Claude's Cycles

Don Knuth, Stanford Computer Science Department
(28 February 2026; revised 06 March 2026)

Shock! Shock! I learned yesterday that an open problem I'd been working on for several weeks had just been solved by Claude Opus 4.6 — Anthropic's hybrid reasoning model that had been released three weeks earlier! It seems that I'll have to revise my opinions about "generative AI" one of these days. What a joy it is to learn not only that my conjecture has a nice solution but also to celebrate this dramatic advance in automatic deduction and creative problem solving. I'll try to tell the story briefly in this note.

Here's the problem, which came up while I was writing about directed Hamiltonian cycles for a future volume of *The Art of Computer Programming*:

> Consider the digraph with m^3 vertices ijk for $0 \le i,j,k < m$, and three arcs from each vertex, namely to i^+jk, ij^+k, and ijk^+, where $i^+ = (i+1) \bmod m$. Try to find a general decomposition of the arcs into three directed m^3-cycles, for all $m > 2$.

I had solved the problem for $m = 3$, and asked for a generalization as part of the answer to an exercise in [3]. My friend Filip Stappers rose to the challenge, and empirically discovered solutions for $4 \le m \le 16$; therefore it became highly likely that the desired decompositions do exist, except when $m \le 2$.

Indeed, it was Filip who had the gumption to pose this question to Claude, using exactly the wording above. He also gave guidance/coaching, instructing Claude to summarize its ongoing progress:

> ** After EVERY `exploreXX.py` run, IMMEDIATELY update this file [`plan.md`] before doing anything else. ** No exceptions. Do not start the next exploration until the previous one is documented here.

And Claude's plan of attack was quite admirable. First it reformulated the problem: "Need sigma: $Z_m^3 \to S_3$, assigning a permutation of $\{0,1,2\}$ at each vertex; cycle c at vertex v goes in direction sigma$(v)[c]$. Each resulting functional digraph must be a single Hamiltonian cycle." Then it tried for a simple answer: "'If sigma is cyclic, i.e. $f_c(v) = (c+g(v)) \bmod 3$ for some $g: Z_m^3 \to Z_3$, then we only need one function g. Tested linear $g = (a\cdot i + b\cdot j + c\cdot k) \bmod 3$ and quadratic variants. *No linear or simple quadratic g works.*"

Claude's second exploration was what it called a DFS search attempt (depth-first search): "Attempted brute-force DFS over sigma assignments for $m = 3$ (6^{27} search space). Too slow without good pruning."

Thirdly, it tried what it called 2D serpentine analysis: "*Key 2D result:* The pattern $[0^{m-1}1]^m$ gives a Hamiltonian cycle in $\mathrm{Cay}(Z_m^2, \{e_0, e_1\})$ for $m = 3$. Its complement also works, giving a valid 2D decomposition." This is really impressive! First Claude recognized the digraph as a so-called Cayley digraph with two generators, traditionally called e_0 and e_1; a mathematician will understand that the arcs act as a group on m^2 vertices, where e_0 increments the first component while e_1 increments the second. Then it formulated the cycle symbolically for general m, although for some reason it still said that it was only looking at $m = 3$. And it decided to call this a "serpentine pattern"!

Exploration number 4 constructed the "3D serpentine pattern" $[0^{m-1}1]^{m-1}[0^{m-1}2]^m$. (That's a typographic error for $[[0^{m-1}1]^{m-1}0^{m-1}2]^m$. It's a classical sequence called the "modular m-ary Gray code" on page 299 of [2].) After deleting this cycle from the graph, Claude looked at the remaining 2-regular digraph and found that it has "rigid structure making decomposition difficult."

Nevertheless, exploration number 5 analyzed that residual digraph, finding some cycles. Unfortunately they didn't work out.

