PRIME DECOMPOSITIONS WITH MINIMUM SUM

by

Ole Østerby

November 1973

DAIMI PB-19

Matematisk Institut Aarhus Universitet
DATALOGISK AFDELING

Prime decompositions with minimum sum

1. Introduction.

During the January 1972 meeting in Aarhus on Automata Theory a number-theoretic problem was posed in connection with determining the size of a language generated by a DOL-system (Deterministic Lindenmayer-system with no interaction between neighbours).

The problem as posed by P. Vitanyi [1] is:

Find an algorithm which for every integer $n > 0$ yields a number of pairwise relative primes $k_1, k_2, \dots, k_m$ and a non-negative integer r such that

$$(1) \qquad \prod_{i=1}^{m} k_i + r = n ,$$

and

$$(2) \qquad \sum_{i=1}^{m} k_i + r = s$$

is minimal.

An accompanying problem is the same as the above except that (1) is replaced by

$$(1^*) \qquad \prod_{i=1}^{m} k_i + r \geq n .$$

We have solved the first problem and shall in section 3 describe how and give examples of minimal decompositions of certain integers. We shall also give some comments on the solution of the second problem.

2. Preliminary remarks.

A few simple observations indicate how to limit the search. Firstly, although the only requirement for the k_i is that they be relatively prime (g. c. d. $(k_i, k_j) = 1$ if $i \neq j$), it is sufficient to consider powers of primes as factors, i. e. $k_i = p_i^{\,j}$ ($j \geq 1$) and $p_i \neq p_j$ if $i \neq j$. To see this, notice that for any two different primes p_i and p_j $p_i + p_j < p_i \cdot p_j$ such that a smaller value of the sum can be obtained by splitting a composite k_i into its prime factors.

Remark: The number 1 does not count as a prime in these considerations, since ones among the k_i only add to the sum without enlarging the product, and therefore a minimal sum can never contain a k_i which is equal to 1. We shall in the following always assume that $k_i \neq 1$.

An upper bound on m , the number of factors in a decomposition, can be obtained easily, by noting that the smallest number n for which a decomposition can contain m factors is the product of the first m primes.

For easy reference we supply in Table 1 a table of the first primes, their products, and sums.

Tabel 1

m	p_m	$n = \prod_1^m p_i$	$s = \sum_1^m p_i$
1	2	2	2
2	3	6	5
3	5	30	10
4	7	210	17
5	11	2310	28
6	13	30 030	41
7	17	510 510	58
8	19	9 699 690	77
9	23	223 092 870	100
10	29	6 469 693 230	129
11	31	200 560 490 130	160
12	37	7 420 738 134 810	197

Based on this information a method of finding s for a given n can be devised by finding the maximal number m_n of factors and then look at all combinations of at most m_n factors, each one the power of a prime. Also, once a decomposition has been found with a corresponding sum σ, then there is no need to consider primes (or prime powers) that are larger than σ.

This method will probably not be efficient for large n and a much simpler and straightforward method was suggested by mr. A. H. Andersen in an informal discussion, and we shall now present this method.

*

Bemærk at $2 \cdot 3 \cdot 23 = 138 < 144 = 16 \cdot 9$

hvorfor $n_{max}(100) = 232\ 792\ 560 > 223\ 092\ 870$

3. An effective method to finding the minimal decomposition of a number n.

Compute the prime decomposition of n and also compute the corresponding sum:

(3) $$n = \prod_{i=1}^{\mu} p_i^{j_i} \; ; \quad \sigma^{(0)} = \sum_{i=1}^{\mu} p_i^{j_i} \; .$$

Next subtract 1 from n , compute the prime decomposition of n−1 and the corresponding sum:

(4) $$n = \prod_{i=1}^{\mu} p_i^{j_i} + 1; \quad \sigma^{(1)} = \sum_{i=1}^{\mu} p_i^{j_i} + 1 \; .$$

(There should also have been superscripts (0) and (1) on μ, p_i, and j_i in formulae (3) and (4) but they have been left out for sake of readability).

