. Because $s(x) = \sigma(x) - x$, some of the proofs naturally rely on properties of the σ function. For example, a recurrence relation useful for computing s values is the expression in terms of s of the well-known multiplicativity of σ.

<u>Theorem</u> 1. If m and n are relatively prime, then

$$s(mn) = s(m)\,s(n) + ms(n) + ns(m)\ .$$

Proof:
$$\begin{aligned} s(mn) &= \sigma(mn) - mn \\ &= \sigma(m)\,\sigma(n) - mn \\ &= [s(m) + m]\,[s(n) + n] - mn \\ &= s(m)\,s(n) + ms(n) + ns(m)\ . \end{aligned}$$

Obviously, $s(p^e) = 1 + p + \ldots + p^{e-1} = (p^e-1)/(p-1)$ so that $s(p^{e+1}) = s(p^e) + p^e$. Therefore, we have

<u>Corollary</u> 1.1. If p is not a factor of m , then

$$s(mp^e) = s(m)\, s(p^{e+1}) + ms(p^e) .$$

<u>Corollary</u> 1.2. If p is not a factor of m , then

$$s(mp) = (1+p)\, s(m) + m .$$

Next we examine the parity of $s(x)$ when x is odd (even). Bouniakowsky (1848, p. 278) proved that for n odd, $\sigma(n)$ is even or odd according as n is not or is a square; for n even, $\sigma(n)$ is even if n is not a square or the double of a square, odd in the contrary case. Hence squares and their doubles are the only integers whose sums of divisors are odd. But for m odd and $e > 0$, it is evident that:

$s(m)$ even iff $\sigma(m)$ odd,

$s(2^e m)$ odd iff $\sigma(2^e m)$ odd.

Therefore, the parity of s is given by

<u>Theorem</u> 2. Suppose $e > 0$ and m is odd. Then

$s(m)$ even <u>iff</u> m = perfect square <u>iff</u> $s(2^e m)$ odd.

To express the next theorem conveniently, I introduce the conventions: $e_1 \geq e_2 \geq \ldots \geq e_k \geq 0$ denote integers; τ is any permutation $(i_1, i_2, \ldots, i_k)$ of $(1, 2, \ldots, k)$;

$$[\tau] = \prod_{\alpha=1}^{k} p_\alpha^{e_{\tau(\alpha)}} \quad \text{and} \quad \{\tau\} = \prod_{\alpha=1}^{k} q^{e_{\tau(\alpha)}} ,$$

where $q_1 < q_2 < \ldots < q_k$ denote primes and p_α is the α-th prime. With this notation, Corollary 1.1 becomes: If q is not a factor of $\{\tau\}$, then

$$s(\{\tau\}\, q^e) = s(\{\tau\})\, s(q^{e+1}) + \{\tau\}\, s(q^e) .$$

And the customary formula (Ore 1948, p.89) for computing values of s in terms of the prime factorization of its argument becomes:

$$(2.1) \qquad s(\{\tau\}) = \prod_{\alpha=1}^{k} \frac{q_\alpha^{e_{\tau(\alpha)}+1} - 1}{q_\alpha - 1} - \{\tau\} .$$

<u>Theorem</u> 3. $\min_\tau s\{\tau\}$ is attained when τ is the identity permutation or any permutation that leaves the e_i in non-increasing order, and only for such τ .

Prior to proving Theorem 3 and its Corollary, three relevant Lemmas will be developed.

<u>Lemma</u> 3.1. $\{(1\ 2)\} < \{(2\ 1)\}$ if $e_1 > e_2 \geq 0$.

Proof: Assume $e_1 > e_2 \geq 0$. Then $e_1 - e_2 \geq 1$ and $q_1 < q_2$ implies that

$$(q_1/q_2)^{e_1 - e_2} < 1 .$$

Thus $(q_1/q_2)^{e_1-e_2} q_2^{e_1} q_1^{e_2} = q_1^{e_1} q_2^{e_2} < q_2^{e_1} q_1^{e_2}$.

<u>Lemma</u> 3.2. $s\{(1\ 2)\} < s\{(2\ 1)\}$ if $e_1 > e_2 \geq 0$.

Proof: Assume $e_1 > e_2 \geq 0$ and $q_1 < q_2$. The assertion is that

$$s(q_1^{e_1} q_2^{e_2}) < s(q_1^{e_2} q_2^{e_1}) .$$

Clearly,

$$s\{(1\ 2)\} = \sigma(q_1^{e_2} q_2^{e_2}) + \sum_{i=e_2+1}^{e_1} \sum_{j=0}^{e_2} q_1^i q_2^j - q_1^{e_1} q_2^{e_2} .$$

$$s\{(2\ 1)\} = \sigma(q_1^{e_2} q_2^{e_2}) + \sum_{i=e_2+1}^{e_1} \sum_{j=0}^{e_2} q_2^i q_1^j - q_2^{e_1} q_1^{e_2}$$

where always $i > j \geq 0$. Applying Lemma 3.1 shows that each term in the double summation of $s\{(1\ 2)\}$ is strictly less than its corresponding term in $s\{(2\ 1)\}$. Hence the desired result.

<u>Lemma</u> 3.3. $s(m\, q_1^{e_1} q_2^{e_2}) < s(m\, q_1^{e_2} q_2^{e_1})$ if $e_1 > e_2 \geq 0$ and $\{m, q_1, q_2\}$ are relatively prime in pairs.

Proof: Under the assumption that m, q_1, and q_2 are relatively prime, Theorem 1 gives

$$\begin{aligned} s(m\, q_1^{e_1} q_2^{e_2}) - s(m\, q_1^{e_2} q_2^{e_1}) &= (s\{(1\ 2)\} - s\{(2\ 1)\})\, \sigma(m) \\ &\quad + (\{(1\ 2)\} - \{(2\ 1)\})\, s(m) \\ &< 0 , \end{aligned}$$

since both parenthesized terms are negative by Lemmas 3.1 and 3.2, assuming $e_1 > e_2 \geq 0$.

A proof of Theorem 3 based on the above three Lemmas follows.

Proof: Suppose $\{\tau\} = \prod_{\alpha=1}^{k} q_\alpha^{e_{\tau(\alpha)}}$. Take the first α if any such that

$$e_{\tau(\alpha)} < e_{\tau(\alpha+1)} ,$$

and interchange $e_{\tau(\alpha)}$ with $e_{\tau(\alpha+1)}$ so that

$$\tau'(\beta) = \begin{cases} \tau(\alpha) , & \text{if } \beta = \alpha+1 \\ \tau(\alpha+1) , & \text{if } \beta = \alpha \\ \tau(\beta) , & \text{otherwise} \end{cases}$$

$$\{\tau'\} = m\, q_\alpha^{e_{\tau(\alpha+1)}} q_{\alpha+1}^{e_{\tau(\alpha)}} , \quad \text{with} \quad m = \prod_{\substack{\beta=1 \\ \beta\neq\alpha,\alpha+1}}^{k} q_\beta^{e_{\tau(\beta)}} .$$

Because $q_\alpha < q_{\alpha+1}$ and $m, q_\alpha, q_{\alpha+1}$ are relatively prime, Lemma 3.3 yields

$$s\{\tau'\} < s\{\tau\} .$$

That is, the interchange of two adjacent exponents $e_{\tau(\alpha)}$ and $e_{\tau(\alpha+1)}$ in $\{\tau\}$ gives a smaller s value if $e_{\tau(\alpha)} < e_{\tau(\alpha+1)}$. For any τ , if an interchange of the form τ' above is possible, then the new s value is smaller. The only τ where this interchange is not possible satisfies

$$e_{\tau(\alpha)} \geq e_{\tau(\alpha+1)} \qquad \text{for} \quad \alpha = 1, 2, \ldots, k-1$$

and this is a monotone decreasing series

$$e_{\tau(1)} \geq e_{\tau(2)} \geq \cdots \geq e_{\tau(k)} .$$

Hence $e_{\tau(\alpha)} = e_\alpha$, or τ is the identity permutation, when $s\{\tau\}$ attains a minimum.

<u>Corollary</u> 3.1. $\min_\tau s[\tau]$ is attained when $\tau = (1\ 2\ \ldots\ k)$ or any permutation that leaves the e_i in nonincreasing order, and only for such τ.

In Section 4 we seek solutions x of $s(x) = n$. For example, $s(x) = 6$ has exactly two solutions $x = 6$ and $x = 25$. Now the following question arises: For fixed n, are there practical bounds on x such that $s(x) = n$? Answers are given by the next two theorems.

<u>Theorem</u> 4. $s(x) = n > 1$ implies $x \leq (n-1)^2$, with equality iff $x = p^2$.

Proof: Assume x is a solution to $s(x) = n > 1$. Let the prime factorization of x equal $\prod_{i=1}^{k} q_i^{e_i}$, for primes $q_1 < q_2 < \ldots < q_k$ and positive exponents e_i. Since $n > 1$, $x \neq q_1$. Thus q_1 and x/q_1 are the smallest and largest proper divisors > 1 of x, respectively. Now if $x = q_1^2$, then $s(x) = 1 + q_1 = n$, so that

$x = (n-1)^2$. Otherwise, $n = s(x) \geq 1 + q_1 + x/q_1$, which implies $q_1 < n-1$ and $x/q_1 < n-1$. Hence $x = q_1(x/q_1) < (n-1)^2$.

Note the upper bound $(n-1)^2$ is attained iff $n-1$ is prime; for example, $s(x) = 284$ only if $x \leq 283^2 = 80089$, with $x = 283^2$ a solution because 283 is prime. In case $x \neq p^2$ the bound in Theorem 4 can be improved to $x \leq (n-1)^2/4 - 1$, with equality iff $x = q_1 q_2$ and $q_2 = q_1+2$, Recall that $s(x) = 0$ iff $x = 0, 1$, whereas $s(x) = 1$ iff x is prime. The proof of Theorem 4 can be specialized to give:

<u>Corollary</u> 4.1. If the prime p divides x , then $x \leq ps(x)$, with equality iff $x = p$.

Proof: Let $x = pm$. If $m = 1$, then $s(x) = 1 = x/p$. Otherwise, $m > 1$ and then $s(x) \geq 1 + m > m = x/p$.

The next result gives upper bounds for $s(x)/x$ in terms of $\omega(x)$, the number of distinct prime factors of x .

<u>Theorem</u> 5. $s(x)/x < \omega(x)$ and $s(x)/x < 4\sqrt{\omega(x)/\pi} - 1$.

Proof: Let $x = \prod_{i=1}^{k} q_i^{e_i}$ be the prime factorization of x such that $q_1 < q_2 < \ldots < q_k$. Meissner (1903) noted that:

$$\frac{\sigma(x)}{x} < \prod_{i=1}^{k} \frac{q_i}{q_i-1} .$$

Since $\sigma(x) = s(x) + x$ and $q_i \geq p_i \geq i+1$ (for $i \geq 1$), this yields:

$$\frac{s(x)}{x} < \prod_{i=1}^{k} \frac{q_i}{q_i - 1} - 1$$

(2.2) $$\leq \prod_{i=1}^{k} \frac{p_i}{p_i - 1} - 1$$

(2.3) $$\leq \frac{(k+1)!}{k!} - 1 = k = \omega(x) .$$

Furthermore, $p_i \geq 2i-1$ for $i \geq 2$, so that:

$$\prod_{i=1}^{k} \frac{p_i}{p_i - 1} \leq 2 \prod_{i=1}^{k-1} \frac{2i+1}{2i} = \frac{2(2k-1)!}{2^{2k-2}(k-1)!\ (k-1)!}$$

$$= \frac{2k^2(2k)!}{(2k)2^{2k-2}\ (k!)^2} = \frac{4k}{2^{2k}} \binom{2k}{k} .$$

Using the double inequality (Feller 1957)

$$\sqrt{2\pi}\ n^{n+\frac{1}{2}}\ e^{-n+\frac{1}{12n+1}} < n! < \sqrt{2\pi}\ n^{n+\frac{1}{2}}\ e^{-n+\frac{1}{12n}} ,$$

we will overestimate $(2k)!$ and underestimate $k!$ to get

$$\binom{2k}{k} = \frac{(2k)!}{(k!)^2} < \frac{2^{2k}}{\sqrt{\pi k}} \exp\left(\frac{1}{24k} - \frac{2}{12k+1}\right) < \frac{2^{2k}}{\sqrt{\pi k}}$$

since the parenthesized exponent is clearly negative. Thus

(2.4) $$\frac{s(x)}{x} < 4\sqrt{\frac{k}{\pi}} - 1 .$$

An asymptotic formula for (2.2) follows from the two results:

$$\prod_{p \le x} \left(1 - \frac{1}{p}\right) \sim e^{-\gamma}/\log x \qquad \text{(Hardy and Wright, Merten's theorem),}$$

$$p_k \sim k \log k \qquad \text{(Hardy and Wright, Theorem 8).}$$

It is obviously

$$\text{(2.5)} \qquad \prod_{i=1}^{k} \frac{p_i}{p_i - 1} - 1 \sim e^{\gamma} \log k - 1 \quad \text{as } k \to \infty ,$$

where γ is Euler's constant $0.57721+$.

Since "round" numbers (Hardy and Wright) are extremely rare, the result of computing $s(x)$ will rarely exceed $5x$, by Theorem 5. (See Section 6 for justification of the definition: x is round iff $\Omega(x) \geq 6$, where $\Omega(x)$ equals the total number of prime factors of x .) When $x < 2.3.5\ldots p_{10} = 6469693230$, $\omega(x) < 10$; thus $9x$ is an upper bound on $s(x)$ for $x < 6469693230$, that is, for those numbers x with fewer than 10 prime factors. On the other hand, if, for example, $s(x) = 10^5$, then Theorem 4 guarantees that $x < 10^{10}$, so that $\omega(x) \leq 10$ which implies that $x > s(x)/\omega(x) \geq 10^4$.

A tabulation of the upper bounds (2.2), (2.3) and (2.4) on $s(x)/x$ and the asymptotic value (2.5) appears in Table 2.1.

We conclude this Section with some remarks on the behavior of $s(n)$ for large values of n . Hardy and Wright prove that $\sigma(n) = O(n \log \log n)$; that is, there exists a positive constant

K such that $\sigma(n) \leq K.n \log\log n$. Therefore, an upper bound of the same form holds for s: $s(n) = \sigma(n) - n = O(n \log\log n)$. The mean $M\{f(n)\}$ of a number theoretic function f is defined as the limit (if it exists)

$$M\{f(n)\} = \lim_{N\to\infty} \frac{1}{N} \sum_{n=1}^{N} f(n) .$$

Using the result (Hardy and Wright) that

$$M\{\frac{\sigma(n)}{n}\} = \pi^2/6$$

it is easy to see that

$$M\{\frac{s(n)}{n}\} = \lim_{N\to\infty} \frac{1}{N} \sum_{n=1}^{N} \frac{\sigma(n)-n}{n} = M\{\frac{\sigma(n)}{n}\} - 1$$

$$= \pi^2/6 - 1 = 0.6449+ .$$

Table 2.1. Upper bounds, derived from (2.2), (2.3), and (2.4) for $s(x)/x$, and the asymptotic formula (2.5).

$\omega(x)$	$\prod_{i=1}^{\omega(x)} \frac{p_i}{p_i-1} - 1$	$4\sqrt{\omega(x)/\pi} \doteq 1$	$e^{\gamma} \log \omega(x) - 1$
1	1	1.25676	-1
2	2	2.19154	0.23455
3	2.75	2.90882	0.95671
4	3.375	3.51352	1.46909
5	3.8125	4.04627	1.86653
6	4.21354	4.52791	2.19125
7	4.53939	4.97082	2.46581
8	4.84713	5.38308	2.70364
9	5.11291	5.77027	2.91342
10	5.33123	6.13650	3.10107
11	5.54227	6.48482	3.27083
12	5.72400	6.81764	3.42580
13	5.89210	7.13686	3.56836
14	6.05620	7.44402	3.70035
15	6.20959	7.74039	3.82323
30	7.71308	11.36077	5.05778
60	9.18962	16.48077	6.29232
120	10.64801	23.72155	7.52687
240	12.09158	33.96155	8.76141
480	13.51709	48.44310	9.99596
960	14.92599	68.92310	11.23051

3. Aliquot Graphs

The function s is studied in this Section from the graph-theory point of view. A reader who wants a more formal introduction to the definitions and results for the graph of a many-to-one correspondence of a set into itself will find them in Ore (1965).

Our directed graph $G = G(V)$ with vertex set $V = \{0, 1, 2, 3, \ldots\}$ has a single directed edge $(v, s(v))$ issuing from each vertex $v \in V$. Define $s(0) = s(1) = 0$, as always in this paper. An edge (v,v) is called a loop and loops correspond to perfect numbers.

Denote by $d(v)$ the number of incoming edges at a vertex v. Hence $d(v)$, called the in-degree of G at v, equals the number of edges having terminal vertex v. For example, $d(6) = 2$ since the only solutions of $s(x) = 6$ are $x = 6$ and $x = 25$. An untouchable number v has $d(v) = 0$ and is never a terminal vertex of an edge. The number of outgoing edges from any vertex always equals 1, for s is single-valued; s is not onto V because untouchable numbers, like 2 and 5, exist. The following theorem implies it is quite probable that every odd number except 5 has positive in-degree.

Theorem 6. If every even integer $n > 6$ is a sum of two distinct odd primes, then for every odd integer $v > 7$, $d(v) > 0$ and $s(x) = v$ for some odd solution $x > v$.

Proof: If $v \geq 9$ is odd, the hypothesis assures the existence of primes $q_1 > q_2 \geq 3$, so that $v-1 = q_1 + q_2$ and hence

$$s(q_1q_2) = 1 + q_1 + q_2 = v \ .$$

Obviously, $q_1q_2 - v = (q_1-1)(q_2-1) - 2 > 0$. Thus $d(v) > 0$ because $x = q_1q_2 > v$ satisfies $s(x) = v$.

Note that $(2,1), (4,3), (8,7) \in G$. Using this and Theorem 6, we have that every odd integer $v \neq 5$ has $d(v) > 0$ (assuming the hypothesis of Theorem 6 holds). The hypothesis of Theorem 6 is a strengthened form of the Goldbach conjecture and from the empirical point of view (Shen 1964; Stein and Stein 1965) seems abundantly true. Numbers which are the product of two distinct odd primes will be called *Goldbach solutions*. Experimental evidence on the number of Goldbach solutions, and hence on a lower bound for $d(v)$ when v is odd, is presented in Section 6.