Back to the drawing board. Several other approaches were tried, for example looking at the choices to be made at vertices that lie on the hyperplane where $i+j+k$ is congruent to -1. Nothing promising showed up until exploration number 15, which introduced what Claude called a *fiber decomposition*: "The quotient map $?(i,j,k) = i+j+k \bmod m$ maps all arcs from fiber F_s to F_{s+1}. The digraph is *layered.* In fiber coordinates (i,j) with $k = (s-i-j) \bmod m$."

Using those fiber coordinates, the problem could now be reformulated as choosing, for each (i,j,s), a permutation of the three operators that shift (i,j) by either $(1,0)$ or $(0,1)$ or $(0,0)$. "Tried sigma depending only on fiber index s (rotation by s): gives the $(i+j+k) \bmod 3$ pattern which doesn't work. Random local search found solutions for $m = 3$ but no general pattern emerged."

Exploration 18 "implemented the full fiber framework as executable code. *Exhaustive backtracking* for $m = 3$ found a valid decomposition in 0.1s (10932 nodes)." In this solution the permutation sigma was the same for all (i, j) in fiber $s = 0$. A similarly "uniform" fiber was found for $m = 4$ by simulated annealing, as part of exploration 20.

In exploration 21, Claude realized that the 2D serpentine function "$Q(i,j) = (i+1,j)$ if $i+j \neq m-1 \bmod m$, else $(i,j+1)$" had nice properties. It looked promising when a uniform choice is made at fiber 0 and a version of the serpentine at all other fibers, if and only if m is odd.

That triggered a bunch of experiments with simulated annealing "at scale"; but all was for naught. The conclusion after exploration 25 was "SA can find solutions but cannot give a general construction. Need pure math."

So now Claude put on a new thinking cap. At one point it said to itself, "Maybe the right framing is: don't think in fibers, think directly about what makes a Hamiltonian cycle."

There was a near miss at exploration number 27. "Take the 3D serpentine for cycle 0, apply cyclic coordinate rotation to get cycles 1 and 2: $d_1(i,j,k) = (d_0(j,k,i)+1) \bmod 3$; $d_2(i,j,k) = (d_0(k,i,j)+2) \bmod 3$; *all three are individually Hamiltonian* for every $m = 3\,..\,9$ tested! Only $3(m-1)$ out of m^3 vertices have conflicts. *all* lie on the hyperplane $i+j+k = m-1 \bmod m$."

Unfortunately, those conflicts couldn't be resolved. Exploration 29 proved that many plausible scenarios were in fact impossible. "This kills the 'single-hyperplane + rotation' approach entirely. ... We must allow the direction function to use *different* values across a rotation orbit."

But, aha. Exploration 30 went back to the solution found by SA in exploration 20 and noticed that the choice at each fiber depends on only a single coordinate: $s = 0$ only on j; $s = 1$ and $s = 2$ only on i. This led to a concrete construction (exploration 31), in the form of a Python program, which produced valid results for $m = 3$, 5, 7, 9, and 11 — hooray! "All three cycles are Hamiltonian, all arcs are used, perfect decomposition!"

Here is a simplification of that Python program, converted into C form:

```
int c, i, j, k, m, s, t;
char *d;
for (c = 0; c < 3; c++) {
  for (t = i = j = k = 0; ; t++) {
    printf("%x%x%x ", i, j, k);
    if (t == m*m*m) break;
    s = (i+j+k) % m;
    if (s == 0) d = (j == m-1? "012" : "210");
    else if (s == m-1) d = (i == 0? "210" : "120");
    else d = (i == m-1? "201" : "102");
    switch (d[c]) {
case '0': i = (i+1) % m; break;
case '1': j = (j+1) % m; break;
case '2': k = (k+1) % m; break;
    }
  }
  printf("\n");
}
```

Filip Stappers tested Claude's Python program for all odd m between 3 and 101, finding perfect decompositions each time. Thus he concluded, quite reasonably, that the problem was indeed solved for odd values of m. And he quickly sent me the shocking news.

Of course, a rigorous proof was still needed. And the construction of such a proof turns out to be quite interesting. Let's look, for example, at the first cycle that is printed; we must prove that it has length m^3.