Formula (4) represents another decomposition of n .

Now subtract 2 from n , compute the prime decomposition of n−2 and repeat the process.

The general step consists of finding the prime decomposition of n−t ($t \geq 0$) such that we have

(5) $$n = \prod_{i=1}^{\mu} p_i^{j_i} + t; \quad \sigma^{(t)} = \sum_{i=1}^{\mu} p_i^{j_i} + t \; .$$

If s is the smallest of $\sigma^{(0)}, \sigma^{(1)}, \ldots, \sigma^{(t)}$, i.e. the best value we have obtained so far then there is no need to continue the process further down than to n − s (and it is not even necessary to go this far) and we can thus terminate the process rather quickly even for large n . All we have to do then is to record the decomposition corresponding to the smallest σ .

If the object is to compute a table of minimal sums corresponding to a certain integer interval then it is not necessary to repeat all of this for each individual n. Once the process has been started and a minimal decomposition has been found for a certain n with sum $s(n)$, then in order to proceed to $n + 1$ it is only necessary to compute the prime decomposition of $n + 1$ and compare the corresponding sum with $s(n) + 1$.

4. Properties of $s(n)$.

We shall in the following use the notation $s(n)$ for the sum corresponding to the minimal decomposition of n.

When computing a table of $s(n)$ it is seen that the minimal decompositions for a sequence of consecutive integers will form groups, each group built upon a certain prime decomposition and with a remainder which increases from zero in steps of one. This behaviour which becomes very expressed for large n provides a very crude upper bound for the growth of $s(n)$. Although $s(n)$ usually grows with slope 1, it drops every once in a while to a lower value and starts the climb again. (See fig. 2 and 3.)

Thus a table of $s(n)$ versus n can be somewhat compressed by only giving the values of n and $s(n)$ each time $s(n)$ drops. Our computer program also supplies the – redundant but useful – information on $s(n)$ just before the drop.

The function $s(n)$ is not at all monotone but it is possible to define two monotone functions related to $s(n)$: the infimum of all monotone functions larger than or equal to $s(n)$, and the supremum of all monotone functions smaller than or equal to $s(n)$. For lack of better names we have called these functions $s^{(1)}$ and $s^{(2)}$ respectively. In Table 4 we have supplied a table of the decompositions of the first 40 integers together with

$s(n)$, $s^{(1)}$, and $s^{(2)}$.

We shall now take a look at the inverse functions of $s^{(1)}$ and $s^{(2)}$ which we call n_{min} and n_{max} respectively.

Let us by $n_{max}(s)$ denote the largest n such that $s(n) = s$. Some values of this function are very easy to find, since it is easy to see, that decompositions involving the small prime numbers (with exponents 1) are very 'good' in the sense that the product n is large compared to the sum s. Therefore, some values of $n_{max}(s)$ can be found from Table 1 and others can be derived from these.

On the other hand, it is rather difficult to compute $n_{max}(s)$ systematically for a given s-interval, the reason being that you 'never' know when a fairly large n will produce a decomposition with a small s. ~~The only~~ way out is to perform computations for a large n-interval, the bounds to be estimated from the values given in table 1.

The other function which gives the smallest n such that $s(n) = s$ is called $n_{min}(s)$. Contrary to n_{max} it is virtually impossible to find a single value of n_{min} for a given s, but it is a straightforward task to compute $n_{min}(s)$ systematically for an interval of the form $1 \le s \le s_0$: You just compute $s(n)$ from $n = 1$ and up until for some n you hit the value $s(n) = s_0$.

A table of n_{min} and n_{max} for $1 \le s \le 70$ is found as Table 5. ~~This~~ table reflects some of the difficulty in computing a full table of n_{max} - we have not invested the computing power necessary to fill in the gaps.

