1 Collatz problem

$$T(n_i)\begin{cases} n_i & = & 2\cdot k_i & \rightarrow & \frac{n_i}{2} & = & n_{i+i} \\ n_i & = & 2\cdot k_i+1 & \rightarrow & 3\cdot n_i+1 & = & n_{i+i} \end{cases}$$

also expressible as:

$$T(n)\begin{cases} \frac{3n+1}{2} \text{ if} & n\equiv 1 \quad (mod 2) \\ \frac{n}{2} \text{ if} & n\equiv 0 \quad (mod 2) \end{cases}$$

terminates when $n_m = 1$.
The number of steps, S s.t. $T(n_r)\rightarrow 1$ is

$$S=\left(\begin{array}{ccccccccc} 1 & 2 & 3 & 4 & 5 & 6 & 7 & 8 & 9\ldots \\ 1 & 2 & 8 & 3 & 6 & 9 & 17 & 4 & 20\ldots \end{array}\right.$$

1.1 Related problems

A similar problem may be formulated

$$g(n)\begin{cases} \frac{2n}{3} \text{ if} & n\equiv 0 \quad (mod 3) \\ \frac{4n-1}{3} \text{ if} & n\equiv 1 \quad (mod 3) \\ \frac{4n+1}{3} \text{ if} & n\equiv 2 \quad (mod 3) \end{cases}$$

For this the number of steps, P s.t. $T(n_r)\rightarrow 1$ is

$$P=\left(\begin{array}{ccccccccc} 1 & 2 & 3 & 4 & 5 & 6 & 7 & 8 & 9\ldots \\ 1 & 3 & 2 & 5 & 7 & 4 & 9 & 11 & 6\ldots \end{array}\right.$$

These and other generalizations will be taken up later.

2 First consideration

Some preliminary observations on the original problem.

Any power of two, 2^n reduces to 1 in n steps. Any odd p times a power of two, $p\cdot 2^n$ reduces to 1 in $n+r$ steps, where r is the “reduction” number of the odd number.

2.1 First approach

Let $n_1 = 2^k n_2 + c$

Case 1.

For $c = 0 \Rightarrow S = k$ for the transformation $T(n_1)\mapsto n_2$

The special case $c = 0 \wedge n_2 = 1 \Rightarrow S = k$ for $T(n_1)\mapsto 1$

So after the steps leading from $T(n_i)\mapsto n_j$ to where $n_j = 2n_{j+1} + c_{j+1}$ and $c_{j+1}\neq 0$.

Case 2.

When $n_1 = 2^k n_2 + c$ and $c \neq 0 \Rightarrow S > k$.

We will take the case where c is odd, since for c even we have

$$c_1 = 2c_2 \Rightarrow 2^k n_2 + c_1 = 2^k n_2 + 2c_2 = 2(2^{k-1} n_2 + c_2)$$

letting $2 \cdot (2^{k-1} n_2 + c_2) = 2 \cdot n_3$ we obtain the case $T(n_3)$.

Case 3.

When $n_1 = 2^k n_2 + c$ and $(c = 2c_1 + 1)$

$$\begin{aligned} &\Rightarrow n_1 = 2^k n_2 + 2c_1 + 1 \\ &2(2^{k-1} n_2 + c_1) + 1 = 2m + 1 \end{aligned}$$

where we let $m = 2^{k-1} n_1 + c_1$. So we will begin with the case $c = 1$

$$\begin{aligned} &T(n_1 = 2^k n_2 + 1) \mapsto 3(n_1) + 1 = \\ &3(2^k n_2 + 1) + 1 = \\ &3 \cdot 2^k n_2 + 3 + 1 = \\ &2^k \cdot 3n_2 + 4 = \\ &2(2^{k-1} 3n_2 + 2 \\ \\ &T\left(2(2^{k-1} 3n_2 + 2)\right) \mapsto 2^{k-1} 3n_2 + 2 \end{aligned}$$

If $k = 1$ then $T(n_1) = 3n_2 + 2$.

This leaves us with the two possibilities, $3n_2 + 2$ odd or even.

So far we have seen that for $T(n_1) \mapsto n_j = 2^k$ then $(n_1 = 2^p : k > p)$ or $(n_{j-1} = \chi_n)$.