Several elementary properties of the in-degree function are contained in the next theorem.

Theorem 7. (1) The only number with infinite in-degree is unity; that is

$$d(v) = \infty \quad \text{iff} \quad v = 1 \ .$$

(2) If the strengthened Goldbach conjecture (see the hypothesis of Theorem 6) is true, then

$$d(v) = 0 \ \text{implies} \ v = 5 \ \text{or} \ v \ \text{is even.}$$

(3) There exist an infinite number of touchable even numbers; that is,

$$d(v) > 0 \text{ for an infinity of even } v .$$

(4) There exist an infinite number of touchable odd numbers; that is,

$$d(v) > 0 \text{ for an infinity of odd } v .$$

Proof: (1) follows from the bound $d(v) < (v-1)^2$, when $v > 1$, of Theorem 4 and from the fact that $s(p) = 1$ for every prime p.

(2) is immediate from the remark after Theorem 6.

To show (3), let $p > 2$ be prime. Then $v = p + 1$ is even and $s(p^2) = v$. Since there are an infinity of odd primes, the result follows.

The odd numbers $v_i = 4 + p_{i+2}$ for $i \geq 1$ satisfy (4), because $s(3p_{i+2}) = 1 + 3 + p_{i+2} = v_i$ implies $d(v_i) > 0$.

Each vertex n defines a unique directed sequence of edges passing through the successive vertices

(3.1) $$n_0 = n , \quad n_1 = s(n) , \quad n_2 = s^2(n) , \ldots .$$

The smallest $k > 0$ such that $n_k = n$, if there is one, yields a finite cycle of length k passing through the vertices

(3.2) $$C = (n, n_1, n_2, \ldots, n_{k-1}) .$$

Loops (perfect numbers) correspond to cycles of length 1, amicable pairs make up the cycles of length 2, and cycles of length k constitute a series of sociable numbers of index k.

If, on the other hand, the vertices in (3.1) never repeat, then n is said to belong to the infinite cycle defined by (3.1). Infinite cycles correspond to unbounded aliquot series. An *infinite reverse cycle* is a directed sequence of edges passing through the infinity of distinct vertices $n_0, n_{-1}, n_{-2}, \ldots$ in the backward direction, where $s(n_{-i}) = n_{1-i}$ for $i \geq 1$. Furthermore, if a cycle is infinite in both directions

$$\ldots, n_{-2}, n_{-1}, n_0, n_1, n_2, \ldots$$

then it will be called a *two-way infinite cycle.*

There exists a decomposition of the vertex set

$$V = \sum_i V_i$$

into disjoint sets such that in each V_i all vertices are connected (ignoring edge direction) while no vertices belonging to two different sets are connected. It induces the direct decomposition

$$G = \sum_i G(V_i)$$

of the graph G into disjoint connected subgraphs $G_i = G(V_i)$ called the generalized cycles of s. Two important results (Ore 1962, Theorem 4.4.2 and 4.4.3) for the connected components G_i of G are:

1. Each generalized cycle contains at most a single finite cycle.

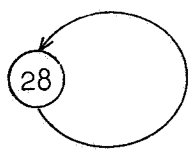
2. A finite generalized cycle always contains a finite cycle.

We shall now describe, by specializing the general results in Ore (1962, section 4.4), the form of the graph G of s'. Assume first that G_i is one of its generalized cycles containing the finite cycle C of (3.2). For each $v \in V_i$ not in C there is a smallest exponent $h > 0$ such that

$$s^h(v) = n_j \in C$$

and it defines a unique directed path of length h from v to n_j. Hence at each vertex n_j in C there will be attached a finite or infinite tree with the root n_j. In the case where the generalized cycle contains no finite cycle, it follows G_i is a tree with infinite cycles. (See Figures 3.1 and 3.2).

No finite generalized cycles (clans of G) are known to the author other then the singular hermit 28 :



Theorem 6 provides a guide in searching for clans; it implies a clan cannot contain an odd number. For clearly 1, 3, 5 and 7 are vertices in the infinite generalized cycle which contains all the primes, whereas every odd vertex $v > 7$ will (assuming the strengthened Goldbach hypothesis of Theorem 6) define an infinite tree with v as root and hence also belong to infinite generalized cycles.

Thus, if Goldbach's conjecture (slightly extended) holds, every odd $v \neq 5$ leads to at least one infinite reverse cycle. A generalized cycle which contains no finite cycle always defines an infinity of infinite cycles. But existence of infinite or two-way infinite cycles is an open question.

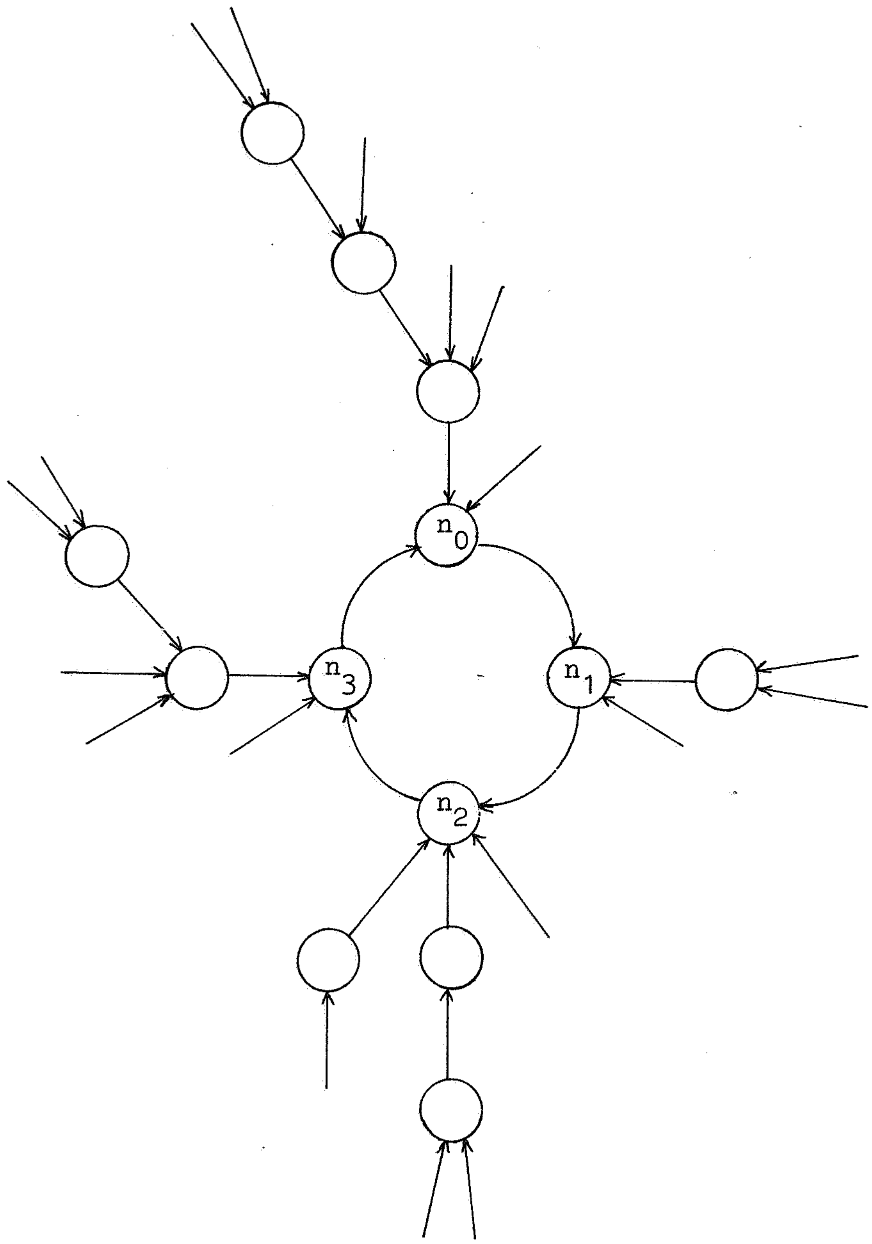
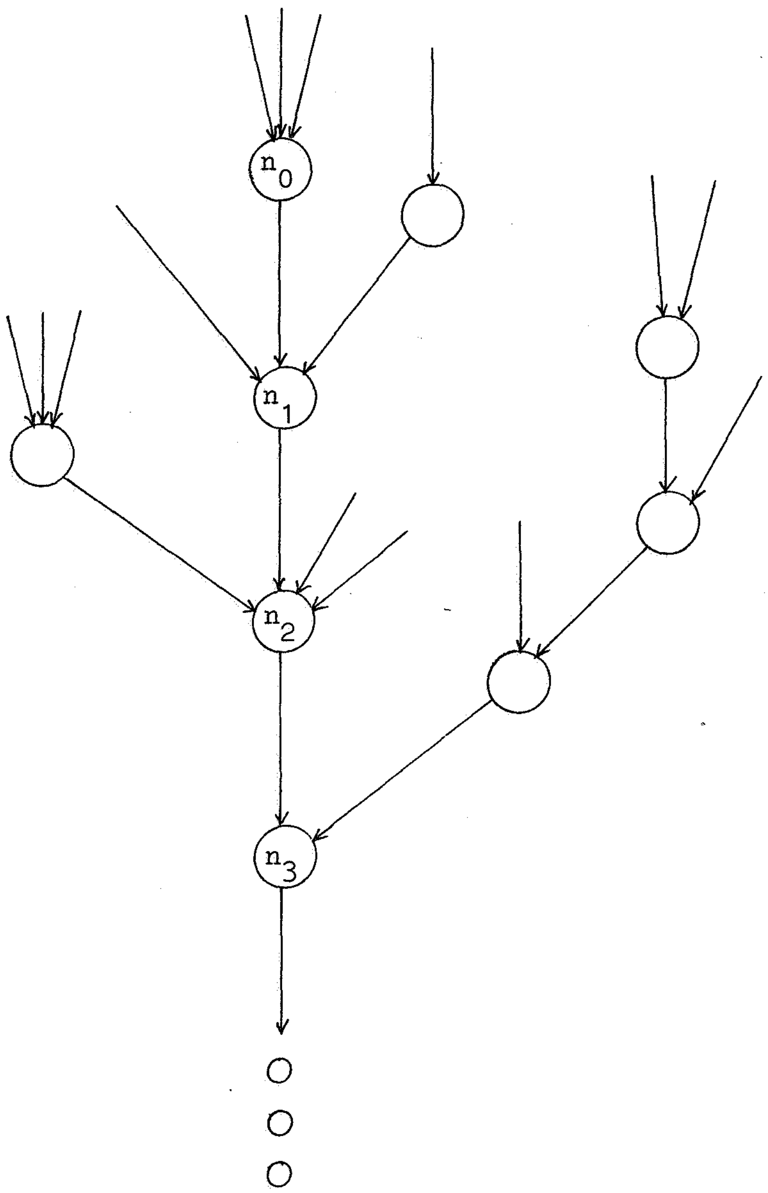


Figure 3.1. Generalized cycle with finite cycle.



<u>Figure</u> 3.2. Generalized cycle with no finite cycle.

4. Solving $s(x) = n$

To show that n is untouchable is to show that $s(x) = n$ has no solution x. To find which numbers have s-values equal to n is to find all the solutions of $s(x) = n$. Theorem 4 provides the dogged forthright approach to these problems; simply enumerate $s(x)$ for $x = 1, 2, 3, \ldots, (n-1)^2$. For $n = 13$ this requires factorization of $12^2 = 144$ numbers and evaluation of their s-values. Algorithm R of Section 7 computes only 46 s-values and does not factor any numbers; it uses the recurrence relations of Corollaries 1.1 and 1.2 to evaluate these 46 s-values efficiently.

Before Algorithm R is fully analysed we define the aliquot tree of $n > 1$, describe how to generate it, and give rules for traversing it. Roughly speaking, at level $k \geq 0$ of the aliquot tree of n are the numbers (arranged in a particular lexicographic order and excluding primes exceeding $n-1$) with s-values $\leq n$ and precisely k distinct prime factors. The root of the entire tree is 1 and it is the only node at level 0. Some nomenclature of Knuth (1968) for tree structures will be reviewed before aliquot trees are rigorously defined.

Let us define an (ordered) <u>tree</u> formally as a finite set T of one or more nodes (integers) such that

(a) There is one specially designated node called the <u>root</u> of the tree, $\mathrm{root}(T)$; and

(b) The remaining nodes (excluding the root) are partitioned into an ordered sequence of $m \geq 0$ sets $T_1, \ldots, T_m$, and each of these sets in turn is a tree.

The trees $T_1, \ldots, T_m$ are called the subtrees of the root T ; when $m \geq 2$ we call T_2 the "second subtree" of the root, etc. Every node of a tree is the root of some subtree and the number of such subtrees is called the degree of that node. A terminal node (leaf) has degree zero, whereas a nonterminal node is called a branch node. The level of a node with respect to T is defined thusly: The root has level 0, and other nodes have a level that is one higher than they have with respect to the subtree of the root, T_j , which contains them.

We hereafter always draw trees with the root at the top and leaves at the bottom. For descriptive terminology to talk about trees, each root is said to be the father of the roots of its subtrees, and the latter are said to be brothers, and they are sons of their father. Tree structure will be represented notationally by nested parentheses: A tree is represented by the information written in its root T, followed by the representation of the ordered subtrees $(T_1, \ldots, T_m)$ of T ; the representation of $(T_1, \ldots, T_m)$ is a parenthesized ordered list of the representations of its trees, separated by commas. For example, the tree in Figure 4.1 has representation

$$(4.1) \qquad 1(2(2.3,\ 2.5,\ 2.7),\ 2^2,\ 2^3,\ 3(3.5,\ 3.7),\ 3^2,\ 3^3,\ 5(5.7),\ 5^2,\ 7,\ 7^2,\ 11,\ 11^2)\ .$$

Note: The tree of (4.1) has 15 terminal nodes, six of them (2.3, 2.5, 2.7, 3.5, 3.7, and 5.7) at level 2 . Branch node 5 is the father of 5.7 , as well as the root of the seventh subtree of 1 . Leaves 2.3, 2.5, and 2.7 are brothers, all sons of 2 .

A sequence of trees is traversed in preorder when we visit its nodes as follows:

Preorder Traversal

(a) Visit the root of the first tree;
(b) Traverse the subtrees of the first tree (in preorder)
(c) Traverse the remaining trees (in preorder).

These tree recursive steps in which preorder traversal proceeds would visit the nodes of tree (4.1) in the sequence

$$1,\ 2,\ 2.3,\ 2.5,\ 2.7,\ 2^2,\ 2^3,\ 3,\ 3.5,\ 3.7,\ 3^2,\ 3^3,$$
$$5,\ 5.7,\ 5^2,\ 7,\ 7^2,\ 11,\ 11^2\ .$$

This is simply the representation (4.1) with the right parentheses removed and the left parentheses replaced by commas.

The search for all possible solutions of $s(x) = n$ for a given n is greatly simplified by constructing a certain tree $T[n]$, the nodes of which are (with unimportant omissions) the integers x for which $s(x) \leq n$ and having all the solutions of $s(x) = n$ among its leaves.

We are prepared to define the *aliquot tree*, $T[n]$, for $n > 1$ by giving rules for building the sons of an arbitrary father node. The root of $T[n]$ is always unity, that is, $\text{root}(T[n]) = 1$. Assume that

$$(4.2)\quad A_k = \prod_{i=1}^{k} p_{I_i}^{e_i} \quad \text{for } e_i > 0 \text{ and } 1 \leq I_1 < I_2 < \dots < I_k ,$$

is any node at level $k \geq 0$. Define $A_0 \equiv 1$. Then all the sons of A_k are the set of integers

$$(4.3)\quad B = \{A_k p_i^e :\ e > 0 \text{ and } i > I_k \text{ and } p_i < n \text{ and } s(A_k p_i^e) \leq n\}$$

and they are ordered as follows: The sons $A_k p_i^{e_1}$ and $A_k p_j^{e_2}$ are the roots of the i_1-th and i_2-th subtrees, respectively, if and only if (1) $i < j$ implies $i_1 < i_2$, and (2) $i = j$ and

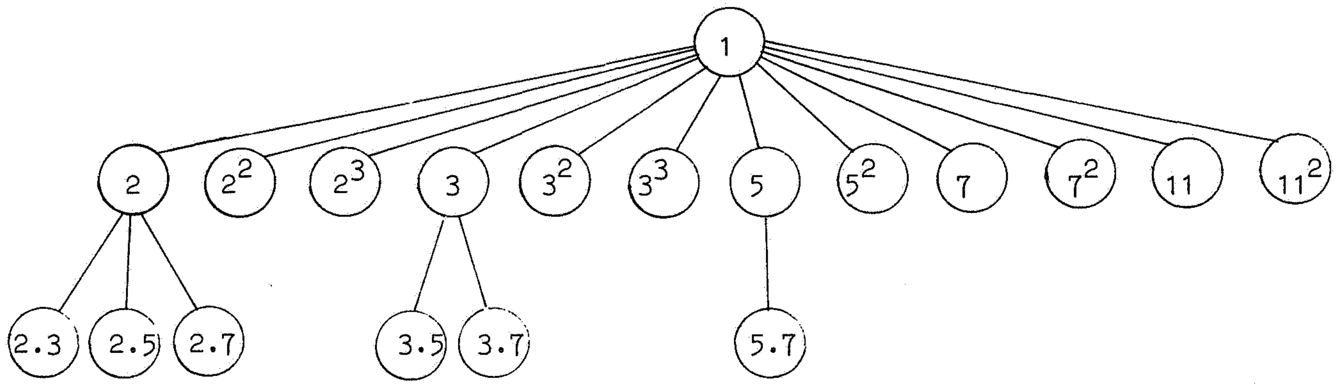


Figure 4.1. Aliquot tree, T[13] , for $n = 13$. In the nested parentheses representation,

$$T[13] = 1(2(2.3,\ 2.5,\ 2.7),\ 2^2,\ 2^3,\ 3(3.5,\ 3.7),\ 3^2,\ 3^3,\ 5(5.7),\ 5^2,\ 7,\ 7^2,\ 11,\ 11^2)\ .$$

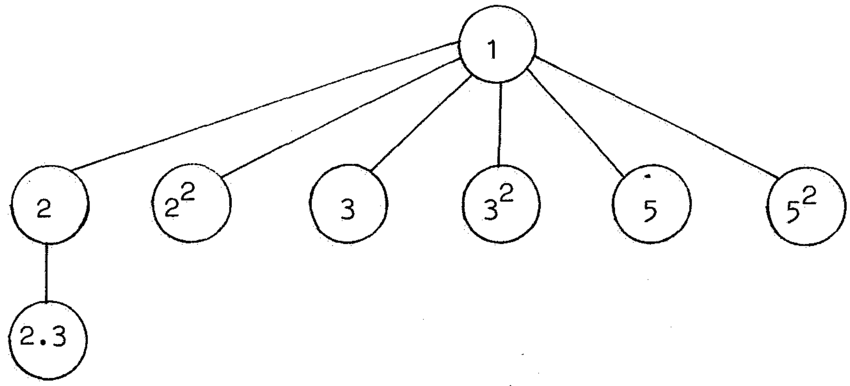


Figure 4.2. Aliquot tree, T[6] , for n = 6 . In nested parentheses representation,

$T[6] = 1(2(2.3), 2^2, 3, 3^2, 5, 5^2)$.