5. Vitanyi's two problems

Vitanyi's first problem is solved by the algorithm which we described in section 3 and which is implemented in the programs in section 7. As for the second problem the minimal sum corresponding to a number n is the value of $s^{(2)}(n)$. In order to find the corresponding decomposition, i.e.

the numbers k_i and r, we must find the largest number q such that $s^{(2)}(q) = s^{(2)}(n)$ and take the decomposition of q which is the one we need; q can be found easily by using a table of n_{max} in the following way: Given n, find two consecutive values of $n_{max}(s)$ such that n is an intermediate number. q is now the larger of the numbers and the corresponding entry s is the minimal sum.

This does not really solve the second problem as it is stated by Vitanyi in a completely satisfactory manner because of the difficulties which we mentioned in section 5 of producing a table of n_{max}. On the other hand, given Table 1 and enough computing power we have here a terminating algorithm for solving the second problem.

In his paper Vitanyi also mentions the problem of, for a given s^*, finding a decomposition such that

$$(6) \qquad s^* = \sum_{i=1}^{m} k_i + r$$

and $n = \prod_{i=1}^{m} k_i + r$ is maximal.

The corresponding n is clearly the value of n_{max} at s^*. An accompanying problem with (6) replaced by

$$(6^*) \qquad s^* \geq \sum_{i=1}^{m} k_i + r$$

yields nothing new, since a solution necessarily must have equality in (6^*) for otherwise a larger n can be obtained by increasing r in order to force equality in (6^*).

6. Minimal decompositions are not unique.

As can be seen from Table 4 it is not correct to talk about the minimal decomposition of a number since there may be more than one. If we exclude from consideration the numbers $n \le 5$ for which only rather trivial decompositions are possible, the number 14 is the first one which possesses two distinct minimal decompositions:

$$14 = 4 \cdot 3 + 2 = 2 \cdot 7 \; ; \quad s(14) = 9 \; .$$

Another example is $18 = 3 \cdot 5 + 3 = 2 \cdot 9$; $s(18) = 11$. These are not unique examples: In the range $6 \le n \le 1\,000\,000$ there are 671 groups totaling 15885 integers which have at least two minimal decompositions.

The number 39 is an example of a number with three minimal decompositions. There are 15 groups totaling 229 numbers with at least three minimal decompositions in the range $6 \le n \le 1\,000\,000$. These are given in Table 6.

Four minimal decompositions is the most that we have encountered and there is only one such group consisting of 7 numbers. The group begins with

$$\begin{aligned} 64782 &= 2 \cdot 9 \cdot 59 \cdot 61 = 4 \cdot 5 \cdot 41 \cdot 79 + 2 = 2 \cdot 13 \cdot 47 \cdot 53 + 16 \\ &= 8 \cdot 9 \cdot 29 \cdot 31 + 54 \; ; \quad s(64782) = 131 \; . \end{aligned}$$

7. ALGOL realizations of the method.

Program 1.

Find s(n) and the minimal decomposition (with smallest r) for a given n .

The strategy of this program when searching for the prime decomposition of a number is to attempt to divide the number first by 2 and then by all odd integers from 3 and up to the smallest value of σ recorded so far. This gives a short program but a longer running time than if we supply a table of primes.

The dimension 1:11 of the arrays allows values of n up to 7 420 738 134 809.

When the program stops, min will contain s(n) and the factors k_i are found in B_1 to $B_{\cancel{k}\,m}$.

```
begin integer a, b, j, k, m, n, nu, r, min, diff, sum;
    integer array B, C[1:11];
    INPUT(n);
    min := n + 1;
    for diff := 0 step 1 until min do
    begin nu := n - diff; k := 1; a := 2;
        for j := 1 while nu > 1 do
        begin for b := nu÷a while b×a = nu do
                begin j := j×a; nu := b end;
            if j > 1 then begin C[k] := j; k := k + 1 end;
            if a > min then nu := 0;
            if a = 2 then a := 3 else a := a + 2
        end;
        if nu > 0 then
        begin sum := 0; k := k - 1;
            for j := k step -1 until 1 do sum := C[j] + sum;
            sum := sum + diff; if k = 0 then sum := n;
            if sum < min then
            begin for j := 1 step 1 until k do B[j] := C[j];
                min := sum; m := k; r := diff
            end
        end
    end;
    comment s(n) is now contained in min and the factors can be
            found in the first m positions of the array B;
end
```