Letting χ_n be a number expressible as

$$2^{2(n-1)} + 2^{2(n-2)} + \ldots 2^{2(n-n)}$$

and having the following properties:

$$4\chi_n + 1 = \chi_{n+1} \wedge 3\chi_n + 1 = 2^{2n}$$

we may then draw up the following table:

$n = 0$	$\chi_0 = 0$	$4 \cdot 0 + 1 = 1$	$3 \cdot 0 + 1 = 2^0$	0
$n = 1$	$\chi_1 = 1$	$4 \cdot 1 + 1 = 5$	$3 \cdot 1 + 1 = 2^{2 \cdot 1}$	$1 = 2^0$
$n = 2$	$\chi_2 = 5$	$4 \cdot 5 + 1 = 21$	$3 \cdot 5 + 1 = 2^{2 \cdot 2}$	$5 = 2^2 + 2^0$
$n = 3$	$\chi_3 = 21$	$4 \cdot 21 + 1 = 85$	$3 \cdot 21 + 1 = 2^{2 \cdot 3}$	$21 = 2^4 + 2^2 + 2^0$

Proceeding as above we have

$$\begin{array}{rlcr}
n=4 & \chi_4 & = & 85 \\
n=5 & \chi_5 & = & 341 \\
n=6 & \chi_6 & = & 1365 \\
n=7 & \chi_7 & = & 5461 \\
n=8 & \chi_8 & = & 21845 \\
n=9 & \chi_9 & = & 87381 \\
n=10 & \chi_{10} & = & 349525 \\
n=11 & \chi_{11} & = & 1398101 \\
n=12 & \chi_{12} & = & 5592405 \\
n=13 & \chi_{13} & = & 22369621 \\
n=14 & \chi_{14} & = & 892478485
\end{array}$$

2.2 example

Applied to a χ_n we get

$$\begin{array}{rcl}
3\,(\chi_n)+1 & = & 3(2^{2(n-1)} + 2^{2(n-2)} + \ldots 2^{2(n-n)}) + 1 \\
 & = & (2+1)(2^{2(n-1)} + 2^{2(n-2)} + \ldots + 2^{2(n-n)}) + 1
\end{array}$$

$$\text{since } (2+1)(2^k) = 2 \cdot 2^k + 2^k = 2^{k+1} + 2^k$$

$$\begin{array}{rl}
= & 2^{2(n-1)+1} + 2^{2(n-1)} + 2^{2(n-2)+1} + 2^{2(n-2)} + \ldots + 2^{2(n-n)+1} + 2^{2(n-n)} + 1 \\
= & 2^{2n-2+1} + 2^{2n-2} + 2^{2n-4+1} + 2^{2n-4} + \ldots + 2^{2\cdot 0)} + 2^{2\cdot 0} + 1 \\
= & 2^{2n-1} + 2^{2n-2} + 2^{2n-3} + 2^{2n-4} + \ldots + 2^1 + 2^0 + 1 \\
= & 2^{2n}
\end{array}$$

And we know that for

$$T(2^{2n}) \mapsto 1 \quad S = 2n$$

An example, given 14, we proceed until we get a χ number. We will simply denote $T(n_i) \mapsto n_{i-1}$ as $n_i \mapsto n_{i-1}$

$$\begin{array}{rcll}
14 & = & 2^3 + 2^2 + 2^1 & \mapsto \\
7 & = & 2^2 + 2^1 + 2^0 & \mapsto \\
22 & = & 2^4 + 2^2 + 2^1 & \mapsto \\
11 & = & 2^3 + 2^1 + 2^0 & \mapsto \\
34 & = & 2^5 + 2^1 & \mapsto \\
17 & = & 2^4 + 2^0 & \mapsto \\
52 & = & 2^5 + 2^4 + 2^2 & \mapsto \\
26 & = & 2^4 + 2^3 + 2^1 & \mapsto \\
13 & = & 2^3 + 2^2 + 2^0 & \mapsto \\
40 & = & 2^5 + 2^3 & \mapsto \\
20 & = & 2^4 + 2^2 & \mapsto \\
10 & = & 2^3 + 2^1 & \mapsto \\
5 & = & 2^2 + 2^0 &
\end{array}$$

and 5 is a χ_n number.

3 appendix

This is a short quick program to generate a few values of S.

```
/* -----------(  COLLATZ.C )------------------ *
 * ------------------------------------------- */
#include "stdio.h"
main()
{
      gener(32000);
}
gener(n)
int n;
{
    long i, c, j;
    for (c=2;c<=n;c++) {
    j=1;
    i = c;
    while (i != 1 ) {
        if ((i % 2) == 0 )
        i = i/2;
          else
        i = ((3 * i) + 1);
        j = j+1;
    }
printf("\n  %10ld %10ld  ",c,j);
    }
}
```

References

[1] Author *Title*. American Mathematical Monthly 92, 1 (Jan. 1985)