The rule for that cycle is nontrivial, yet fairly simple: Let $s = (i+j+k) \bmod m$.

$$\begin{aligned}
\text{When } s = 0, &\ \text{ bump } i \text{ if } j = m-1, \text{ otherwise bump } k. \\
\text{When } 0 < s < m-1, &\ \text{ bump } k \text{ if } i = m-1, \text{ otherwise bump } j. \\
\text{When } s = m-1, &\ \text{ bump } k \text{ if } i = 0, \text{ otherwise bump } j.
\end{aligned}$$

("Bump" means "increase by 1, mod m.") Hence in the special case $m = 3$ that cycle is

$$\begin{aligned}
&022 \longrightarrow 002 \longrightarrow 000 \longrightarrow 001 \longrightarrow 011 \longrightarrow 012 \longrightarrow 010 \longrightarrow 020 \longrightarrow 021 \longrightarrow \\
&121 \longrightarrow 101 \longrightarrow 111 \longrightarrow 112 \longrightarrow 122 \longrightarrow 102 \longrightarrow 100 \longrightarrow 110 \longrightarrow 120 \longrightarrow \\
&220 \longrightarrow 221 \longrightarrow 201 \longrightarrow 202 \longrightarrow 200 \longrightarrow 210 \longrightarrow 211 \longrightarrow 212 \longrightarrow 222 \longrightarrow 022.
\end{aligned}$$

And in the special case $m = 5$ it's

$$\begin{aligned}
&042 \longrightarrow 002 \longrightarrow 012 \longrightarrow 022 \longrightarrow 023 \longrightarrow 024 \longrightarrow 034 \longrightarrow 044 \longrightarrow 004 \longrightarrow 000 \longrightarrow 001 \longrightarrow 011 \longrightarrow 021 \longrightarrow \\
&031 \longrightarrow 032 \longrightarrow 033 \longrightarrow 043 \longrightarrow 003 \longrightarrow 013 \longrightarrow 014 \longrightarrow 010 \longrightarrow 020 \longrightarrow 030 \longrightarrow 040 \longrightarrow 041 \longrightarrow \\
&141 \longrightarrow 101 \longrightarrow 111 \longrightarrow 121 \longrightarrow 131 \longrightarrow 132 \longrightarrow 142 \longrightarrow 102 \longrightarrow 112 \longrightarrow 122 \longrightarrow 123 \longrightarrow 133 \longrightarrow 143 \longrightarrow \\
&103 \longrightarrow 113 \longrightarrow 114 \longrightarrow 124 \longrightarrow 134 \longrightarrow 144 \longrightarrow 104 \longrightarrow 100 \longrightarrow 110 \longrightarrow 120 \longrightarrow 130 \longrightarrow 140 \longrightarrow \\
&240 \longrightarrow 200 \longrightarrow 210 \longrightarrow 220 \longrightarrow 230 \longrightarrow 231 \longrightarrow 241 \longrightarrow 201 \longrightarrow 211 \longrightarrow 221 \longrightarrow 222 \longrightarrow 232 \longrightarrow 242 \longrightarrow \\
&202 \longrightarrow 212 \longrightarrow 213 \longrightarrow 223 \longrightarrow 233 \longrightarrow 243 \longrightarrow 203 \longrightarrow 204 \longrightarrow 214 \longrightarrow 224 \longrightarrow 234 \longrightarrow 244 \longrightarrow \\
&344 \longrightarrow 304 \longrightarrow 314 \longrightarrow 324 \longrightarrow 334 \longrightarrow 330 \longrightarrow 340 \longrightarrow 300 \longrightarrow 310 \longrightarrow 320 \longrightarrow 321 \longrightarrow 331 \longrightarrow 341 \longrightarrow \\
&301 \longrightarrow 311 \longrightarrow 312 \longrightarrow 322 \longrightarrow 332 \longrightarrow 342 \longrightarrow 302 \longrightarrow 303 \longrightarrow 313 \longrightarrow 323 \longrightarrow 333 \longrightarrow 343 \longrightarrow \\
&443 \longrightarrow 444 \longrightarrow 440 \longrightarrow 441 \longrightarrow 401 \longrightarrow 402 \longrightarrow 403 \longrightarrow 404 \longrightarrow 400 \longrightarrow 410 \longrightarrow 411 \longrightarrow 412 \longrightarrow 413 \longrightarrow \\
&414 \longrightarrow 424 \longrightarrow 420 \longrightarrow 421 \longrightarrow 422 \longrightarrow 423 \longrightarrow 433 \longrightarrow 434 \longrightarrow 430 \longrightarrow 431 \longrightarrow 432 \longrightarrow 442 \longrightarrow 042.
\end{aligned}$$