comment by introducing the statement min:=n+1 instead of min:=n at the beginning, the program will work also for $1 \leq n \leq 5$. For this interval only trivial decompositions exist, i.e. either $m = 0$, $r = n$ or $m = 1$, k_i arbitrary in the interval $1 \leq k_1 \leq n$, and $r = n - k_1$. The decompositions with smallest r are thus

$$m = 1, \; k_i = n, \; r = 0 .$$

For $n = 1$ the program will give $m = 0$, $r = 1$, though.

Program 2.

Find $s(n)$ for $p \leq n \leq q$ where p and q are given integers.

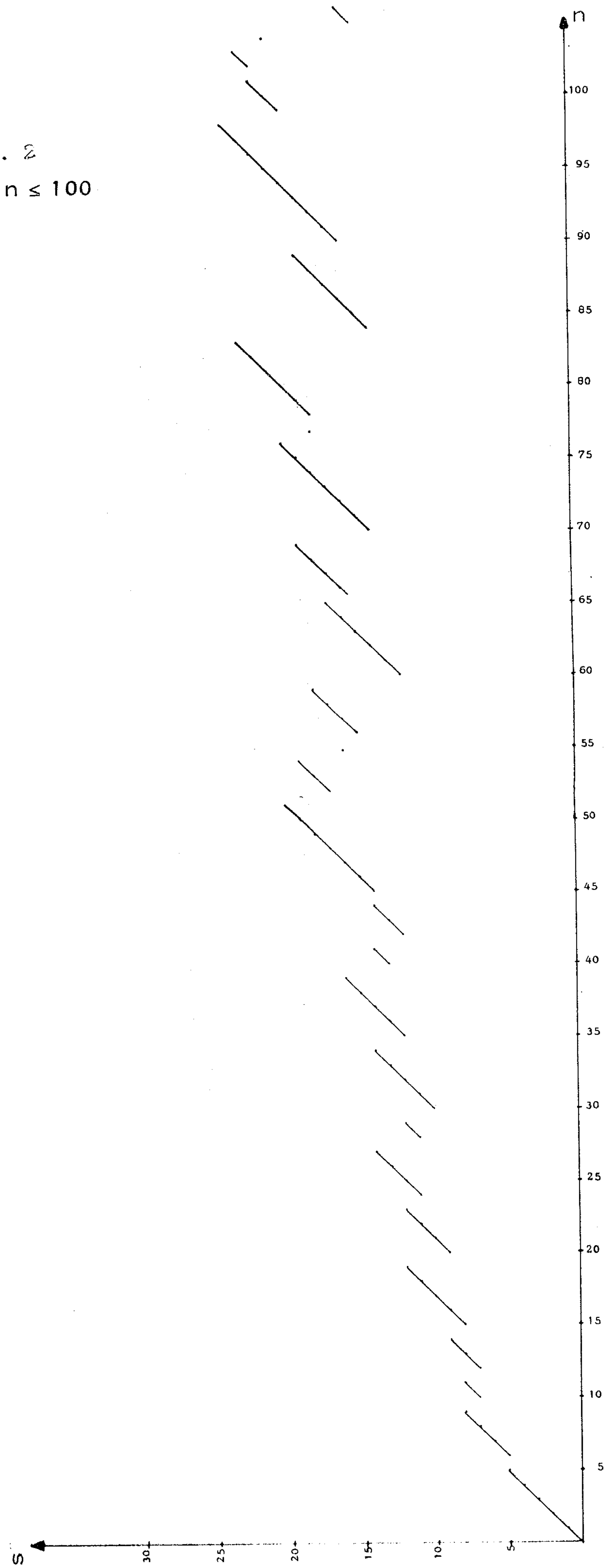
In this program we have used the alternate approach of supplying a table of primes for use in the prime decomposition of n. The range is again $1 \leq p \leq q \leq 7\,420\,738\,134\,809$ provided the table of primes is large enough. It is believed that no primes larger than 3001 will be needed here.