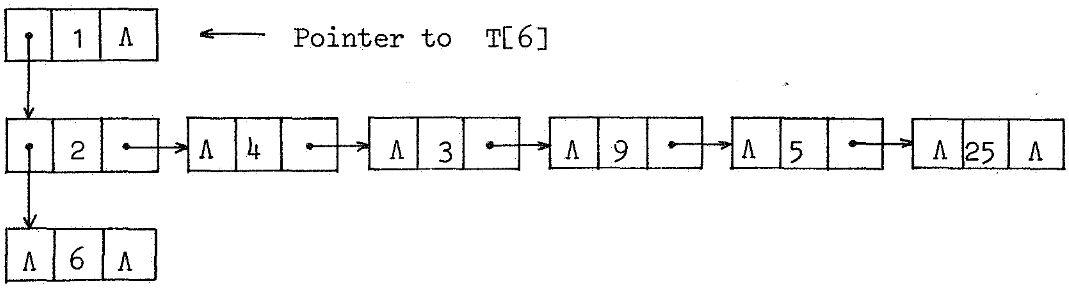


Figure 4.3. Picture of list representation for

T[6] = 1(2(6), 4, 3, 9, 5, 25) . Links are shown by arrows, except the null link Λ .

$e_1 < e_2$ implies $i_1 < i_2$.

Obviously, conditions (1) and (2) are sufficient to order the elements of B , so that the subtrees whose roots are the sons of A_k will also be ordered. This ordering also guarantees that certain sons of A_k will have ordered s values, because $e > 0$, $p < q$, and p,q not factors of A_k implies:

$$s(A_k p^e) = s(A_k p^e q^0) < s(A_k p^0 q^e) = s(A_k q^e) \text{ , by Theorem 3.}$$

$$s(A_k p^{e+1}) = S(A_k p^e) + A_k s(p^{e+1}) > s(A_k p^e) \text{ , by Corollary 1.1.}$$

Furthermore, the requirements that $s(A_k p_i^e) \leq n$ and $p_i < n$ ensure B is finite at each level. For not that $p_i \geq n$ implies $s(A_k p_i^e) \geq A_k + p_i^e > n$, except when $A_k = e = 1$. Since $A_k = 1$ iff $k = 0$, the requirement $p_i < n$ prevents all primes p_i $(i = 1, 2, \ldots)$ from being sons of the root 1 ; it implies $i \leq \pi(n-1)$, where the prime function $\pi(x)$ denotes the number of primes not exceeding x . Then excluding these prime nodes at level 1 , only a finite number of p_i^e will satisfy $1 < s(A_k p_i^e) \leq n$, namely at most $(n-1)^2$ by Theorem 4. And the terminal nodes of $T[n]$ cannot have level numbers exceeding the maximum k for which $s(p_1 \ldots p_k) \leq n$. Hence the nodes of $T[n]$ also comprise a finite set. See (4.1) or Figure 4.1 for $T[13]$.

It is apparent that a solution x of $s(x) = n > 1$ would appear as a leaf of $T[n]$, if it appeared in $T[n]$ at all, because nonterminal nodes always have s-values less then n . That <u>every</u> solution x of $s(x) = n$ is to be found among the terminal nodes of $T[n]$ will next be shown. Assume that x has the factorization A_k as defined by (4.2). Then Corollary 1.1 yields the ordering

$$s(1) = 0 < s(p_{I_1}^{e_1}) < s(p_{I_1}^{e_1} p_{I_2}^{e_2}) < \dots < s(x) = n .$$

Hence, by the way $T[n]$ is constructed, $A_0 \equiv 1$ is the father of $p_{I_1}^{e_1}$ is the father of $p_{I_1}^{e_1} p_{I_2}^{e_2}$ is the father of ... is the father of x, in the aliquot tree of n; that is,

$$\text{node } A_i = A_{i-1} p_{I_i}^{e_i} \text{ is the son of } A_{i-1} \text{ for } i = 1, 2, \dots, k .$$

Thus x is a node of $T[n]$.

In addition, the aliquot tree $T[n]$ contains <u>all</u> solutions to $1 < s(x) \leq n$ among its nodes. This follows immediately from the fact that every node of $T[n-1]$ qualifies as a node of $T[n]$.

In Section 7 algorithms will be given for exploiting the trees $T[n]$. Algorithm T there is a precise expression of the procedure for building $T[n]$ while visiting its nodes in preorder. This algorithm is introduced merely as a logical step; for it is soon replaced by a modification, Algorithm R, that takes advantage of the fact that evaluation of s values can be done without factoring numbers. If the reader will attempt to play through Algorithm T using the aliquot tree (4.1) as a test case, he will easily see the reasons behind the procedure: Just before visiting a node N at level $k \geq 0$ in step A2, we save it on a stack A with pointer k. When we get to step T3, we want to traverse the subtrees whose roots are the sons of N. This is done by successively visiting in preorder the sons of N and their respective subtrees, using the rules (4.2)-(4.3) for building the sons of an arbitrary father node of $T[n]$. After visiting these subtrees we will return to step T3 with the value N on top of stack A again. Then the stack is popped up at step T8 and we seek further sons of a node at one lower level, $k-1$.

Algorithm R in Section 7 is a modified version of Algorithm T to take advantage of the fact that s is a <u>"top-down" locally-defined function</u> of the nodes of $T[n]$; that is, s has the property that its value at a node x can be computed from the value x and the value of s at the father of x . Thus s should be evaluated at the father of a node before it is evaluated at the node as specified by Theorem 1. Then the evaluation of s values in Algorithm R is accomplished without factoring numbers.

A refinement to Algorithms T and R is possible if Goldbach solutions to $s(x) = n$ are not required. We use the result

<u>Theorem</u> 8. Let $A_k = \prod_{i=1}^{k} q_i^{e_i}$ with $k \geq 2$, $q_k \geq \sqrt{n}$, and $s(A_k) \leq n$. Then (i) $e_k = 1$ and (ii) $k > 2$ implies $q_1, \ldots, q_{k-1} < \sqrt{n}$.

Proof: Under the first two assumptions

$$s(A_k) > q_k^{e_k} \geq (\sqrt{n})^{e_k} ,$$

so that $n^{e_k/2} < s(A_k) \leq n$, which holds only if $e_k = 1$. Hence (i). If $k > 2$ and, contrariwise, $q_i \geq \sqrt{n}$ for some $i \leq k-1$, then

$$s(A_k) > q_i q_k \geq (\sqrt{n})^2 = n ,$$

which contradicts the third assumption. Hence (ii).

<u>Corollary</u> 8.1. Let $A_k = \prod_{i=1}^{k} q_i^{e_i}$ with $k \geq 2$, $q_k \geq \sqrt{n}$, and $s(A_k) = n$. Then

$$q_k = (n-A_{k-1})/s(A_{k-1}) - 1 .$$

Proof: $e_k = 1$ by Theorem 8. By Corollary 1.2,

$$s(A_{k-1}q_k) = n = (q_k+1)\ s(A_{k-1}) + A_{k-1} .$$

Now if Algorithm R is employed only to solve $s(x) = n$, we can replace step R3 with

R3'. [Terminate?] If $p[i] \leq \sqrt{n}$, then go to step R4 . If $k = 0$, then terminate. Set $t \leftarrow (n-A[k])/S[k]-1$. If $t > \sqrt{n}$ and t is prime, then $s(A[k]t) = n$. Go to step R6 .

and use "$p_i \leq \sqrt{n}$" in place of "$p_i \leq n$" in step R1 , thereby gaining considerable savings in the number of nodes traversed, at the expense of not finding all Goldbach solutions to $s(x) = n$. Although the algorithm would now omit certain solutions where x is the product of two primes, a check for these can be done by (1) test $n-1$ prime (if it is, $s((n-1)^2) = n$), and (2) test $n-1-p$ prime for some $p < n/2$ (if it is, $s(p(n-1-p)) = n$). The latter test can be programmed for high speed by using a packed bit table where the k-th bit is 1 iff $2k+1$ is prime. Then the test is made by <u>and</u>ing the entries to a corresponding bit table for $n-1-p$. Or by foregoing these "product of two primes" solutions, the number of primes needed by the algorithm is reduced from $\pi(n-1)+1$ to $\pi(\sqrt{n})+1$.

The next result can be used to further reduce the number of nodes in $T[n]$ when n is even and only those nodes x satisfying $s(x) = n$ are being sought:

<u>Theorem</u> 9. Let $s(x) = n$ be even. Then for all $k \geq 0$ and p odd, p^{2k+1} is never the first term $q_1^{e_1}$ in the prime factorization $\prod_i q_i^{e_i}$ of x.

Proof: Suppose, contrariwise, that $x = p^{2k+1}m$ and m has no prime factor $\leq p$. Then

$$n = s(x) = s(m)\ s(p^{2k+2}) + m\ s(p^{2k+1})\ .$$

But n and $s(p^{2k+2})$ are even, whereas $s(p^{2k+1})$ is odd. Hence m must be even, which contradicts our hypothesis that m has only prime factors $> p$.

Applying this result to the case $n = 6$ (see Figure 4.2), the nodes 3, 3^3, 5, and 5^3, <u>along with their subtrees</u>, never need to be considered as solutions to $s(x) = 6$.

5. Searching for sociable numbers

The usual approach to detect cycles is to examine the aliquot series starting successively with i ($i = 0, 1, 2, 3, \ldots, n$), and to compute this series $i, s(i), s^2(i), \ldots$ until a term exceeds some large number N or until a term equals some preceding term (in which case a cycle has been captured). In this approach one can stop with a particular series after detecting a cycle without missing other cycles because a generalized cycle of s contains at most one finite cycle. Algorithm E specifies the details.

A refinement of this straightforward approach is to keep track of the series elements which have already been examined; thus when $N = n = 284$, the cycle (284, 220) would not be detected after (220, 284) is found. Refer to Algorithm H for details.

Because Algorithm R can be used to generate efficiently (that is, without factoring numbers) all $0 \leq x \leq N$ for which $s(x) \leq N$, a fast method for detecting cycles is to store these s-values in a table S[0], S[1], ..., S[N] and then traverse this table systematically looking for cycles. Algorithm D gives details.

Comparisons between Algorithms E, H, and D will be made at this point. All three algorithms yield the finite cycles whose numbers do not exceed N and whose leader is $\leq n$. Algorithm E actually requires the least memory, but factors many numbers and always duplicates its work when a series leads into another one previously completed. Algorithm H also factors many numbers, but avoids duplication of s-value computations at the cost of memory;

it requires an additional Boolean array B of N+1 elements (or (N+1)/b locations if B is packed into computer words of b bits). Algorithm D coupled with Algorithm R requires no factorizations and less memory than Algorithm H . Table 5.1 summarizes these memory and factorization comparisons between the three algorithms for the "best" and "worst" cases.

	Number of factorizations		Memory locations for arrays	
Algorithm	minimum	maximum	minimum	maximum
E	n+1	(n+1)(N+1)	1	N+1
H	n+1	N+1	N+2*	2(N+1)*
D	N+1**	N+1**	N+1	N+1

* If array B is packed into b bit computer words, then the minimum and maximum become (N+1)/b+1 and (N+1)(1+1/b) , respectively.

** Or 0 if Algorithm R is used to generate the S array.

Table 5.1. "Best" and "worst" case analyses for data storage and for evaluation of s-values in Algorithms E, H, and D .

It is unfortunate that the crude procedure of Algorithm E seems to be the only feasible one for systematically detecting cycles when $n > 10^6$, because then both Algorithms H and D require too much storage even under ideal conditions, whereas Algorithm E requires that large numbers be repeatedly factored and the amount of computer time to do this rapidly exceeds practical limits. Instead of systematically exhausting leader possibilities from 1 to n and computing all of their series terms up to some large value N, restricting conditions can be placed on the leaders and/or their series terms, so that the total number of possiblities examined is reduced while the probability of finding a cycle is not reduced significantly. For example, Cohen tried all leaders to $n = 6.10^7$ but stopped computing their series after ten terms; even then his computer program ran for "around three weeks full time". Further conditions are considered in Section 6 and are based upon heuristic arguments and empirical observations on aliquot series.

6. Computed results

Programs to compute results of this Section were written entirely in ALGOL 60 (Grune 1970) for an Electrologica X8 computer (cycle time of 2.5 micro-seconds; 64K core memory of 27 bit words). Advanced features of ALGOL 60 such as recursion and Jensen's device were never used, so it is possible to code the algorithms directly in other high-level programming languages like FORTRAN, BASIC or MAD. Whenever machine time to compute a result exceeded 10 minutes, total time for that calculation is given to the nearest minute. The computer experiments which generated the statistics to follow are asserted to be both reliable and reproducible; for they are based upon algorithms analysed in Section 7 and they require only minimal amounts of machine time.

A description of the Tables in this Section follows. Table 6.1 lists every solution x and its prime factorization to the equation $s(x) = n$ for n from 0 to 100. For each n the number of such solutions, $d(n)$, is also given. Table 6.2 extends Table 6.1 to values of n between 101 and 500, only with the omission of Goldbach solutions $x = p_i p_j$ $(i \neq j)$. These solutions were omitted as uninteresting and to conserve space; they are easily computed separately by using the procedure set forth before Theorem 9 in Section 4. Table 6.6 restricts its pairs of values $(n, d(n))$ to the minimal odd values $n \leq 500$ for which $s(x) = n$ has only Goldbach solutions x. Table 6.4 gives the minimal odd solution n to $d(n) = k$ for k from 0 to 28. Table 6.3 presents every untouchable number, along with its prime factorization, below 5001. Table 6.10 tabulates the frequency distribution of the distances between successive untouchable numbers below 5000. Table 6.8 shows how many aliquot series lead into primes, perfect numbers, amicable numbers, Poulet's sociable series, or terms exceeding 10^{10}, based upon series leaders from 0 to 10000 and 1000 unit intervals of these leaders. Table 6.9 extends Table 6.8 to leaders up to 40000, using 10000 unit intervals. Table 6.5 sets forth those seven leaders $n \leq 1000$ which define series with "large" terms. Table 6.7 specifies the distribution of round numbers (those having six or more prime factors) among the amicable pairs below 10^8. Lastly, Table 6.11 tabulates the number of solutions n to $s(n) = k$ for $k = 0, 1, 2, \ldots$ and for $n \in [0,500]$.

A summary of how the Tables of this Section were programmed will now be given. Tables 6.1 and 6.2 were obtained by using Algorithm R as a subroutine to generate all x values such that $1 < s(x) \leq 500$. More explicitly, with $n = 500$ each time Algorithm R visited a node x of $T[n]$, the pair $(x,s(x))$ was saved in an array L; then L was sorted and the values of $d(x)$ were determined. Running time was 10 minutes. Tables 6.4 and 6.6 are readily derived as a byproduct.

Table 6.3 was also prepared by using Algorithm R as a subroutine, only with $n = 5000$. After initializing a 5000 element Boolean array B to "false", each time a node x of $T[5000]$ was visited, $B[s(x)]$ was set "true". Finally, x is untouchable if and only if $B[x] =$ "false". Running time was 18 minutes. Note that the straightforward method (based on Theorem 4) of computing $s(x)$ for all $x \leq 4999^2 = 24990001$ to find the untouchables below 5000 would require days of computer time.

Table 6.5 was the result of simply modifying Algorithm E with $n = 1000$ and $N = 1099511627775 = 2^{40}-1$ to output the appropriate information. Running time was 10 minutes.

Tables 6.8 and 6.9 were computed in 3 hours by simply enumerating the series $n, n_1, n_2, \ldots$ for each $n \leq 40000$ until it either became periodic or a term exceeded 10^{10}.

Table 6.10 was derived by hand from Table 6.3, while Table 6.7 was also hand constructed from the literature on amicable numbers in the interval $(0,10^8]$.

Next follow some conjectures and computed results which derive from the computational experiments described above. Each conjecture has been put into a form in which it can be further tested on a computer; numerical evidence is supplied for these conjectures. A computer can, of course, best settle a conjecture by finding a counterexample to it! However, there is meaning in allowing a computer to verify an infinite existence conjecture up to some high case, even though this verification cannot be duplicated by humans. For if the computer program used has been proved correct, then this program and its execution can be viewed as a finite, definite, and effective (Knuth 1968, pp. 4-6) process. Compare, for example, the "mathematically precise" result that an i-th prime always exists, although the case $i = 10^{80}$ cannot be exhibited. Indeed, only a computer experiment can provide even the first million primes with "sufficient rigor" for some people, and I would add the phrase "complete rigor" when a program correctness proof is supplied. When the correctness of a program, its compiler, and the hardware of the computer are all precisely established, then the output of that program can be viewed with the confidence of mathematical certainty. Thus the result of a careful computation is a mathematical fact and the cumulative results of calculations provide valuable data for an empirical mathematical study.

Dickson (1913) tabulated most aliquot series with leader $n < 1000$, but his tables contain many errors and he gave up whenever a series term exceeded 10^7. Calculating every aliquot series with leader $n < 1000$ by computer showed that these series are all periodic, except possibly for the six values of n displayed in Table 6.5 along with any series which lead into one of these six series. For example $s^{116}(696) = 2133148752623068133100$ and also $s^2(276) = s(396) = 696$; indeed, $n = 276$ is the smallest leader for which the behaviour of the series is unknown (Cohen has also calculated to $s^{118}(276)$). We state this new computed result equivalently as:

<u>Computed result</u> 1. An aliquot series with leader $n \leq 1000$ is periodic if it does not contain a term equal to one of the series terms whose leaders are 660, 696, 780, 840, 888, 966, or 990 .

Using multiple precision arithmetic along with methods (Knuth 1969) for factoring large numbers by computer, the series with leader 276 could be extended. Nevertheless, any series with a large even term will usually continue to have large even terms for a while.