Triples for the same value of s are spaced exactly m steps apart. To prove that all m^3 triples occur, we need to prove that the m^2 triples for any given value of s are all present.

Notice that the first coordinate, i, changes only when $s = 0$ and $j = m-1$. Thus the m^2 triples with any given value of the first coordinate i all occur consecutively. (Our example cycles indicate this by starting at the vertex where i has just changed to 0, instead of starting at vertex 000.)

Notice that the cycle elements with $i = 0$ must start in general with the vertex $0(m-1)2$, because the previous vertex must have had $i = j = m-1$ and $s = 0$. And $0(m-1)2$, which has $s = 1$, is followed in general by 002, 012, ..., $0(m-3)2$, $0(m-3)3$, taking us back to $s = 0$.

In general $0(m-k)k$ is immediately followed by $0(m-k)(k+1)$ when $1 < k < m$; and then j increases until we reach $0(m-k-2)(k+1)$, whose successor is $0(m-k-2)(k+2)$. Thus k is increased by 2, modulo m, each time we hit a vertex with $s = 0$. Since m is odd, we'll eventually get to $0(m-1)1$—at which time we will have seen all m^2 vertices of the form $0jk$. And the successor of $0(m-1)1$ is $1(m-1)1$.

So far so good! In general, the cycle elements whose first component i satisfies $0 < i < m-1$ will start with $i(m-1)(2-i)$, where $2-i$ is of course evaluated mod m. If all goes well those elements should include all m^2 vertices that begin with i, ending with $i(m-i)(1-i)$—whose successor is $(i+1)(m-1)(1-i)$. And all *does* go well: We repeatedly advance the second component except when $s = 0$; hence the vertices that we see when $s = 0$ are $i(m-2)(2-i)$, $i(m-3)(3-i)$, ..., $i0(m-i)$, $i(m-1)(1-i)$.

Finally we reach $(m-1)(m-1)3$, the first vertex for which $i = m-1$. The local rules change again: From now on we'll repeatedly advance the *third* component, except when $s = m-1$. The vertices we see when $s = 0$ are $(m-1)01$, $(m-1)10$, ..., $(m-1)(m-1)2$. QED.

We've now proved that the first cycle (the cycle for $c = 0$ in the C program) is Hamiltonian. Similar proofs can be carried out for the other two cycles. (See the Appendix.)

For fun, let's consider a larger class of cycles for which such proofs exist. Indeed, it turns out that there are hundreds of way to solve the stated decomposition problem for odd m; Claude Opus 4.6 happened to discover just one of them, by deducing where to look.

We shall say that a Hamiltonian cycle C for $m = 3$ is *generalizable* if the following construction yields a Hamiltonian cycle for all odd $m \geq 3$: "Given an arbitrary vertex IJK for $0 \leq I, J, K < m$, let $i = I'$, $j = J'$, $s = S'$, and $k = (s-i-j) \bmod 3$, where $S = (I+J+K) \bmod m$, $0' = 0$, $(m-1)' = 2$, and $x' = 1$ for $0 < x < m-1$. Obtain the successor of IJK by bumping the coordinate that cycle C bumps when forming the successor of ijk." For example, if $m = 5$ and $IJK = 301$, we have $i = 1$, $j = 0$, $S = 4$, $s = 2$, $k = 1$. So the successor of 301 will be either 401 or 311 or 302, depending on whether C contains the arc $101 \longrightarrow 201$ or $101 \longrightarrow 111$ or $101 \longrightarrow 102$.