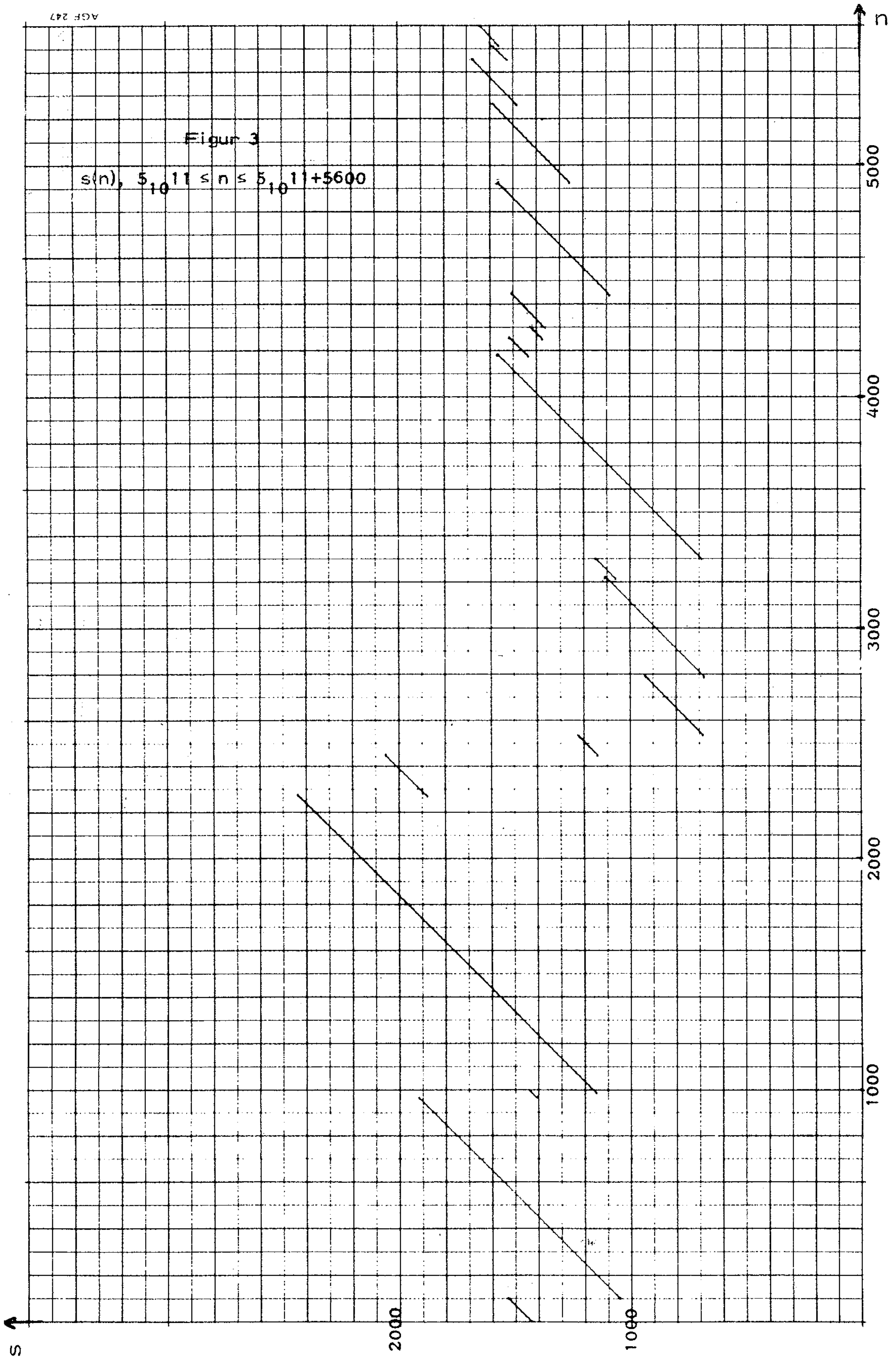
```
begin integer a, b, i, j, k, m, n, nu, min, diff, sum, p, 1, r;
    integer array P[1:432], B, C[1:11];
    i := 1;
    for j := 2, 3, 5, 7, 11, ...., 3001, 100 000 000 do
        begin P[i] := j; i := i + 1 end;
    comment P now contains a table of primes. The four dots
            represent 425 interjacent primes and the number
            100 000 000 is used as a safeguard;
    INPUT(p); INPUT(q);
    min := 3000; if min > p then min := p + 1;
    comment necessary if p happens to be a prime;
    if q ≥ p then
    for diff := 0 step 1 until min do
    begin nu := p - diff; i := k := 1; a := 2;
        for j := 1 while nu > 1 do
        begin for b := nu÷a while b×a = nu do
                  begin j := j×a; nu := b end;
              if j > 1 then begin C[k] := j; k := k + 1 end;
              if a > min then nu := 0;
              i := i + 1; a := P[i]
        end;
```

```
if nu > 0 then
begin sum := 0; k := k - 1;
      for j := 1 step 1 until k do sum := C[j] + sum;
      sum := sum + diff; if k = 0 then sum := p;
      if sum < min then
      begin for j := 1 step 1 until k do B[j] := C[j];
            min := sum; m := k; r := diff
      end
end
end this is the end of the initial stage, that of finding a minimal
   decomposition of p (and of all numbers down to p - r as
   well). This was merely a repetition of Program 1;
for n := p + 1 step 1 until q do
begin nu := n; i := k := 1; a := 2;
      r := r + 1; min := min + 1; sum := 0;
      for j := 1 while nu > 1 do
      begin for b := nu÷a while b×a = nu do
                begin j := j×a; nu := b end;
            if j > 1 then
            begin C[k] := j; sum := C[k] + sum; k := k + 1 end;
            if a > min then nu := 0;
            i := i + 1; a := P[i]
      end if nu = 1 now, then we have found a prime decomposition
            of n , and the corresponding sum must be compared to
            min. Otherwise nu = 0 and we have decided to stop further
            searching for a such decomposition since the primes
            necessary are larger than min;
```

```
if nu > 0 then
begin if sum ≤ min then
          begin comment output of the minimal decompo-
                sition of n and s(n) , possibly preceded
                by that of n - 1 and s(n-1), should be
                placed here.
                This will give a condensed table of s(n)
                providing only values where s(n) drops
                or - in the case sum = min - where a
                different minimal decomposition is found;
                min := sum; r := 0
          end
end
     end
end
```