<u>Conjecture</u> 1. The series with leader 276 extends to over 188 terms.

Evidence: Successive even terms of a series do not decrease rapidly. For by Corollary 4.1, as long as n_k is even, $n_{k+1} \geq n_k/2$; hence a series with leader n and all even terms cannot lead to 2 in fewer than $\lfloor \log_2 n \rfloor$ terms. Furthermore, an even term n_k rarely leads to an odd n_{k+1} (Theorem 2 states that n_{k+1} is odd iff every odd prime factor of n_k enters to an even power), so with high probability the series with leader $n = 276$ is neither a cycle, nor terminates, for at least

$$\lfloor \log_2 n_{118} \rfloor = 70$$

terms beyond $n_{118} = 213314875262306813 3100$.

According to the argument for Conjecture 1 applied to the maximum term $n_{117} = 179931895322$ of the series with leader $n = 138$, there would be at least $\lfloor \log_2 n_{117} \rfloor = 37$ terms after n_{117} . In fact, the final five terms of this series are:

$$n_{174} = 200$$

$$n_{175} = 265$$

$$n_{176} = 59$$

$$n_{177} = 1$$

$$n_{178} = 0$$

with all 57 terms from n_{118} to n_{174} being even.

It has been recently reported (personal note, February 1972) that the D.H. Lehmers have pursued the series 276 to its 349-th term, which has 31 decimal digits. Since $\lfloor \log_2 10^{30} \rfloor = 99$, we can update Conjecture 1 to:

<u>Conjecture</u> 2. The series with leader 276 extends to over 448 terms.

In further recent unpublished work, H. te Riele has shown that the series with leader $n = 3P$, where P is the largest perfect number currently known, must have at least 3000 strictly monotone increasing terms.

Further work in tabulating aliquot series with leader $n < 10^4$ has recently been done by Guy and Selfridge ("Interim report on aliquot series", November, 1971). They also report that a table of aliquot series through $n = 3040$ was deposited by G.A. Paxson in the UMT file in 1956.

A search for new sociable series was conducted by implementing Algorithms H and D . With $N = n = 200000$, Algorithm H ran for 1.1 hours without discovering something new; a more precise formulation of this statement is:

<u>Computed result</u> 2. The only sociable series

$$n, n_1, n_2, \ldots, n_k \quad \text{with} \quad n_i \leq 200000 \; (0 \leq i \leq k)$$

are the well-known perfect numbers, amicable pairs, and two cycles of Poulet.

With $N = n = 52000$, the output of Algorithm D supported this result. See Figure 7.3 for the corresponding profile.

Table 6.5. Values of $n \leq 1000$ such that the series with leader n *may* not terminate, or at least reaches a large term n_k which is difficult to factor. All values of $n \leq 1000$ which do not appear below are known to be leaders of series which either terminate or else lead into one of the series below.

n	k	n_k
660	134	357914540801318244984
696	116	2133148752623068133100
780	149	11666515530384271818
840	95	2243091044561433020754
888	105	40210935174977155764
966*	130	495428635818378741108

* This series is strictly monotone increasing up to n_k .

It would be interesting to know precisely - or even roughly - how Poulet (1918) discovered the two sociable series with leaders 12496 and 14316 . Poulet's series with leader $n = 12496$ has index 5 and the other has index 28 . See Table 1.1. Because these two cycles both contain round numbers (Hardy and Wright, section 22.14), the following possibility exists:

<u>Conjecture</u> 3. The two sociable series announced in 1918 by Poulet were determined by a systematic hand-calculation of those aliquot series whose leader is a round number $n < 10000$.

Evidence: A number n will be called round iff $\Omega(n) \geq 6$. This definition is based upon the function Ω (the number of prime factors) as a natural measure of "roundness". Because $\Omega(n)$ is usually about $\log \log n$ (Hardy and Wright, Theorem 436), a number near 10^7 will usually have about 3 prime factors and a number near 10^{80} about 5 or 6 . Thus $\Omega(n) \geq 6$ and $n < 10000$ imply that n is the product of a considerable number of comparatively small factors, which is the vague description of "roundness" for n .

Such round numbers (there are 901 of them) are easily read from a factor table to 10000 .

Given a round leader $n < 10000$ and the available factor tables, the series $n, n_1, n_2, \ldots, n_k$ could have been hand-computed until one of the following conditions was met:

(1) $n_k = 1$

(2) $n_k > 10^6$ (factor tables to ten million existed in 1909)

(3) $k > 30$

(4) n_k repeats a previous term.

This computation is amenable to humans and yields the two desired cycles, because $s(9464) = 12496$ and $s(7524) = 14316$. Further using Dickson's 1913 table of aliquot series with leaders < 1000 clearly allows one to also stop when

(5) $n_k < 1000$.

That a cycle (including perfect and amicable numbers) usually contains at least one round number is suggested by the two observations:

(i) $n, n_1, \ldots, n_k$ a cycle implies $n_i = s(n_{i-1}) \geq n_{i-1}$ for some $i \geq 1$ (that is, there exists at least one term m in the series such that $s(m) \geq m$);

(ii) a round number m often satisfies $s(m) > m$, whereas non-round numbers usually do not.

The 24 known perfect numbers are even and, except for the first three (6, 28 and 496), they are round; indeed, every even perfect number n is known to be of the form

$$n = 2^{p-1}(2^p-1), \quad \text{where } 2^p-1 \text{ is a Mersenne prime,}$$

so that $\Omega(n) = p$, which yields a round number for $p \geq 7$.

Among the 236 pairs of amicables whose lesser number is below 10^8 , there are 211 (89%) pairs which contain at least one round number. Refer to Table 6.7 for the distribution of round numbers among these 236 pairs less than 10^8 .

Note that a current digital computer requires an hour to work out by factorization every series with leader ≤ 10000 and terms $< 10^{10}$.

It has been observed that the known perfect numbers and amicable pairs usually include round numbers. This property also holds for the thirteen known sociable series. The two sociable series of Poulet contain 2 and 10 round numbers, respectively. The eleven sociable series of index four contain a total of 16 round numbers; only two of these series contain none, though they are rich in nearly round numbers.

Another empirical observation is that the known sociables contain 29 terms of the form

$$2^i pq \quad \text{for} \quad 2 \leq i \leq 4 \; ; \quad q > p > 2 \; ,$$

among their 77 numbers. Only two sociable series fail to contain a term of this form. Furthermore, it is an empirical fact that within each sociable series, except the Poulet series of index 28 , the series terms all have the same number of digits. Based upon these observations, a computer search was conducted for sociables with leader n above the 6.10^7 limit tried by Cohen (1970). Recall that he abandoned a series computation when the number of terms exceeded ten. In our computer search starting with leaders of the form

$$n = 2^i pq > 6.10^7 \quad (i = 2,3, \text{ or } 4 \; ; \quad q > p > 2) \; ,$$

a series calculation $n, n_1, \ldots, n_k$ was halted whenever any one of the following three conditions obtained:

(i) the number of decimal digits in n_k does not equal that in n .

(ii) the number of series terms exceeds thirty $(k \geq 30)$.

(iii) a series term n_k has a prime factor exceeding 10^8 .

The details are specified by the program in Figure 6.12. Execution time was 15 hours and no new sociables were discovered. The large running time was caused by the factorizations of many eight to ten digit numbers; an average of eight terms were computed for each of the $3.167.200 = 100200$ series considered. Nevertheless, this computer time is small compared to the "around 500 hours" of a Honeywell 516 (0.96 micr : cycle time) which Cohen reported he used.

How many aliquot series lead into prime numbers (and hence end in 1,0)? Do many series result in terms so large that computation of further terms becomes difficult? What is the frequency with which series "bump" into cycles such as perfect numbers, amicable pairs, and Poulet's two sociable series? To partially answer these questions, the series $n, n_1, n_2, \ldots, n_k$ with leader $n \leq 40000$ were computed until either:

(1) $n_k = 0$; (2) $n_k > 10^{10}$; or

(3) n_k = a term of some sociable series.

Table 6.8 shows the frequency of these three cases for n within 1000 unit intervals from 0 to 10000 , and Table 6.9 does the same for the four intervals of 10000 units from 0 to 40000.

<u>Figure</u> 6.12. Program to find those sociable series $n, n_1, \ldots, n_k$ with leader $n = 2^i pq$; $i = 2, 3$ or 4 ; $3 \leq p \leq 9973$; q equal to the first 200 prime values such that $n \geq 6.10^7$; $k < 30$; and each term n_j having the same number of digits as n .

<u>comment</u> p[i] = i-th prime, procedure s computes s-values, procedure digits computes the number of decimal digits in its argument, and procedure nextprime equals the index to the first ≥ its argument;

```
integer l, i, j, jmin, n, x, k;
for l:= 4, 8, 16 do
for i:= 2 step 1 until 168 do
begin jmin:= nextprime ((6*10↑7) ÷ (l*p[i]));
      for j:= jmin step 1 until jmin +199 do
      begin n:= x:= l * p[i] * p[j]; k:= 1;
            x:= (2*l-1) * (1+p[i]) * (1+p[j]) - x;
            for k:= k+1 while k ≤ 29 ∧ x ≠ n ∧ digits(x) =
                digits(n) do x:= s(x);
            if x = n then print(n)
      end
end;
```

Computer time used was three hours.

A summary of facts gleaned from computing Tables 6.8 and 6.9 follows. Over 85% of the series with leader to 40000 terminated in a cycle. A great number (a mean of 68.75 per 10000, with standard deviation 5.5) of these series ended in the perfect number 6, whereas only three (220, 284, and 562) ended in the amicable number 220. On the other hand, Poulet's two sociable series terminated numerous (0.1%) series considering the scarcity of such sociables; for instance, $s(17496) = s(18696) = 31704$ and $s^{28}(3360) = s^{28}(5784) = 376736$, both terms in the sociable series of index 28. Slightly more than 14% lead to terms exceeding 10^{10}; for example, $s^{44}(3876) > > 10^{10}$ and $s^{21}(840) > 2.10^{10}$. Some of these large terms occur only after many terms ($s^{213}(14004) = 17565705600$, and $s^{117}(138) = 179931895322$ which is the maximum term for the series with leader 138 before it goes "downhill" to the prime 59 at the term number 177), but a series can also terminate after many terms ($s^{208}(9126) = s^{210}(7686) = 59$) or it can remain small (1723148 from 3876 in 100 steps). The final possibility, a series which increases rapidly, also obtains (840 reaches 5.10^{11} in 26 steps).

Next, we investigate the behaviour of the in-degree function $d(n)$, which equals the number of solutions x to $s(x) = n$. The case $d(n) = 0$ is of particular interest for it means that n is untouchable. A list of the 570 untouchable numbers below 5000 is given in Table 6.3. After examining some empirical properties of these untouchables, we will return to consider the number of solutions n to $d(n) = k$ for $k = 1, 2, 3, \ldots$.

Interval	$n_k=0$	$n_k>10^{10}$	$n_k=$ perfect	$n_k=$ amicable	$n_k=$ Poulet sociable
(0, 1000]	948	30	19	3	0
(1000, 2000]	891	83	11	15	0
(2000, 3000]	878	96	10	15	1
(3000, 4000]	875	101	10	13	1
(4000, 5000]	874	108	8	10	0
(5000, 6000]	871	107	5	15	2
(6000, 7000]	864	117	5	14	0
(7000, 8000]	851	120	11	16	2
(8000, 9000]	841	141	7	10	1
(9000,10000]	853	127	7	10	3
(0,10000]	8746	1030	93	121	10
percentage	87.5%	10.3%	0.9%	1.2%	0.1%

Table 6.8. Distribution of "final" terms n_k in series $n, n_1, \ldots, n_k$ whose leaders n fall in 1000 unit intervals from 0 to 10000 .

Interval	$n_k=0$	$n_k>10^{10}$	$n_k=$ perfect	$n_k=$ amicable	$n_k=$ Poulet sociable
(0,10000]	8746	1030	93	121	10
(10000,20000]	8342	1417	79	144	18
(20000,30000]	8300	1496	75	121	8
(30000,40000]	8062	1733	78	109	18
(0,40000]	33450	5676	325	495	54
percentage	83.6%	14.2%	0.8%	1.2%	0.1%

Table 6.9. Distribution of "final" terms n_k in series $n, n_1, \ldots, n_k$ whose leaders n fall in 10000 unit intervals from 0 to 40000 .

Except for the case $n = 5$, the untouchable numbers in Table 6.3 are even in conformity with Theorem 6 and the extended Goldbach conjecture, so any two consecutive untouchables must have a distance that is at least equal to 2. Pairs of untouchables with this shortest distance will be called *untouchable twins*; for instance

$$(246, 248), (288, 290), (304, 306), \ldots, (4982, 4984) .$$

Similarly, triples of untouchable numbers such as

$$(322, 324, 326), (516, 518, 520), \ldots, (4980, 4982, 4984) ,$$

and quadruples of untouchable numbers such as

$$(892, 894, 896 , 898), \ldots, (4316, 4318, 4320, 4322) ,$$

which have minimum distance exist. The greatest distance between any two successive untouchable numbers below 5000 is the 62 units for the pair (2642, 2704). Table 6.10 displays the frequency $f(x)$ of occurrences of distance x between successive untouchables in the interval $(0,5000]$. The graph of nonzero f values looks roughly exponential and has a mean 8.8, standard deviation 7.8, mode 2, and median 6. There is no tendency for these distances to increase or decrease systematically as one considers larger untouchable pairs.

Table 6.10. Frequency distribution f of distances x between successive untouchable numbers below 5000 . All values of x not listed have frequency $f(x) = 0$.

x	2	3	4	6	8	10	12	14	16	18	20	22	24
$f(x)$	150	1	69	76	57	59	48	25	17	15	8	7	11

x	26	28	30	32	34	36	38	40	47	62
$f(x)$	4	3	6	3	2	2	2	2	1	1

The frequency distribution for the 570 untouchable numbers in the interval $(0,5000]$ is relatively uniform; there is a mean of 11.4 untouchables per 100 numbers, with standard deviation 3.16, minimum 5, maximum 18, and median 12. Let $y = f(x)$ be the number of untouchables in the interval $(0,x]$. Using the ten observed values (500, 38), (1000, 89), (1500, 144), (2000, 196), (2500, 263), (3000, 318), (3500, 379), (4000, 443), (4500, 509), (5000, 570) of (x,y), it is easy to see that a straight line provides a good fit for estimating y from x in the interval $(0,5000]$. Indeed, the least squares straight line through the origin is $\hat{y} = 0.10978x$, while $\hat{y} = -32.67 + 0.1191x$ if this least squares estimator is not forced through $(0,0)$. By extrapolation it appears there are an infinity of untouchable numbers; we conjecture the stronger result:

Conjecture 4. There exists an infinite number of untouchable numbers of the form $2p$, where p is an odd prime.

Evidence: Based on Table 6.3, for the 70 values

$$p = 73, 103, 131, \ldots, 2441,$$

the numbers $2p$ are untouchable. These account for over 12% of the 570 untouchable numbers below 5000. Since $\pi(2500) = 367$, over 14% of all even numbers below 5000 are the doubles of primes. This suggests that among even numbers, being untouchable and being the double of a prime are not independent events. The following 2×2 contingency table yields a chi-squared value of 3.08 (with Yates' correction), so that the hypothesis of independence is rejected at the 90% level ($\chi^2_{0.90} = 2.71$ with 1 degree of freedom):

$n = 2p$ for prime $p > 1$.

		YES	NO	
n untouchable	YES	70	499	569
	NO	297	1634	1931
		367	2133	2500

Contingency table for all positive even $n \leq 5000$.

Related to the number, $d(n)$, of solutions x to $s(x) = n$ is the function v_{2n} studied by Stein and Stein (1965), and Benedetti (1967). A "Goldbach decomposition" of the positive even integer $2n$ is defined to be any pair of primes $\{p_i, p_j\}$ satisfying the equation $p_i + p_j = 2n$. The possibility $p_i = 1$ is allowed. Then v_{2n} equals the number of distinct Goldbach decompositions of $2n$, and has been tabulated for all even arguments in the range $2n < 200000$. This table (Stein and Stein, TABLE IV) indicates that $v_{2n} > 50$ if $2n > 4688$, is true. Accordingly,

$$d(2n+1) \geq 49 \quad \text{for} \quad 4688 < 2n < 200000 ,$$

since the two cases $p_i = p_j$ and $p_i = 1$ must be excluded.

Experimentally, v_{2n} increases with n so that, for example, it further appears that $v_{2n} > 500$ when $2n > 85616$. A prescription for predicting v_{2n} is put forth by Stein and Stein. And obviously their table of v_{2n} versus $2n$ serves to bound $d(2n+1)$ since, in general, $d(2n+1) \geq v_{2n}-2$. This inequality ties in with Theorem 6 and leads to:

<u>Conjecture</u> 5. $\lim_{n\to\infty} d(2n+1) = \infty$.

By comparing $d(2n+1)$ with v_{2n} , checks on Tables 6.1, 6.2 and 6.6 are possible. For instance, $d(197) = 9 = v_{196}$ and in fact the 9 solutions of $s(x) = 197$ each yield Goldbach decompositions of 196 .

<u>Conjecture</u> 6. For every integer $k \geq 0$ there exists at least one odd number n such that $d(n) = k$.

Evidence: Based on the data of Table 6.4, it is true for all $k \leq 28$. Stein and Stein conjectured a similar result for v_{2n} and indeed, for $0 < k \leq 1911$, the number of solutions of the equation $v_{2n} = k$ is quite respectable. Furthermore, Table 6.6 suggests that these two conjectures are related because for positive $k \leq 24$ there exist odd numbers $2n + 1$ such that $s(x) = 2n + 1$ has <u>only</u> Goldbach solutions and hence $d(2n+1) = v_{2n} = k$ holds. An empirical tabulation, based on Tables 6.1 and 6.2, of the number of n such that $d(n) = k$ can be found in Table 6.11.

Table 6.11. Tabulation of the number of n which satisfy d(n) = k for k = 0, 1, 2, ..., ∞ . Based on Tables 6.1 and 6.2.

k	n∈[0,100]	(100,200]	(200,300]	(300,400]	(400,500]	[0,500]
0	5	5	12	8	8	38
1	31	21	16	22	23	113
2	25	18	17	12	14	86
3	14	5	3	6	3	31
4	7	3	2	0	1	13
5	7	11	0	2	1	21
6	6	5	4	2	0	17
7	1	5	4	2	0	12
8	2	6	3	1	1	13
9	2	6	13	6	2	29
10	0	3	7	6	2	18
11	.	2	2	6	4	14
12	.	3	1	4	4	12
13	.	5	1	2	5	13
14		1	0	4	9	14
15		1	3	1	1	6
16		0	4	1	2	7
17		.	2	1	3	6
18		.	1	2	0	3
19		.	1	1	1	3
20			4	3	0	7
21			0	3	3	6
22			.	0	4	4
23			.	2	3	5
24			.	1	0	1
25				1	0	1
26				0	2	2
27				1	0	1
28				0	1	1
29				.	0	0
30				.	1	1
31					0	0
32					2	2
33					0	0
.					.	.
.					.	.
.					.	.
∞	1	0	0	0	0	1

Table 6.1. Solutions of $s(x) = n$ for $0 \le n \le 100$.

n	d(n)	The d(n) values and prime factorizations of x such that s(x) = n .
0	2	0(0), 1(1).
1	∞	2(2), and every odd prime p .
2	0	untouchable.
3	1	$4(2^2)$.
4	1	$9(3^2)$.
5	0	untouchable.
6	2	6(2.3), $25(5^2)$.
7	1	$8(2^3)$.
8	2	10(2.5), $49(7^2)$.
9	1	15(3.5).
10	1	14(2.7).
11	1	21(3.7).
12	1	$121(11^2)$.
13	2	$27(3^3)$, 35(5.7).
14	2	22(2.11), $169(13^2)$.
15	2	$16(2^4)$, 33(3.11)
16	2	$12(2^2.3)$, 26(2.13).
17	2	39(3.13), 55(5.11).
18	1	$289(17^2)$
19	2	65(5.13), 77(7.11).
20	2	34(2.17), $361(19^2)$.
21	3	$18(2.3^2)$, 51(3.17), 91(7.13).
22	2	$20(2^2.5)$, 38(2.19).
23	2	57(3.19), 85(5.17).
24	1	$529(23^2)$.
25	3	95(5.19), 119(7.17), 143(11.13).
26	1	46(2.23).
27	2	69(3.23), 133(7.19),
28	1	$28(2^2.7)$.
29	2	115(5.23), 187(11.17).
30	1	$841(29^2)$.

n	d(n)	The d(n) values and prime factorizations of x such that s(x) = n .
31	5	32(2^5), 125(5^3), 161(7,23), 209,(11.19), 221(13,17).
32	2	58(2.29), 961(31^2).
33	3	45(3^2.5), 87(3.29), 247(13.19).
34	1	62(2.31).
35	3	93(3.31), 145(5.29), 253(11.23).
36	1	24(2^3.3).
37	4	155(5.31), 203(7.29), 299(13.23), 323(17.19).
38	1	1369(37^2).
39	1	217(7.31).
40	3	44(2^2.11), 74(2.37), 81(3^4).
41	4	63(3^2.7), 111(3.37), 319(11.29), 391(17.23)
42	2	30(2.3.5), 168$(41)^2$.
43	5	50(2.5^2), 185(5.37), 341(11.31), 377(13.29), 437(19.23).
44	2	82(2.41), 1849(43^2).
45	3	123(3.41), 259(7.37), 403(13.31).
46	2	52(2^2.13), 86(2.43).
47	3	129(3.43), 205(5.41), 493(17.29).
48	1	2209(47^2).
49	6	75(3.5^2), 215(5.43), 287(7.41), 407(11.37), 527(17.31), 551(19,29).
50	2	40(2^3.5), 94(2.47).
51	4	141(3.47), 301(7.43), 481(13.37), 589(19.31).
52	0	untouchable.
53	3	235(5.47), 451(11.41), 667(23.29).
54	2	42(2.37), 2809(53^2).
55	6	36(2^2.3^2), 329(7.47), 473(11.43), 533(13.41), 629(17.37), 713(23.31).
56	1	106(2.53).
57	5	99(3^2.11), 159(3.53), 343(7^3), 559(13.43), 703(19.37).
58	1	68(2^2.17).
59	3	265(5.53), 517(11.47), 697(17.41).
60	1	3481$(59)^2$.
61	6	371(7.53), 611(13.47), 731(17.43), 779(19.41), 851(23.37), 899(29.31).

n	d(n)	The d(n) values and prime factorizations of x such that s(x) = n .
62	2	118(2.59), 3721(61^2).
63	3	64(2^6), 177(3.59), 817(19.43).
64	3	56(2^3.7), 76(2^2.19), 122(2.61).
65	6	117(3^2.13), 183(3.61), 295(5.59), 583(11.53), 799(17.47), 943(23.41).
66	1	54(2.3^3).
67	6	305(5.61), 413(7.59), 689(13.53), 893(19.47), 989(23.43), 1073(29.37).
68	1	4489(67^2).
69	2	427(7.61), 1147(31.37).
70	1	134(2.67).
71	5	201(3.67), 649(11.59), 901(17.53), 1081(23.47), 1189(29.41).
72	1	5041(71^2).
73	8	98(2.7^2), 175(5^2.7), 335(5.67), 671(11.61), 767(13.59), 1007(19.53), 1247(29.43), 1271(31.41).
74	3	70(2.57), 142(2.71), 5329(73^2).
75	4	213(3.71), 469(7.67), 793(13.61), 1333(31.43).
76	3	48(2^4.3), 92(2^2.23), 146(2.73).
77	5	219(3.73), 355(5.71), 1003(17.59), 1219(23.53), 1363(29.47).
78	1	66(2.3.11).
79	7	365(5.73), 497(7.71), 737(11.67), 1037(17.61), 1121(19.59), 1457(31.47), 1517(37.41).
80	1	6241(79^2).
81	6	147(3.7^2), 153(3^2.17), 511(7.73), 871(13.67), 1159(19.61), 1591(37.43).
82	1	158(2.79).
83	4	237(3.79), 781(11.71), 1357(23.59), 1537(29.53).
84	1	6889(83^2).
85	8	395(5.79), 803(11.73), 923(13.71), 1139(17.67), 1403(23.61), 1643(31.53), 1739(37.47), 1763(41.43).
86	1	166(2.83).

n	d(n)	The d(n) values and prime factorizations of x such that s(x) = n .
87	5	105(3.5.7), 249(3.83), 553(7.79), 949(13.73), 1273(19.67).
88	0	untouchable.
89	5	171(3^2.19), 415(5.83), 1207(17.71), 1711(29.59), 1927(41.47).
90	2	78(2.3.13), 7921(89^2).
91	9	581(7.83), 869(11.79), 1241(17.73), 1349(19.71), 1541(23.67), 1769(29.61), 1829(31.59), 1961(37.53), 2021(43.47).
92	2	88(2^3.11), 178(2.89).
93	4	267(3.89), 1027(13.79), 1387(19.73), 1891(31.61).
94	1	116(2^2.29).
95	4	445(5.89), 913(11.83), 1633(23.71), 2173(41.53).
96	0	untouchable.
97	9	245(5.7^2), 275(5^2.11), 623(7.89), 1079(13.83), 1343(17.79), 1679(23.73), 1943(29.67), 2183(37.59), 2279(43.53).
98	1	9409(97^2).
99	3	1501(19.79), 2077(31.67), 2257(37.61).
100	2	124(2^2.31), 194(2.97).

Table 6.2. Non-Goldbach solutions of $s(x) = n$ for $101 \le n \le 500$.

n	d(n)	The non-Goldbach solutions and prime factorizations of x such that $s(x) = n$.
101	6	
102	1	$10201(101^2)$.
103	8	
104	2	$202(2.101)$, $10609(103^2)$.
105	7	$135(3^3.5)$, $207(3^2.23)$.
106	4	$80(2^4.5)$, $104(2^3.13)$, $110(2.5.11)$, $206(2.103)$.
107	5	
108	2	$60(2^2.3.5)$, $11449(107^2)$.
109	9	$325(5^2.13)$.
110	2	$214(2.107)$, $11881(109^2)$.
111	6	
112	1	$218(2.109)$.
113	7	
114	2	$102(2.3.17)$, $12769(113^2)$.
115	10	
116	1	$226(2.113)$.
117	7	$100(2^2.5^2)$.
118	1	$148(2^2.37)$.
119	5	
120	0	untouchable.
121	13	$243(3^5)$.
122	1	$130(2.5.13)$.
123	5	$72(2^3.3^2)$, $165(3.5.11)$.
124	0	untouchable.
125	5	
126	1	$114(2.3.19)$.
127	11	$128(2^7)$.
128	1	$16129(127^2)$.
129	4	$261(3^2.29)$.
130	2	$164(2^2.41)$, $254(2.127)$.
131	8	$189(3^3.7)$.

n	d(n)	The non-Goldbach solutions and prime factorizations of x such that s(x) = n .
132	1	17161(131^2).
133	11	425(5^2.17), 1331(11^3).
134	3	136(2^3.17), 154(2.7.11), 262(2.131).
135	5	
136	2	112(2^4.7), 172(2^2.43).
137	6	279(3^2.31).
138	1	18769(137^2).
139	8	
140	3	84(2^2.3.7), 274(2.137), 19321(139^2).
141	8	195(3.5.13).
142	1	278(2.139).
143	7	
144	1	90(2.3^2.5).
145	13	475(5^2.19), 539(7^2.11).
146	0	untouchable.
147	5	
148	2	152(2^3.19), 188(2^2.47).
149	5	
150	2	138(2.3.23), 22201(149^2).
151	12	
152	2	298(2.149), 22801(151^2).
153	5	231(3.7.11).
154	3	170(2.5.17), 182(2.7.13), 302(2.151).
155	8	
156	2	96(2^5.3), 625(5^4).
157	12	242(2.11^2).
158	1	24649(157^2).
159	4	
160	1	314(2.157).
161	10	333(3^2.37), 637(7^2.13).
162	0	untouchable.
163	10	
164	1	26569(163^2).
165	5	
166	2	212(2^2.53), 326(2.163).

n	d(n)	The non-Goldbach solutions and prime factorizations of x such that s(x) = n .
167	5	
168	1	27889(167^2).
169	15	363(3.11^2).
170	2	190(2.5.19), 334(2.167).
171	9	
172	1	108($2^2.3^3$).
173	6	
174	1	29929(173^2).
175	12	273(3.7.13).
176	2	184($2^3.23$), 346(2.173).
177	9	255(3.5.17), 369($3^2.41$).
178	1	225($3^2.5^2$).
179	6	
180	1	32041(179^2).
181	14	
182	2	358(2.179), 32761(181^2).
183	8	297($3^3.11$), 2197(13^3).
184	2	236($2^2.59$), 362(2.181).
185	9	387($3^2.43$).
186	2	126($2.3^2.7$), 174(2.3.29).
187	13	
188	0	untouchable.
189	5	
190	1	244($2^2.61$).
191	9	385(5.7.11).
192	1	36481(191^2).
193	13	605(5.11^2), 833($7^2.17$).
194	3	238(2.7.17), 382(2.191), 37249(193^2).
195	7	285(3.5.19).
196	3	140($2^2.5.7$), 176($2^4.11$), 386(2.193).
197	9	
198	2	186(2.3.31), 38809(197^2).
199	13	
200	2	394(2.197), 39601(199^2).

n	d(n)	The non-Goldbach solutions and prime factorizations of x such that s(x) = n .
201	10	$162(2.3^4)$, $423(3^2.47)$.
202	2	230(2.5.23), 398(2.199).
203	9	$196(2^2.7^2)$.
204	1	$132(2^2.3.11)$.
205	15	$725(5^2.29)$.
206	0	untouchable.
207	6	
208	1	$268(2^2.67)$.
209	9	$351(3^3.13)$, $931(7^2.19)$.
210	0	untouchable.
211	20	$338(2.13^2)$.
212	1	$44521(211^2)$.
213	6	
214	2	266(2.7.19), 422(2.11).
215	7	
216	0	untouchable.
217	16	455(5.7.13), $775(5^2.31)$, $847(7.11^2)$.
218	4	$160(2^5.5)$, $232(2^3.29)$, $250(2.5^3)$, 286(2.11.13).
219	7	357(3.7.17).
220	1	$284(2^2.71)$.
221	9	
222	1	$150(2.3.5^2)$.
223	11	
224	1	$49729(223^2)$.
225	9	$477(3^2.53)$, $507(3.13^2)$.
226	3	$208(2^4.13)$, $292(2^2.73)$, 446(2.223).
227	6	
228	1	$51529(227^2)$.
229	12	
230	2	454(2.227), $52441(229^2)$.
231	10	345(3.5.23).
232	2	$248(2^3.31)$, 458(2.229).
233	7	
234	2	222(2.3.37), $54289(233^2)$.

n	d(n)	The non-Goldbach solutions and prime factorizations of x such that s(x) = n .
235	15	
236	2	$156(2^2.3.13)$, $466(2.233)$.
237	9	
238	0	untouchable.
239	9	
240	2	$120(2^3.3.5)$, $57121(239^2)$.
241	20	$399(3.7.19)$, $1127(7^2.23)$.
242	2	$478(2.239)$, $58081(241^2)$.
243	9	$429(3.11.13)$.
244	2	$316(2^2.79)$, $482(2.241)$.
245	9	
246	0	untouchable.
247	16	
248	0	untouchable.
249	8	$375(3.5^3)$, $531(3^2.59)$.
250	1	$290(2.5.29)$.
251	9	
252	1	$63001(251^2)$.
253	18	$845(5.13^2)$, $925(5^2.37)$.
254	2	$322(2.7.23)$, $502(2.251)$.
255	9	$256(2^8)$.
256	1	$332(2^2.83)$.
257	9	$549(3^2.61)$.
258	2	$246(2.3.41)$, $66049(257^2)$.
259	15	$144(2^4.3^2)$.
260	1	$514(2.257)$.
261	11	$459(3^3.17)$.
262	0	untouchable.
263	8	
264	1	$69169(263^2)$.
265	17	$200(2^3.5^2)$.
266	2	$310(2.5.31)$, $526(2.263)$.
267	8	
268	0	untouchable.

n	d(x)	The non-Goldbach solutions and prime factorizations of x such that s(x) = n .
269	10	595(5.7.17).
270	3	198($2.3^2.11$), 258(2.3.43), 72361(269^2).
271	19	
272	2	538(2.269), 73441(271^2).
273	7	
274	4	296($2^3.37$), 356($2^2.89$), 374(2.11.17), 542(2.271).
275	10	
276	0	untouchable.
277	17	1025($5^2.41$).
278	1	76729(277^2).
279	6	
280	2	224($2^5.7$), 554(2.277).
281	16	603($3^2.67$), 1183(7.13^2).
282	1	78961(281^2).
283	16	
284	3	220($2^2.5.11$), 562(2.281), 80089(283^2).
285	10	435(3.5.29), 483(3.7.23).
286	2	272($2^4.17$), 566(2.283).
287	13	513($3^3.19$).
288	0	untouchable.
289	20	1075($5^2.43$), 1421($7^2.29$), 1573($11^2.13$).
290	0	untouchable.
291	10	
292	0	untouchable.
293	9	715(5.11.13).
294	2	282(2.3.47), 85849(293^2).
295	20	665(5.7.19).
296	1	586(2.293).
297	9	639($3^2.71$).
298	1	388($2^2.97$).
299	10	
300	2	204($2^2.3.17$), 441($3^2.7^2$).
301	21	
302	2	328($2^3.41$), 418(2.11.19).

n	d(n)	The non-Goldbach solutions and prime factorizations of x such that s(x) = n .
303	10	465(3.5.31), 561(3.11.17).
304	0	untouchable.
305	12	$657(3^2.73)$, $1519(7^2.31)$.
306	0	untouchable.
307	16	$4913(17^3)$.
308	1	$94249(307^2)$.
309	9	$315(3^2.5.7)$.
310	2	$404(2^2.101)$, 614(2.307).
311	12	
312	3	$168(2^3.3.7)$, $234(2.3^2.13)$, $96721(311^2)$.
313	18	$1175(5^2.47)$.
314	5	370(2.5.37), 406(2.7.29), 442(2.13.17), 622(2.311), $97969(313^2)$.
315	8	
316	5	$192(2^6.3)$, $304(2^4.19)$, $344(2^3.43)$, $412(2^2.103)$, 626(2.313).
317	10	
318	1	$100489(317^2)$.
319	15	
320	1	634(2.317).
321	12	$405(3^4.5)$.
322	0	untouchable.
323	11	
324	0	untouchable.
325	20	
326	0	untouchable.
327	6	
328	2	$260(2^2.5.13)$, $428(2^2.107)$.
329	11	$711(3^2.79)$.
330	1	318(2.3.53).
331	24	
332	2	$228(2^2.3.19)$, $109561(331^2)$.
333	7	627(3.11.19).
334	3	434(2.7.31), $436(2^2.109)$, 662(2.331).

n	d(n)	The non-Goldbach solutions and prime factorizations of x such that s(x) = n .
335	10	
336	0	untouchable.
337	21	$1859(11.13^2)$, $2057(11^2.17)$.
338	1	$113569(337^2)$.
339	10	$621(3^3.23)$.
340	1	674(2.337).
341	13	
342	0	untouchable.
343	19	$578(2.17^2)$, 1001(7.11.13).
344	1	$376(2^3.47)$.
345	12	663(3.13.17), $747(3^2.83)$.
346	3	410(2.5.41), $452(2^2.113)$, 494(2.13.19).
347	9	805(5.7.23).
348	1	$120409(347^2)$.
349	17	$1325(5^2.53)$.
350	2	694(2.347), $121801(349^2)$.
351	14	609(3.7.29).
352	1	698(2.349).
353	11	$1813(7^2.37)$.
354	1	$124609(353^2)$.
355	20	
356	1	706(2.353).
357	10	555(3.5.37).
358	1	506(2.11.23).
359	9	
360	1	$128881(359^2)$.
361	25	$867(3.17^2)$, 935(5.11.17), $2299(11^2.19)$.
362	2	430(2.5.43), 718(2.359).
363	7	
364	2	$308(2^2.7.11)$, $729(3^6)$.
365	14	
366	3	$180(2^2.3^2.5)$, 210(2.3.5.7), 354(2.3.59).
367	18	
368	1	$134689(367^2)$.