Fig. 2
s(n), 1≤ n ≤ 100




AGF 247
Figur 3
s(n), 5 10 11 ≤ n ≤ 5 10 11+5600
n
5000
4000
3000
2000
1000
s
2000
1000

n	k_i			r	$s(n)$	$s^{(1)}$	$s^{(2)}$
1	1				1	1	1
2	2				2	2	2
3	3				3	3	3
4	4				4	4	4
5	5				5	5	5
6	2	3			5	5	5
7	2	3		1	6	6	6
8	2	3		2	7	7	7
9	2	3		3	8	8	7
10	2	5			7	8	7
11	2	5		1	8	8	7
12	3	4			7	8	7
13	3	4		1	8	8	8
14	2	7			9	9	8
15	3	5			8	9	8
16	3	5		1	9	9	9
17	3	5		2	10	10	9
18	2	9			11	11	9
19	2	9		1	12	12	9
20	4	5			9	12	9
21	3	7			10	12	10
22	3	7		1	11	12	10
23	3	7		2	12	12	10
24	8	3			11	12	10
25	8	3		1	12	12	10
26	8	3		2	13	13	10
27	8	3		3	14	14	10
28	4	7			11	14	10
29	4	7		1	12	14	10
30	2	3	5		10	14	10
31	2	3	5	1	11	14	11
32	2	3	5	2	12	14	12
33	2	3	5	3	13	14	12
34	2	3	5	4	14	14	12
35	5	7			12	14	12
36	4	9			13	14	12
37	4	9		1	14	14	12
38	4	9		2	15	15	12
39	3	13			16	16	12
40	8	5			13	16	12

Table 4.