n	d(n)	The non-Goldbach solutions and prime factorizations of x such that s(x) = n .
369	9	$801(3^2.89)$.
370	1	$734(2.367)$.
371	14	
372	0	untouchable.
373	20	$651(3.7.31)$, $875(5^3.7)$.
374	1	$139129(373^2)$.
375	10	
376	2	$368(2^4.23)$, $746(2.373)$.
377	11	
378	1	$366(2.3.61)$.
379	23	$741(3.13.19)$.
380	1	$143641(379^2)$.
381	14	$6859(19^3)$.
382	1	$758(2.379)$.
383	9	
384	2	$216(2^3.3^3)$, $146689(383^2)$.
385	21	$1475(5^2.59)$, $2009(7^2.41)$.
386	2	$424(2^3.53)$, $766(2.383)$.
387	11	
388	1	$508(2^2.127)$.
389	9	
390	2	$294(2.3.7^2)$, $151321(389^2)$.
391	27	
392	1	$778(2.389)$.
393	13	$615(3.5.41)$, $759(3.11.23)$.
394	3	$350(2.5^2.7)$, $470(2.5.47)$, $518(2.7.37)$.
395	11	$1045(5.11.19)$.
396	2	$276(2^2.3.23)$, $306(2.3^2.17)$.
397	23	$1445(5.17^2)$, $1525(5^2.61)$.
398	1	$157609(397^2)$.
399	6	
400	3	$524(2^2.131)$, $794(2.397)$, $2401(7^4)$.
401	17	$567(3^4.7)$, $873(3^2.97)$, $2107(7^2.43)$.

n	d(n)	The non-Goldbach solutions and prime factorizations of x such that s(x) = n .
402	1	160801(401^2).
403	17	
404	2	352(2^5.11), 802(2.401).
405	11	
406	0	untouchable.
407	14	1105(5.13.17).
408	0	untouchable.
409	21	2783(11^2.23).
410	2	598(2.13.23), 167281(409^2).
411	14	645(3.5.43).
412	1	818(2.409).
413	11	
414	1	402(2.3.67).
415	21	
416	1	340(2^2.5.17).
417	12	783(3^3.29), 909(3^2.101).
418	1	548(2^2.137).
419	12	1309(7.11.17).
420	2	364(2^2.7.13), 175561(419^2).
421	32	722(2.19^2), 2873(13^2.17).
422	2	838(2.419), 177241(421^2).
423	10	
424	2	556(2^2.139), 842(2.421).
425	14	927(3^2.103), 1015(5.7.29).
426	0	untouchable.
427	21	
428	1	472(2^3.59).
429	9	
430	0	untouchable.
431	14	
432	1	185761(431^2).
433	22	1675(5^2.67), 2023(7.17^2), 2303(7^2.47).
434	4	574(2.7.41), 646(2.17.19), 862(2.431), 187489(433^2).
435	13	

n	d(n)	The non-Goldbach solutions and prime factorizations of x such that s(x) = n .
436	1	866(2.433).
437	11	
438	2	$342(2.3^2.19)$, 426(2.3.71).
439	22	777(3.7.37).
440	2	$280(2^3.5.7)$, $192721(439^2)$.
441	17	$495(3^2.5.11)$, $963(3^2.107)$, $1083(3.19^2)$.
442	5	$320(2^6.5)$, $488(2^3.61)$, 530(2.5.53), 638(2.11,29), 878(2.439).
443	14	$837(3^3.31)$.
444	1	$196249(443^2)$.
445	22	1235(5.13.19).
446	1	886(2.443).
447	14	$484(2^2.11^2)$, 705(3.5.47), 879(3.13.23).
448	0	untouchable.
449	15	$981(3^2.109)$, $3211(13^2.19)$.
450	3	$270(2.3^3.5)$, 438(2.3.73), $201601(449^2)$.
451	28	1085(5.7.31).
452	1	898(2.449).
453	12	
454	2	$596(2^2.149)$, 602(2.7.43).
455	11	
456	1	$264(2^3.3.11)$.
457	26	1463(7.11.19), $1775(5^2.71)$.
458	1	$208849(457^2)$.
459	8	
460	3	$380(2^2.5.19)$, $604(2^2.151)$, 914(2.457).
461	16	
462	1	$212521(461^2)$.
463	30	$392(2^3.7^2)$, 1265(5.11.23).
464	2	922(2.461), $214369(463^2)$.
465	13	$1017(3^2.113)$.
466	3	$416(2^5.13)$, $464(2^4.29)$, 926(2.463).
467	13	$525(3.5^2.7)$.

n	d(n)	The non-Goldbach solutions and prime factorizations of x such that s(x) = n .
468	1	218089(467^2).
469	26	1547(7.13.17), 1825(5^2.73).
470	2	682(2.11.31), 934(2.467).
471	16	969(3.17.19).
472	0	untouchable.
473	13	
474	0	untouchable.
475	23	
476	1	252(2^2.3^2.7).
477	14	
478	1	628(2^2.157).
479	10	
480	1	229441(479^2).
481	32	1805(5.19^2), 2597(7^2.53), 3509(11^2.29).
482	1	958(2.479).
483	12	861(3.7.41), 957(3.11.29).
484	1	536(2^3.67).
485	14	
486	1	474(2.3.79).
487	23	
488	1	237169(487^2).
489	9	
490	2	590(2.5.59), 974(2.487).
491	19	
492	2	348(2^2.3.29), 241081(491^2).
493	22	
494	2	658(2.7.47), 982(2.491).
495	13	
496	2	496(2^4.31), 652(2^2.163).
497	14	1375(5^3.11).
498	0	untouchable.
499	23	
500	1	249001(499^2).

Table 6.3. Untouchable numbers $n \leq 5000$.

Values and prime factorizations of n such that $s(x) = n$ has no solution.

$2(2)$	$406(2.7.29)$	$738(2.3^2.41)$
$5(5)$	$408(2^3.3.17)$	$748(2^2.11.17)$
$52(2^2.13)$	$426(2.3.71)$	$750(2.3.5^3)$
$88(2^3.11)$	$430(2.5.43)$	$756(2^2.3^3.7)$
$96(2^5.3)$	$448(2^6.7)$	$766(2.383)$
$120(2^3.3.5)$	$472(2^3.59)$	$768(2^8.3)$
$124(2^2.31)$	$474(2.3.79)$	$782(2.17.23)$
$146(2.73)$	$498(2.3.83)$	$784(2^4.7^2)$
$162(2.3^4)$	$516(2^2.3.43)$	$792(2^3.3^2.11)$
$188(2^2.47)$	$518(2.7.37)$	$802(2.401)$
$206(2.103)$	$520(2^3.5.13)$	$804(2^2.3.61)$
$210(2.3.5.7)$	$530(2.5.53)$	$818(2.409)$
$216(2^3.3^3)$	$540(2^2.3^3.5)$	$836(2^2.11.19)$
$238(2.7.17)$	$552(2^3.3.23)$	$848(2^4.53)$
$246(2.3.41)$	$556(2^2.139)$	$852(2^2.3.71)$
$248(2^3.31)$	$562(2.281)$	$872(2^3.109)$
$262(2.131)$	$576(2^6.3^2)$	$892(2^2.223)$
$268(2^2.67)$	$584(2^3.73)$	$894(2.3.149)$
$276(2^2.3.23)$	$612(2^2.3^2.17)$	$896(2^7.7)$
$288(2^5.3^2)$	$624(2^4.3.13)$	$898(2.449)$
$290(2.5.29)$	$626(2.313)$	$902(2.11.41)$
$292(2^2.73)$	$628(2^2.157)$	$926(2.463)$
$304(2^4.19)$	$658(2.7.47)$	$934(2.467)$
$306(2.3^2.17)$	$668(2^2.167)$	$936(2^3.3^2.13)$
$322(2.7.23)$	$670(2.5.67)$	$964(2^2.241)$
$324(2^2.3^4)$	$708(2^2.3.59)$	$966(2.3.7.23)$
$326(2.163)$	$714(2.3.7.17)$	$976(2^4.61)$
$336(2^4.3.7)$	$718(2.359)$	$982(2.491)$
$342(2.3^2.19)$	$726(2.3.11^2)$	$996(2^2.3.83)$
$372(2^2.3.31)$	$732(2^2.3.61)$	$1002(2.3.167)$

Values and prime factorizations of n such that s(x) = n has no solution.

1028(2^2.257)
1044(2^2.3^2.29)
1046(2.523)
1060(2^2.5.53)
1068(2^2.3.89)
1074(2.3.179)
1078(2.7^2.11)
1080(2^3.3^3.5)
1102(2.19.29)
1116(2^2.3^2.31)
1128(2^3.3.47)
1134(2.3^4.7)
1146(2.3.191)
1148(2^2.7.41)
1150(2.5^2.23)
1160(2^3.5.29)
1162(2.7.83)
1168(2^4.73)
1180(2^2.5.59)
1186(2.593)
1192(2^3.149)
1200(2^4.3.5^2)
1212(2^2.3.101)
1222(2.13.47)
1236(2^2.3.103)
1246(2.7.89)
1248(2^5.3.13)
1254(2.3.11.19)
1256(2^3.157)
1258(2.17.37)
1266(2.3.211)
1272(2^3.3.211)
1288(2^3.7.23)
1296(2^4.3^4)
1312(2^5.41)
1314(2.3^2.73)
1316(2^2.7.47)
1318(2.659)
1326(2.3.13.17)
1332(2^2.3^2.37)
1342(2.11.61)
1346(2.673)
1348(2^2.337)
1360(2^4.5.17)
1380(2^2.3.5.23)
1388(2^2.347)
1398(2.3.233)
1404(2^2.3^3.13)
1406(2.19.37)
1418(2.709)
1420(2^2.5.71)
1422(2.3^2.79)
1438(2.719)
1476(2^2.3^2.41)
1506(2.3.251)
1508(2^2.13.29)
1510(2.5.151)
1522(2.761)
1528(2^3.191)
1538(2.769)
1542(2.3.257)
1566(2.3^3.29)
1578(2.3.263)
1588(2^2.397)
1596(2^2.3.7.19)
1632(2^5.3.17)
1642(2.821)
1650(2.3.5^2.11)
1680(2^4.3.5.7)
1682(2.29^2)
1692(2^2.3^2.47)
1716(2^2.3.11.13)
1718(2.859)
1728(2^6.3^3)
1732(2^2.433)
1746(2.3^2.97)
1758(2.3.293)
1766(2.883)
1774(2.887)
1776(2^4.3.37)
1806(2.3.7.43)
1816(2^3.227)
1820(2^2.5.7.13)
1822(2.911)
1830(2.3.5.61)
1838(2.919)
1840(2^4.5.23)
1842(2.3.307)
1844(2^2.461)
1852(2^2.463)
1860(2^2.3.5.31)
1866(2.3.311)
1884(2^2.3.157)
1888(2^5.59)
1894(2.947)
1896(2^3.3.79)
1920(2^7.3.5)
1922(2.31^2)
1944(2^3.3^5)

Values and prime factorizations of n such that $s(x) = n$ has no solution.

1956(2^2.3.163)
1958(2.11.89)
1960(2^3.5.7^2)
1962(2.3^2.109)
1972(2^2.17.29)
1986(2.3.331)
1992(2^3.3.83)
2008(2^3.251)
2010(2.3.5.67)
2022(2.3.337)
2024(2^3.11.23)
2036(2^2.509)
2048(2^{11})
2050(2.5^2.41)
2052(2^2.3^3.19)
2058(2.3.7^3)
2062(2.1031)
2068(2^2.11.47)
2078(2.1039)
2096(2^4.131)
2098(2.1049)
2108(2^2.17.31)
2118(2.3.353)
2120(2^3.5.53)
2128(2^4.7.19)
2136(2^3.3.89)
2148(2^2.3.179)
2152(2^3.269)
2158(2.13.83)
2168(2^3.271)
2174(2.1087)
2178(2.3^2.11^2)
2190(2.3.5.73)
2196(2^2.3^2.61)
2198(2.7.157)
2212(2^2.7.79)
2218(2.1109)
2226(2.3.7.53)
2228(2^2.557)
2232(2^3.3^2.31)
2248(2^3.281)
2258(2.1129)
2262(2.3.13.29)
2302(2.1151)
2304(2^8.3^2)
2306(2.1153)
2316(2^2.3.193)
2322(2.3^3.43)
2324(2^2.7.83)
2330(2.5.233)
2338(2.7.167)
2356(2^2.19.31)
2364(2^2.3.197)
2366(2.7.13^2)
2376(2^2.3^3.11)
2388(2^2.3.199)
2404(2^2.601)
2408(2^3.7.43)
2410(2.5.241)
2416(2^4.151)
2422(2.7.173)
2430(2.3^5.5)
2432(2^7.19)
2436(2^2.3.7.29)
2446(2.1223)
2452(2^2.613)
2454(2.3.409)
2464(2^5.7.11)
2482(2.17.73)
2484(2^2.3^3.23)
2490(2.3.5.83)
2496(2^6.3.13)
2498(2.1249)
2500(2^2.5^4)
2502(2.3^2.139)
2514(2.3.419)
2518(2.1259)
2530(2.5.11.23)
2564(2^2.641)
2568(2^3.3.107)
2572(2^2.643)
2576(2^4.7.23)
2586(2.3.431)
2588(2^2.647)
2590(2.5.7.37)
2600(2^3.5^2.13)
2602(2.1301)
2606(2.1303)
2608(2^4.163)
2614(2.1307)
2628(2^2.3^2.73)
2640(2^4.3.5.11)
2642(2.1321)
2704(2^4.13^2)
2718(2.3^2.151)
2724(2^2.3.227)
2726(2.29.47)
2736(2^4.3^2.19)
2748(2^2.3.229)

Values and prime factorizations of n such that s(x) = n has no solution.

2758(2.7.197)	3036(2^2.3.11.23)	3312(2^4.3^2.23)
2760(2^3.3.5.23)	3060(2^2.3^2.5.17)	3318(2.3.7.79)
2762(2.1381)	3072(2^{10}.3)	3328(2^8.13)
2766(2.3.461)	3076(2^2.769)	3340(2^2.5.167)
2774(2.19.73)	3078(2.3^4.19)	3356(2^2.839)
2784(2^5.3.29)	3102(2.3.11.47)	3366(2.3^2.11.17)
2788(2^2.17.41)	3104(2^5.97)	3378(2.3.563)
2808(2^3.3^3.13)	3114(2.3^2.173)	3384(2^3.3^2.47)
2824(2^3.353)	3126(2.3.521)	3388(2^2.7.11^2)
2828(2^2.7.101)	3132(2^2.3^3.29)	3396(2^2.3.283)
2850(2.3.5^2.19)	3136(2^6.7^2)	3400(2^3.5^2.17)
2856(2^3.3.7.17)	3142(2.1571)	3402(2.3^5.7)
2874(2.3.479)	3144(2^3.3.131)	3406(2.13.131)
2876(2^2.719)	3152(2^4.197)	3412(2^2.853)
2894(2.1447)	3156(2^2.3.263)	3420(2^2.3^5.5.19)
2902(2.1451)	3162(2.3.17.31)	3422(2.29.59)
2914(2.31.47)	3174(2.3.23^2)	3428(2^2.857)
2922(2.3.487)	3186(2.3^3.59)	3430(2.5.7^3)
2932(2^2.733)	3198(2.3.13.41)	3432(2^3.3.11.13)
2944(2^7.23)	3202(2.1601)	3448(2^3.431)
2946(2.3.491)	3208(2^3.401)	3454(2.11.157)
2950(2.5^2.59)	3228(2^2.3.269)	3476(2^2.11.79)
2952(2^3.3^2.41)	3234(2.3.7^2.11)	3484(2^2.13.67)
2968(2^3.7.53)	3236(2^2.809)	3486(2.3.7.83)
2978(2.1489)	3238(2.1619)	3488(2^5.109)
2982(2.3.7.71)	3246(2.3.541)	3504(2^4.3.73)
2984(2^3.373)	3266(2.23.71)	3506(2.1753)
2992(2^4.11.17)	3270(2.3.5.109)	3510(2.3^3.5.13)
2994(2.3.499)	3276(2^2.3^2.7.13)	3524(2^2.881)
2996(2^2.7.107)	3278(2.11.149)	3538(2.29.61)
3008(2^6.47)	3292(2^2.823)	3556(2^2.7.127)
3018(2.3.503)	3296(2^5.103)	3564(2^2.3^4.11)
3028(2^2.757)	3306(2.3.19.29)	3576(2^3.3.149)

<u>Values and prime factorizations of n such that s(x) = n has no solution</u>.

$3580(2^{2}.5.179)$	$3820(2^{2}.5.191)$	$4072(2^{3}.509)$
$3588(2^{2}.3.13.23)$	$3828(2^{2}.3.11.29)$	$4078(2.2039)$
$3590(2.5.359)$	$3832(2^{3}.479)$	$4086(2.3^{2}.227)$
$3592(2^{3}.449)$	$3842(2.17.113)$	$4088(2^{3}.7.73)$
$3600(2^{4}.3^{2}.5^{2})$	$3860(2^{2}.5.193)$	$4098(2.3.683)$
$3604(2^{2}.17.53)$	$3862(2.1931)$	$4104(2^{3}.3^{3}.19)$
$3630(2.3.5.11^{2})$	$3868(2^{2}.967)$	$4116(2^{2}.3.7^{3})$
$3636(2^{2}.3^{2}.101)$	$3872(2^{5}.11^{2})$	$4120(2^{3}.5.103)$
$3642(2.3.607)$	$3876(2^{2}.3.17.19)$	$4148(2^{2}.17.61)$
$3648(2^{6}.3.19)$	$3888(2^{4}.3^{5})$	$4168(2^{3}.521)$
$3650(2.5^{2}.73)$	$3900(2^{2}.3.5^{2}.13)$	$4170(2.3.5.139)$
$3652(2^{2}.11.83)$	$3902(2.1951)$	$4172(2^{2}.7.149)$
$3656(2^{3}.457)$	$3904(2^{6}.61)$	$4184(2^{3}.523)$
$3666(2.3.13.47)$	$3936(2^{5}.3.41)$	$4188(2^{2}.3.349)$
$3670(2.5.367)$	$3940(2^{2}.5.197)$	$4190(2.5.419)$
$3682(2.7.263)$	$3942(2.3^{3}.73)$	$4198(2.2099)$
$3684(2^{2}.3.307)$	$3954(2.3.659)$	$4206(2.3.701)$
$3708(2^{2}.3^{2}.103)$	$3958(2.1979)$	$4216(2^{3}.17.31)$
$3738(2.3.7.89)$	$3960(2^{3}.3^{2}.5.11)$	$4224(2^{7}.3.11)$
$3744(2^{5}.3^{2}.13)$	$3972(2^{2}.3.331)$	$4238(2.13.163)$
$3746(2.1873)$	$3974(2.1987)$	$4248(2^{3}.3^{2}.59)$
$3748(2^{2}.937)$	$3982(2.11.181)$	$4258(2.2129)$
$3752(2^{3}.7.67)$	$3986(2.1993)$	$4268(2^{2}.11.97)$
$3758(2.1879)$	$4018(2.7^{2}.41)$	$4280(2^{3}.5.107)$
$3760(2^{4}.5.47)$	$4026(2.3.11.61)$	$4296(2^{3}.3.179)$
$3774(2.3.17.37)$	$4032(2^{6}.3^{2}.7)$	$4302(2.3^{2}.239)$
$3786(2.3.631)$	$4036(2^{2}.1009)$	$4304(2^{4}.269)$
$3788(2^{2}.947)$	$4046(2.7.17^{2})$	$4308(2^{2}.3.359)$
$3792(2^{4}.3.79)$	$4048(2^{4}.11.23)$	$4312(2^{3}.7^{2}.11)$
$3808(2^{5}.7.17)$	$4056(2^{3}.3.13^{2})$	$4316(2^{2}.13.83)$
$3812(2^{2}.953)$	$4062(2.3.677)$	$4318(2.17.127)$
$3816(2^{3}.3^{2}.53)$	$4068(2^{2}.3^{2}.113)$	$4320(2^{5}.3^{3}.5)$
$3818(2.23.83)$	$4070(2.5.11.37)$	$4322(2.2161)$

Values and prime factorizations of n such that s(x) = n has no solution.