A table of minimal decompositions of n, $s(n)$, and

$s^{(1)}(n) = \max_{1 \le \nu \le n} \{s(\nu)\}$

and

$s^{(2)}(n) = \min_{n \le \nu \le \infty} \{s(\nu)\}$

$s^{(1)}$ and $s^{(2)}$ are inverse finctions of n_{min} and n_{max} respectively.

s	$n_{min}(s)$	$n_{max}(s)$	s	$n_{min}(s)$	$n_{max}(s)$
1	1	1	36	296	13860
2	2	2	37	297	13861
3	3	3	38	298	16380
4	4	4	39	355	16381
5	5	6	40	356	27720
6	7	7	41	491	30030
7	8	12	42	492	32760
8	9	15	43	493	60060
9	14	20	44	654	60061
10	17	30	45	655	60062
11	18	31	46	656	60063
12	19	60	47	657	120120
13	26	61	48	658	120121
14	27	84	49	659	180180
15	38	105	50	984	180181
16	39	140	51	985	180182
17	48	210	52	1214	180183
18	49	211	53	1215	360360
19	50	420	54	1216	360361
20	51	421	55	1217	360362
21	81	422	56	1354	360363
22	82	423	57	1355	471240
23	83	840	58	1356	510510
24	98	841	59	1357	556920
25	139	1260	60	1358	1021020
26	150	1261	61	1359	
27	151	1540	62	1424	
28	164	2310	63	2124	
29	179	2520	64	2125	
30	194	4620	65	2126	
31	247	4621	66	2127	
32	248	5460	67	2498	
33	249	5461	68	2499	
34	250	9240	69	2500	
35	251	9241	70	2501	

Table 5.

A table of $n_{min}(s)$ and $n_{max}(s)$.

n	group length	first decomposition	second decomposition		third decomposition		s(n)
		k_i	k_i	r	k_i	r	
39	1	3 13	4 9	3	5 7	4	16
651	9	3 7 31	2 25 13	1	4 7 23	7	41
10965	15	3 5 17 43	2 27 7 29	3	4 7 17 23	17	68
14763	21	3 7 19 37	8 9 5 41	3	4 7 17 31	7	66
16226	10	2 7 19 61	2 9 17 53	8	3 5 23 47	11	89
45322	31	2 17 31 43	9 5 19 53	7	5 13 17 41	17	93
64780	10	4 5 41 79	2 13 47 53	14	8 9 29 31	52	129
91314	7	2 27 19 89	4 25 11 83	14	7 13 17 59	41	137
108669	6	3 11 37 89	2 25 41 53	19	7 11 17 83	22	140
109858	7	2 49 19 59	3 19 41 47	19	7 13 17 71	21	129
192882	6	2 3 17 31 61	3 5 13 23 43	27	2 25 7 19 29	32	114
459030	56	2 3 5 11 13 107	25 7 43 61	5	13 17 31 67	13	141
596900	4	4 25 47 127	5 19 61 103	15	9 17 47 83	47	203
680085	23	9 5 7 17 127	2 25 7 29 67	35	3 7 13 47 53	42	165
882702	23	2 9 19 29 89	4 25 7 13 97	2	8 9 13 23 41	54	148
1037210	34	2 5 19 53 103	8 9 5 43 67	50	16 3 17 31 41	74	182

Table 6

A table of the first 16 groups of integers with at least three minimal decompositions.

[1] P. M. B. Vitanyi, DOL-Languages and a Feasible Solution for a Word Problem, Mathematisch Centrum, MR 138/72, Amsterdam, 1972.
Also in: Proceedings of the Open House in Unusual Automata Theory, January 1972; Comp. Sci. Dept. Publ. No 15, Aarhus University, 1973.

[2] Mario Magidin, The "Best" Partition of an Integer
BIT 14 (1974) 203-208.

rettelser side 6, 7, 9, 10