$4336(2^4.271)$
$4344(2^3.3.181)$
$4356(2^2.3^2.11^2)$
$4368(2^4.3.7.13)$
$4370(2.5.19.23)$
$4380(2^2.3.5.73)$
$4382(2.7.313)$
$4386(2.3.17.43)$
$4388(2^2.1097)$
$4396(2^2.7.157)$
$4402(2.31.71)$
$4406(2.2203)$
$4416(2^6.3.23)$
$4430(2.5.443)$
$4462(2.23.97)$
$4472(2^3.13.43)$
$4476(2^2.3.373)$
$4480(2^7.5.7)$
$4488(2^3.3.11.17)$
$4490(2.5.449)$
$4492(2^2.1123)$
$4498(2.13.173)$
$4500(2^2.3^2.5^3)$
$4506(2.3.751)$
$4512(2^5.3.47)$
$4530(2.3.5.151)$
$4534(2.2267)$
$4574(2.2287)$
$4580(2^2.5.229)$
$4588(2^2.31.37)$
$4612(2^2.1153)$
$4614(2.3.769)$
$4618(2.2309)$

$4630(2.5.463)$
$4648(2^3.7.83)$
$4662(2.3^2.7.37)$
$4668(2^2.3.389)$
$4672(2^6.73)$
$4678(2.2339)$
$4686(2.3.11.71)$
$4688(2^4.293)$
$4690(2.5.7.67)$
$4700(2^2.5^2.47)$
$4710(2.3.5.157)$
$4712(2^3.19.31)$
$4718(2.7.337)$
$4738(2.23.103)$
$4740(2^2.3.5.79)$
$4742(2.2371)$
$4748(2^2.1187)$
$4750(2.5^3.19)$
$4758(2.3.13.61)$
$4764(2^2.3.397)$
$4770(2.3^2.5.53)$
$4772(2^2.1193)$
$4782(2.3.797)$
$4808(2^3.601)$
$4830(2.3.5.7.23)$
$4838(2.41.59)$
$4840(2^3.5.11^2)$
$4842(2.3^2.269)$
$4850(2.5^2.97)$
$4854(2.3.809)$
$4856(2^3.607)$
$4869(2^2.3^5.5)$
$4868(2^2.1217)$

$4882(2.441)$
$4884(2^2.3.11.37)$
$4886(2.7.349)$
$4896(2^5.3^2.17)$
$4898(2.31.79)$
$4908(2^2.3.409)$
$4914(2.3^3.7.13)$
$4916(2^2.1229)$
$4926(2.3.821)$
$4928(2^6.7.11)$
$4942(2.7.353)$
$4956(2^2.3.7.59)$
$4962(2.3.827)$
$4964(2^2.17.73)$
$4980(2^2.3.5.83)$
$4982(2.47.53)$
$4984(2^3.7.89)$
$4998(2.3.7^2.17)$

n	d(n)	n	d(n)
5	0	169	15
3	1	217	16
13	2	265	17
21	3	253	18
37	4	271	19
31	5	211	20
49	6	301	21
79	7	433	22
73	8	379	23
91	9	331	24
115	10	361	25
127	11	457	26
151	12	391	27
121	13	451	28
181	14		

Table 6.4. The minimum odd solution n to $d(n) = k$ for each $k \leq 28$.

n	d(n)	n	d(n)
9	1	187	13
17	2	181	14
25	3	235	15
37	4	247	16
71	5	403	17
61	6	367	18
79	7	271	19
85	8	325	20
91	9	301	21
115	10	493	22
223	11	475	23
151	12	331	24

Table 6.6. The minimum odd solution n to $d(n) = k$, for each positive $k \leq 24$, such that every one of the $d(n)$ solutions to $s(x) = n$ is a Goldbach solution.

Table 6.7. Distribution of round numbers $(x: \Omega(x) \geq 6)$ among amicable pairs whose lesser number $\leq 10^8$.

Interval	Number of amicable pairs	Both numbers of pair round	Single numbers of pair round	Neither numbers of pair round
$(0,10^5]$	13	2(15%)	6(46%)	5(38%)
$(10^5,10^6]$	29	8(28%)	18(62%)	3(10%)
$(10^6,10^7]$	66	27(41%)	27(41%)	12(18%)
$(10^7,10^8]$	128	83(65%)	40(31%)	5(4%)
$(0,10^8]$	236	120(51%)	91(39%)	25(11%)

7. Algorithms

Contained in this Section are the five Algorithms mentioned in Sections 4 and 5. The format chosen for the Algorithms is based upon the style in (Knuth 1968, page 2). An effort is made to analyse these Algorithms so the reader will be convinced that each computer procedure is unambiguously specified, does terminate, has well-defined input and output, and can be performed in a reasonable number of steps. Correctness proofs for nontrivial program sections are outlined. In addition, a study of the properties of the Algorithms is attempted; for example, a frequency analysis (how many times each part of the algorithm is likely to be executed) and a storage analysis (how much memory it is likely to need) is specified. The general principles used in the field of algorithmic analysis are described in (Knuth 1971).

An analysis is designed to measure relevant factors about the performance of an algorithm by studying properties of that algorithm. For examples, consider the frequency analysis (Figure 7.2) of Algorithm R and the storage analysis (Figure 5.1) of Algorithm D. With $n = 5000$, Algorithm R requires 268074 steps versus the 24990001 steps if straightforward enumeration is used. Algorithm D is efficient with respect to factorization steps, but its memory requirements exceed practical bounds for large n, say $n > 10^5$. Thus, the analysis of these two Algorithms directly assists in measuring their computational efficiency in terms of "steps" executed and auxiliary memory required.

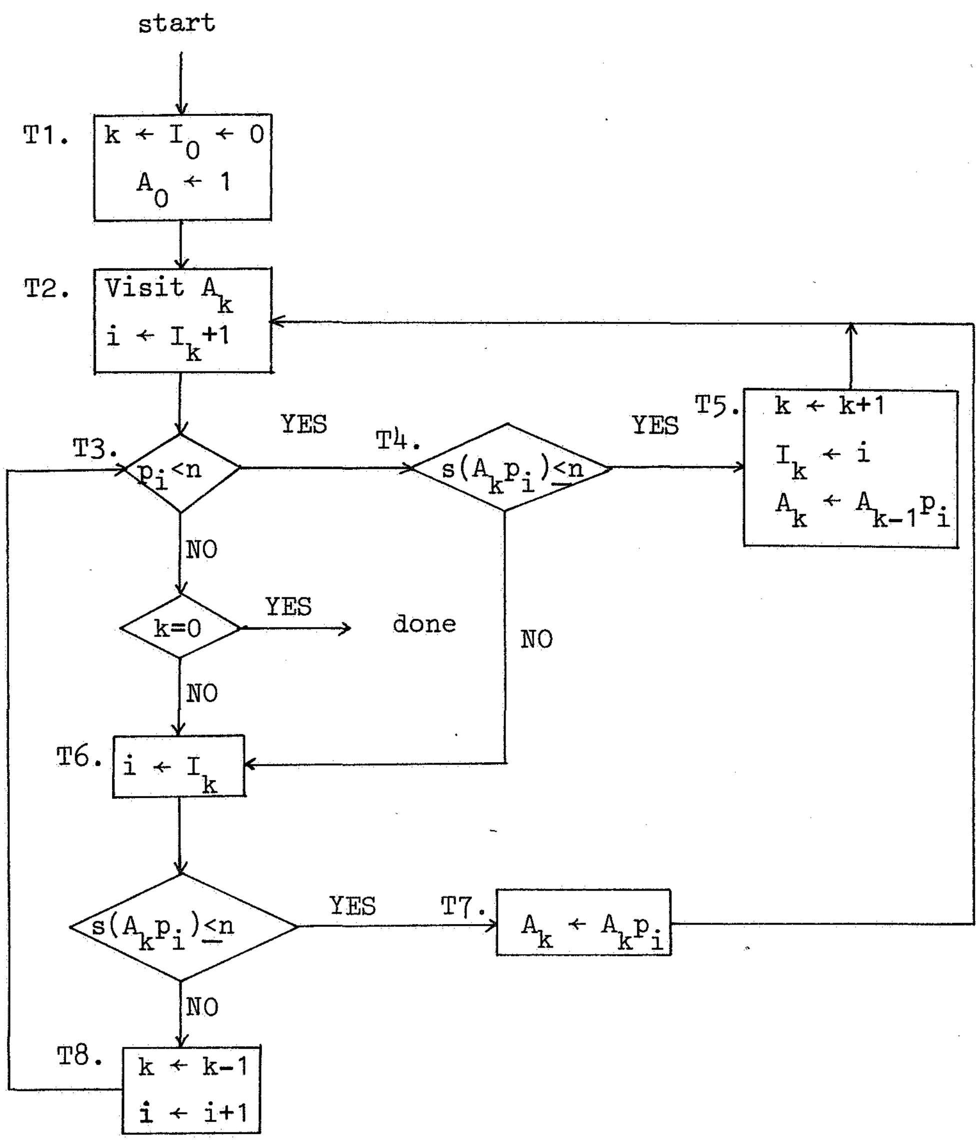


Figure 7.1. Flowchart of Algorithm T , which visits every node of the aliquot tree T[n] in the preorder sequence.

Algorithm T . (Traverse T[n] in preorder.) Let $n > 1$. This algorithm traverses the aliquot tree of n in preorder; that is, it visits every node of T[n] in the preorder sequence. Variable k equals the current level number and stack A contains items such that A[j] is a son of A[j-1] for $1 \le j \le k$. Stack I corresponds to A in the sense that p[I[j]] is the largest prime factor of A[j] .

T1. [Initialize.] Set $k \leftarrow I[0] \leftarrow 0$ and $A[0] \leftarrow 1$.

T2. [Visit A[k] and save index to next prime.] Visit A[k] and set $i \leftarrow I[k] + 1$.

T3. [Terminate?] If $p[i] < n$, then go to step T4. If $k = 0$, then terminate; otherwise go to step T6.

T4. [Does node A[k] have another son?] If $s(A[k]p[i]) > n$, then go to step T6.

T5. [Node A[k]p[i] is a son of A[k].] Set $k \leftarrow k+1$, $I[k] \leftarrow i$, $A[k] \leftarrow A[k-1]p[i]$, and go to step T2.

T6. [Does node A[k-1] have another son?] Set $i \leftarrow I[k]$. If $s(A[k]p[I]) > n$, then go to step T8.

T7. [Node A[k]p[i] is a brother of A[k].] Set $A[k] \leftarrow A[k]p[i]$ and go to step T2.

T8. [Backtrack.] Set $k \leftarrow k-1$, $i \leftarrow i+1$, and go to step T3.

Analysis of Algorithm T. We attempt to prove that Algorithm T traverses the $N > 0$ nodes of $T[n]$ in preorder by using induction on N. To motivate and simplify this correctness proof for Algorithm T, the following relatively straightforward assertion is offered without formal verification. (Remark: The flowchart of Algorithm T will be helpful in distinguishing the four cases of assertion A.)

A. Starting at step T6 with $A_k = mp_\alpha^e$, where $m \equiv A_{k-1} \geq 1$, $e \geq 1$, and $\alpha \equiv I_k > I_{k-1}$, the procedure of steps T2-T8 will either arrive at step T2 (case A1 or A2), step T6 (case A3), or terminate (case A4). In all cases, the items $A_0, \ldots, A_{k-1}$, $I_0, \ldots, I_{k-1}$ remain unchanged. The state of affairs for each case are:

A1. $A_k = mp_\alpha^{e+1}$ and $s(A_k) \leq n$ (which implies $p_\alpha < n$); that is A_k is the "next" son of A_{k-1} after mp_α^e.

A2. $A_k = mp_{\alpha+1}$, $s(A_k) \leq n$, $p_{\alpha+1} < n$, $s(mp_\alpha^{e+1}) > n$, and $I_k = \alpha + 1$; that is, A_k is the "next" son of A_{k-1} after mp_α^e.

A3. k is decreased by 1, $s(mp_\alpha^{e+1}) > n$, and ($p_{\alpha+1} \geq n$ or $s(mp_{\alpha+1}) > n$); that is, m has no more sons after mp_α^e.

A4. $k = 0$, $s(mp_\alpha^{e+1}) > n$, and $p_{\alpha+1} \geq n$; that is, k was originally 1 and $A_0 \equiv m = 1$ has no more sons after p_α^e.

If the reader will now attempt to play through Algorithm T beginning at step T6 with the above assumptions, he will easily arrive at each one of the four cases depending upon the tests at steps T3, T4, and T6: When control passes from step T6 to T7, case A1 obtains; otherwise, from step T6 we get to step T8 and then step T3, where either case A3 or A4 obtains, or else we reach step T5 and hence case A2 holds. These are the mutually disjoint and exhaustive possibilities.

Now our correctness proof is readily established if we can prove the slightly more general assertion:

"Starting at step T2 with $k \geq 0$ and $p[I[k]]$ the largest prime factor of the node $A[k]$ which is at level k of $T[n]$, the procedure of steps T2-T8 will traverse in preorder that subtree of $T[n]$ with $N > 0$ nodes whose root is $A[k]$, and will then arrive at step T6 (or terminate iff $k = 0$) with k returned to its original value and stack entries $A[0], \ldots, A[k], I[0], \ldots, I[k]$ unchanged".

This statement is obviously true when $N = 1$, because step T2 visits $A[k]$ and then we reach T6 since $p_i \geq n$ or $s(A[k]p_i) > n$ for all $i > I[k]$ when $A[k]$ has no sons. If $N > 1$, we first visit the root $A[k]$ at step T2 and it remains to show that each subtree defined by a son of $A[k]$ is visited in preorder. Clearly these subtrees must have $\leq N-1$ nodes, so the induction hypothesis ensures that they will be traversed in preorder if we successively enter step T2 with their ordered roots, the sons of $A[k]$. From visiting $A[k]$ we proceed via steps T3 and T4 to T5 because $A[k]$ has at least one son and its first son must in fact be $A[k]p[I[k]+1]$. At step T5 we store this son into $A[k+1]$ and set $I[k+1] = I[k] + 1$; next we go to step T2 where (using the induction hypothesis) the subtree defined by it is traversed; then we arrive at step T6 with k the index to the first son of our original root $A[k]$. Now assertion A is of use for it guarantees that all the ordered brothers of the first son will also reach step T2 in preorder (cases A1 and A2) until there are none remaining (case A3 or A4), at which time control reaches step T6 or terminates (iff $k = 0$). This completes the proof.

Step T1 clearly accomplishes the proper initialization so that the entire tree $T[n]$ would be traversed in preorder, according to the general assertion just proved.

Coding Algorithm T in a programming language is easy when subscript ranges for array p and stacks A , I are specified. The primes used pass test $p_i < n-1$ in step T3 except for one. Hence $i \leq \pi(n-1)+1$. Pointer k to stacks A and I never exceeds the highest level number of $T[n]$, equal to $\max_k \{k: s(p_1 p_2 \dots p_k) \leq n\}$.

Understanding and proving correctness of Algorithm T can both be enhanced by the elegance of a so-called "recursive solution" to traversing the nodes of $T[n]$ in preorder. To motivate a recursive statement of Algorithm T , we first clarify how trees can be represented and traversed recursively in preorder within a computer.

A common computer representation for a tree uses nodes which contain two links, a left link LLINK(P) pointing to the first son of NODE(P) and a right link RLINK(P) pointing to the next ordered brother of NODE(P) . A null link is denoted by Λ . Pictorially

P $\longrightarrow$

LLINK(P)	INFO(P)	RLINK(P)

NODE(P)

where, of course, INFO(P) contains the information in the tree node. See Figure 4.3 for the corresponding picture of the aliquot tree T[6] .

Using this representation for aliquot trees, the previously defined notion of traversing a tree in preorder can be restated more precisely by the following recursive procedure:

Algorithm TRAVERSE(P) .

T1. If $P = \Lambda$, then skip the next three steps (i.e., do nothing).

T2. "Visit" NODE(P) .

T3. TRAVERSE(LLINK(P)) .

T4. TRAVERSE(RLINK(P)) .

We next adapt the TRAVERSE algorithm to aliquot trees, using ALGOL 60 notation:

```
procedure T(A,i,e); value A,i,e; integer A,i,e;
begin integer y; if ¬ (A=1 ∧ p[i] ≥ n ∧ e=1) then
      begin y:= A×p[i]↑e; VISIT(y);
            if s(y×p[i+1]) ≤ n then T(y,i+1,1);
            if s(y×p[i]) ≤ n then T(A,i,e+1)
            else if s(A×p[i+1]) ≤ n then T(A,i+1,1)
      end
end;
```

The calling sequence is "VISIT(1);T(1,1,1)" to visit all the nodes of T[n] in preorder. When $n = 6$, the operation of procedure T proceeds in the following fashion:

T(1,1,1) ≡ VISIT(2);T(2,2,1);T(1,1,2)

T(2,2,1) ≡ VISIT(6)

T(1,1,2) ≡ VISIT(4);T(1,2,1)

T(1,2,1) ≡ VISIT(3);T(1,2,2)

T(1,2,2) ≡ VISIT(9);T(1,3,1)

$T(1,3,1) \equiv VISIT(5);T(1,3,2)$

$T(1,3,2) \equiv VISIT(25);T(1,4,1) \equiv VISIT(25)$

Hence, with $n = 6$, we have the desired result:

$$VISIT(1);T(1,1,1) \equiv VISIT(1);VISIT(2);VISIT(6);VISIT(4);VISIT(3); VISIT(9);VISIT(5);VISIT(25) .$$

A formal correctness proof that procedure T does indeed traverse T[n] in preorder would be based upon the following considerations: (1) Pointer P in the TRAVERSE Algorithm is replaced by the 3-tuple (A,i,e) corresponding to node $y = Ap_i^e$; (2) The initial conditional in procedure T ensures that traversal terminates at the first node $y = p_i \geq n$; (3) LLINK(P) in TRAVERSE points to the first son of node $y = Ap_i^e$, which is $Ap_i^e p_{i+1}$ <u>iff</u> $s(yp_{i+1}) \leq n$; (4) RLINK(P) in TRAVERSE points to the next, ordered brother of node $y = Ap_i^e$, which is either (i) Ap_i^{e+1} <u>iff</u> $s(yp_i) \leq n$, or else (ii) Ap_{i+1} <u>iff</u> $s(Ap_{i+1}) \leq n$; (5) Invoking T(1,1,1) starts traversal of T[n] at node $y = 2$; (6) Arguments for finiteness of T[n] and termination of traversal stated in the proof of Algorithm T apply also to procedure T .

Just as Algorithm R is a modification of Algorithm T which evaluates s values without factoring numbers, we can rewrite procedure T as procedure R to take advantage of the top-down locally-defined function s . Furthermore, to reduce the possibly large recursion depth of procedure T , two of the recursive calls of procedure T have been replaced in procedure R by iteration, so that procedure R clearly has a maximum recursion depth equal to

$$\max_k \{k: s(p_1 \, p_2 \, \ldots \, p_k) \leq n\} ,$$

which is the highest level number of T[n] . Because our aims in restating Algorithms T and R as procedures T and R are clearer expression and easier correctness proofs, go to statements have been avoided (the Boolean variable LOOP in procedure R is our mechanism for structuring the iteration therein without using undesirable jumps).

<u>Procedure</u> R . (Procedure T with recurrence relations to evaluate values of s and with two recursive calls replaced by iteration.)

```
procedure R(A,i,e,sA); value A,i,e,sA; integer A,i,e,sA;
begin integer y,sy; Boolean LOOP;
LOOP:= true;
for i:= i while LOOP ∧ ¬(A=1 ∧ p[i] ≥ n ∧ e=1) do
    begin y:= A×p[i]↑e; VISIT(y);
          sy:= sA×TABLE[i,e+1] + A×TABLE[i,e];
          if sy×TABLE[i+1,2] + y ≤ n then R(y,i+1,1,sy);
          if sA×TABLE[i,e+2] + A×TABLE[i,e+1] ≤ n then e:= e+1
          else if sA×TABLE[i+1,2] + A ≤ n then
               begin i:= i+1; e:= 1 end
               else LOOP:= false;
    end
end;

comment Array element TABLE[i,e] equals s(p[i]↑e) and could be
        replaced by a procedure TABLE(i,e) that computes
        1 + p[i] + p[i]↑2 + ... + p[i]↑(e-1) . Formal parameter
        sA and variable sy have values s(A) and s(y) ,
        respectively. The calling sequence "VISIT(1), R(1,1,1)"
        will traverse T[n] in preorder sequence;
```

Figure 7.2. Profile of Algorithms T and R. The unknowns α, β, γ have the following characteristics:

α = Number of nodes in T[n] ; β = Number of "node groups" in T[n] ; γ = Number of nodes in T[n] which are divisible by the largest prime less than n .

Step	Times each step is executed for given n.				
	n = 13	50	500	5000	general
T1,R1	1	1	1	1	1
T2,R2	19	114	3157	134550	α
T3,R3	30	203	6160	268077	$\alpha+\beta$
T4,R4	27	198	6157	268074	$\alpha+\beta-\gamma-1$
T5,R5	11	89	3003	133527	β
T6,R6	18	113	3156	134549	$\alpha-1$
T7,R7	7	24	153	1022	$\alpha-\beta-1$
T8,R8	11	89	3003	133527	β

The above profile was derived as follows. Firstly, with step Ti (Ri) being executed x_i times, the eight unknowns $(x_1, \ldots, x_8)$ were reduced by application of "Kirchoff's" conservation law for flowcharts (Knuth 1968, section 2.3.4.1). This yielded:

Step	Times	Step	Times
T1,R1	1	T5,R5	x_2-x_7-1
T2,R2	x_2	T6,T6	x_7+x_8
T3,R3	x_2+x_8	T7,R7	x_7
T4,R4	x_4	T8,R8	x_8

Next, it follows that $x_5 = x_8 = x_2 - x_7 - 1$ because k is initialized to zero (step T1) and then the algorithm terminates only when $k = 0$. Thus for every time k is increased by one in step T5, k must be decreased by one in step T8. There remain three unknowns and these can be interpreted by relating them to pertinent characteristics of the aliquot tree of n. Let

α = number of nodes in $T[n]$

β = number of "node groups" in $T[n]$

γ = number of nodes in $T[n]$ which are divisible by the largest prime less than n.

Two nodes $p_{i_1}^{e_1} \ldots p_{i_k}^{e_k}$ and $p_{j_1}^{f_1} \ldots p_{j_r}^{f_r}$ belong to the same "node group" if and only if $k = r$, $i_t = j_t$ for $1 \leq t \leq k$, and $1 \leq t \leq k-1$. (Thus they differ only in their last exponents e_k and f_k.) The root 1 is not considered part of a node group.

For example, we have $\alpha = 19$ nodes, $\beta = 11$ node groups, and $\gamma = 2$ (for the two nodes 11 and 11^2) in the aliquot tree $T[13]$ of Figure 4.1.

Step T5 is clearly executed once for each node group in $T[n]$. Hence $x_5 = \beta$.

Step T2 visits every node of $T[n]$ precisely once. Hence $x_2 = \alpha$.

Step T4 is entered only when $p_i < n$ in the test of step T3. Further, step T3 is performed $x_2 + x_8 = \alpha + \beta$ times so that $x_4 = \alpha + \beta - y$, where y is the number of times that $p_i \geq n$ in step T3. Obviously, the only time that $p_i \geq n$ obtains is when the last node visited has a factor equal to the largest prime less than n . Hence $y = \gamma + 1 = \alpha + \beta - x_4$. (There is one extra test where $p_i \geq n$ and $k = 0$.)

We remark on the behaviour of the quantities α, β, and γ as n increases. The quantity γ is obviously very small; indeed, when $n-1$ is prime, $\gamma = 2$. The quantity $\alpha-\beta$ seems to grow as $n^{0.815}$, which predicts observed values within relative error 3%. Finally, β increases a little faster than $0.2\, n^{1.6}$, so that Algorithms R and T would perform about 10^7 steps to handle the case $n = 50000$.

<u>Algorithm</u> R . (Algorithm T with recurrence relations to evaluate values of s .) Like Algorithm T , for input $n > 1$ the aliquot tree T[n] is traversed in preorder. In addition, calculation of s-values at each node is speeded up by using a table of values $s(p_i^e)$, a stack E whose item E[j] , for $0 \le j \le k$, corresponds to the exponent of factor p[I[j]] in A[j] , and a stack S with S[j] = s(A[j]) .

R1. [Initialize.] Generate entries of TABLE[i,e] = $s(p_i^e)$ for all $p_i \le n$ and $s(p_i^{e-2}) \le n$. Set S[0] ← k ← I[0] ← 0 and A[0] ← 1 .

R2. [Visit A[k] and save index to next prime.] Visit A[k] (Note S[k] = s(A[k])) and set i ← I[k] + 1 .

R3. [Terminate?] If p[i] < n , then go to step R4. If k = 0 , then terminate; otherwise go to step R6.

R4. [Does node A[k] have another son?] Set t ← A[k] + + S[k].TABLE[i,2] . If t > n , then go to step R6.

R5. [Node A[k]p[i] is a son of A[k].] Set k ← k+1 , I[k] ← i , E[k] ← 1 , A[k] ← A[k-1]p[i] , S[k] ← t , and go to step R2.

R6. [Does node A[k-1] have another son?] Set i ← I[k] , e ← E[k] + 1 , and t ← S[k-1].TABLE[i,e+1] + + A[k-1].TABLE[i,e] . If t > n , then go to step R8.

R7. [Node A[k]p[i] is a brother of A[k].] Set E[k] ← e , A[k] ← A[k]p[i] , S[k] ← t , and to step R2.

R8. [Backtrack.] Set k ← k-1 , i ← i+1 , and go to step R3.

Analysis of Algorithm R . Because Algorithm R is one-to-one with Algorithm T we will only show that steps R4 and R5 evaluate values of s correctly. First, at step R4 we have

$$\begin{aligned} t &= A[k] + S[k].TABLE[i,2] \\ &= A[k] + s(A[k])\ s(p_i^2) \\ &= (1+p_i)\ s(A[k]) + A[k] \\ &= s(a[k]p_i)\ , \end{aligned}$$

by application of Corollary 1.2. Using Corollary 1.1 and the relation

$$A[j] = A[j-1]p[I[j]]^{E[j]} \qquad \text{for } 1 \le j \le k\ ,$$

at step R6 yields

$$\begin{aligned} t &= S[k-1].TABLE[i,e+1] + A[k-1].TABLE[i,e] \\ &= s(A[k-1])\ s(p_i^{e+1}) + A[k-1]\ s(p_i^e) \\ &= s(A[k-1]p_i^e) = s(A[k]p_i)\ . \end{aligned}$$

Memory requirements for array TABLE increase rapidly with n . A space saving alternate approach is to make TABLE into a subroutine with two arguments (i,e) that computes

$$\begin{aligned} s(p_i^e) &= 1 + p + p^2 + \ldots + p^{e-1} \\ &= 1 + p(1+p(1+\ldots p))\ . \end{aligned}$$

Stacks E and S require the same storage as stack A , except E[0] is never referenced.

<u>Algorithm</u> E . (Examine aliquot series for cycles.) Let $N \geq n \geq 0$. This algorithm examines and detects cycles in every aliquot series with leader $\leq n$ and with terms $\leq N$. List A with index k serves to save the series terms, while i and x are the current series leader and term, respectively.

E1. [Initialize.] Set $i \leftarrow -1$.

E2. [Done?] Set $i \leftarrow i+1$. If $i > n$, then terminate; otherwise set $x \leftarrow i$, $k \leftarrow 1$, and $A[1] \leftarrow x$.

E3. [Series terminates?] If $s(x) = 1$ or $s(x) > N$ or $s(x) < n$, then go to step E2.

E4. [Cycle?] If $s(x) \notin \{A[j]: 1 \leq j \leq k\}$, then go to step E5. Otherwise, a cycle is captured in the A list; if $s(x) = A[j]$, then $(A[j], A[j+1], \ldots, A[k])$ is a cycle of length $k-j+1$ with terms $\leq N$. Go to step E2.

E5. [Move along series.] Set $x \leftarrow s(x)$, $k \leftarrow k+1$, $A[k] \leftarrow x$, and go to step E3.

<u>Algorithm</u> H . (Search for cycles and keep a history.) Given $N \geq n \geq 0$ this algorithm gives the same output as Algorithm E , except it keeps a history in the Boolean list B of which numbers have been previously encountered in a series, so that no series or subseries is visited more than once.

H1. [Initialize.] Set $B[i] \leftarrow$ "false" for $0 \leq i \leq N$. Set $i \leftarrow -1$.

H2. [Done?] Set $i \leftarrow i+1$. If $i > n$, then terminate; otherwise set $x \leftarrow i$ and initialize the A list to x .

H3. [Previous series?] If $B[x] =$ "true" , then go to step H2.

H4. [Series terminates?] If $s(x) > N$, then set $B[x] \leftarrow$ "true" and go to step H2.

H5. [Cycle detected?] If $s(x)$ not in A list, then go to step H6. Otherwise, a cycle is captured in the A list; set $B[x] \leftarrow$ "true" and go to step H2.

H6. [Move along series.] Set $B[x] \leftarrow$ "true" , $x \leftarrow s(x)$, add x to the A list, and go to step H3.

<u>Algorithm</u> D . (Detect cycles <u>after</u> computing and saving s-values.) For inputs $N \geq n \geq 0$, this algorithm produces exactly the same output as Algorithm H . The difference is that it computes and saves all necessary s-values in array S before seeking cycles. Remark: Marking is to be idempotent; that is, marking a marked element of S simply leaves it as originally marked.

D1. [Initialize.] For $0 \leq i \leq N$, set $S[i] \leftarrow s(i)$; if $s(i) = 1$ or $s(i) > N$, then set $S[i] \leftarrow 0$. Each S entry is assumed initially unmarked. Set $i \leftarrow 0$ and output trivial cycle (0) .

D2. [Done?] Set $i \leftarrow i+1$. If $i > n$, then terminate. Otherwise set $k \leftarrow i$ and mark S[k] .

D3. [Delete series?] If $S[k] \neq 0$, then go to step D4. Otherwise, delete cycle candidate series i, S[i], S[S[i]], ..., k by setting their S entries to zero; return to step D2.

D4. [Cycle detected?] If S[S[k]] is not marked, then go to step D5. Otherwise, output the cycle (S[S[k]], S[S[S[k]]], ..., k); then set $S[k] \leftarrow 0$ and go to step D3.

D5. [Mark S entry.] Mark S[k] , set $k \leftarrow S[k]$, and go to step D3.

Analysis of Algorithm D . The method of frequency counts has been applied to Algorithm D in order to determine the number of times each step was actually performed for various inputs.
Table 7.3 is the resultant profile (collection of frequency counts) of Algorithm D for the case where N = n . The column headed "Times" represents the number of times the corresponding step will be executed during the course of the algorithm.

From the profile of Figure 7.3, it is clear that once the S array has been set up (step D1), the running time of Algorithm D is proportional to n when N = n .

Figure 7.3. Profile of Algorithm D when N = n . The unknown α and β have the following important characteristics: α = Number of cycles outputed; β = Number of zero entries in list S after step D1 is performed.

Step	Times each step is executed for given N = n .				
N = n =	10	100	1000	10000	52000	general
D1	1	1	1	1	1	1
D2	11	101	1001	10001	52001	n+1
D3	15	167	1766	18160	95452	$2n-\beta$
D4	5	67	766	8160	43452	$n-\beta$
D5	4	65	762	8151	43439	$n-\beta-\alpha+1$
	2	3	5	10	14	α
	5	33	234	1840	8548	β

References

Alanen, J.D., O. Ore, and J. Stemple, 1967. Systematic computations on amicable numbers. Mathematics of Computation **21** 242-245.

Benedetti, C., 1967. Ricerche statistiche sui numeri primi. Metron **26** (3-4) 237-313.

Borho, W., 1969. Über die Fixpunkte der k-fach iterierten Teilersummenfunktion. Mitteilungen der Mathematischen Gesellschaft in Hamburg **4** 35-48.

Bouniakowsky, V., 1850. Mem. Ac. Sc. St. Petersbourg (Sc. Math. Phys.) **6** 259-295.

Bratley, P., and J. McKay, 1968. More amicable numbers. Mathematics of Computation **22** 677-678.

Buxton, J.N., and B. Randell, 1970. Software Engineering Techniques NATO Science Committee Report 1-164.

Catalan, M.E., 1887-8. Propositions et questions diverses. Bulletin de la Société Mathématique de France **16** 128-9.

Cohen, H., 1970. On amicable and sociable numbers. Mathematics of Computation **24** 423-429.

Dickson, L.E., 1913. Theorems and tables on the sum of the divisors of a number. Quart. J. Pure Appl. Math. **44** 264-296.

Dickson, L.E., 1919. History of the Theory of Numbers **1** Chelsea Publishing Company, New York.

Feller, W., 1957. An Introduction to Probability Theory and its Applications **1** (2nd edition) Wiley.

Gardner, M., 1968. Mathematical games: a short treatise on the useless elegance of perfect numbers and amicable pairs. Scientific American March 121-124.

Grune, D., 1970. Manual of the Milli-system for the Electrologica X8. Mathematical Centre report MC-LR 1/70, Amsterdam (in Dutch).

Guy, R.K. and J.L. Selfridge, 1971. Interim report on aliquot series. The University of Calgary. research paper no. 141.

Hardy, G.H., and E.M. Wright, 1960. An Introduction to the Theory of Numbers Oxford University Press, New York.

Kac, M., 1959. Statistical Independence in Probability, Analysis and Number Theory The Mathematical Association of America.

Knuth, D.E., 1968. The Art of Computer Programming **1** Addison-Wesley Publishing Company.

Knuth, D.E., 1969. The Art of Computer Programming **2** Addison-Wesley Publishing Company.

Knuth, D.E., 1971. Mathematical Analysis of Algorithms. Stanford University report STAN-CS-71-206.

Lee, E.J., 1969. The discovery of amicable numbers. J. Recreational Math. to appear.

McCarthy, P.J., 1957. Odd perfect numbers. Scripta Math. **23** 43-47.

Meissner, O., 1907. Über einige zahlentheoretische Funktionen. Archiv. Math. Phys. **3** 199.

Muskat, J.B., 1966. On divisors of odd perfect numbers. Mathematics of Computation **20** 141-144.

Ore, O., 1948. Number Theory and Its History. Mc-Graw-Hill Book Company, Inc.

Ore, O., 1965. Theory of Graphs **38** American Mathematical Society Colloquium Publications.

Poulet, P., 1918. Note 4865. L'intermédiaire des mathématiques **25** 100-101.

Poulet, P., 1929. La Chasse aux Nombres **1** Stevens, Brussels.

Riele, H. te, 1972. A note on the Catalan-Dickson conjecture. Mathematics of Computation (to appear).

Rolf, H.L., 1967. Friendly numbers. The Mathematics Teacher **60** 157-160.

Savage, L.J., 1968. Statistical inference. In The Future of Statistics (editor: D.G. Watts) Academic Press 139-160.

Shen, M., 1964. On checking the Goldbach conjecture. BIT **4** 243-245.

Stein, M.L., and P.R. Stein, 1965. New experimental results on the Goldbach conjecture. Mathematics Magazine **38** 72-80.

Tuckerman, B., 1971. The 24th Mersenne prime. Notices of the AMS **18** (4) 608.