Claude's cycle in the example above for $m = 3$ is generalizable; indeed, we've seen its generalization for $m = 5$. But it's easy to find cycles that are not generalizable. For instance, if C is the cycle

$$000 \longrightarrow 001 \longrightarrow 002 \longrightarrow 012 \longrightarrow 010 \longrightarrow 011 \longrightarrow 021 \longrightarrow 022 \longrightarrow 020 \longrightarrow 120 \longrightarrow 100 \longrightarrow 101 \longrightarrow 111 \longrightarrow 121 \longrightarrow$$
$$122 \longrightarrow 102 \longrightarrow 112 \longrightarrow 110 \longrightarrow 210 \longrightarrow 211 \longrightarrow 212 \longrightarrow 222 \longrightarrow 220 \longrightarrow 221 \longrightarrow 201 \longrightarrow 202 \longrightarrow 200 \longrightarrow 000$$

(which happens to be the lexicographically smallest of all Hamiltonian cycles for the case $m = 3$), we get the following "generalization" for $m = 5$:

$$000 \longrightarrow 001 \longrightarrow 002 \longrightarrow 003 \longrightarrow 004 \longrightarrow 014 \longrightarrow 010 \longrightarrow 011 \longrightarrow 012 \longrightarrow 013 \longrightarrow 023 \longrightarrow 024 \longrightarrow 020 \longrightarrow$$
$$021 \longrightarrow 022 \longrightarrow 032 \longrightarrow 033 \longrightarrow 034 \longrightarrow 030 \longrightarrow 031 \longrightarrow 041 \longrightarrow 042 \longrightarrow 043 \longrightarrow 044 \longrightarrow 040 \longrightarrow 140 \longrightarrow$$
$$100 \longrightarrow 101 \longrightarrow 102 \longrightarrow 103 \longrightarrow 113 \longrightarrow 123 \longrightarrow 124 \longrightarrow 120 \longrightarrow 121 \longrightarrow 221 \longrightarrow 231 \longrightarrow 232 \longrightarrow 233 \longrightarrow$$
$$234 \longrightarrow 334 \longrightarrow 344 \longrightarrow 340 \longrightarrow 341 \longrightarrow 342 \longrightarrow 302 \longrightarrow 312 \longrightarrow 313 \longrightarrow 314 \longrightarrow 310 \longrightarrow 410 \longrightarrow 411 \longrightarrow$$
$$412 \longrightarrow 413 \longrightarrow 414 \longrightarrow 424 \longrightarrow 420 \longrightarrow 421 \longrightarrow 422 \longrightarrow 423 \longrightarrow 433 \longrightarrow 434 \longrightarrow 430 \longrightarrow 431 \longrightarrow 432 \longrightarrow$$
$$442 \longrightarrow 443 \longrightarrow 444 \longrightarrow 440 \longrightarrow 441 \longrightarrow 401 \longrightarrow 402 \longrightarrow 403 \longrightarrow 404 \longrightarrow 400 \longrightarrow 000.$$

Oops — this cycle has length 75, not 125.

It turns out that there are exactly 11502 Hamiltonian cycles for $m = 3$, of which exactly 1012 generalize to Hamiltonian cycles for $m = 5$. Furthermore, exactly 996 of them generalize to Hamiltonian cycles for both $m = 5$ and $m = 7$. And those 996 are in fact generalizable to all odd $m > 1$.

Now here's the point: Let's say that a decomposition is "Claude-like" if it can be generated by a C program like the one above, in which the permutations d of $\{0, 1, 2\}$ depend only on whether i, j, and s are 0 or $m-1$ or not. No special cases other than 0 and $m-1$ are allowed to affect the choice of d.

Theorem. *A Claude-like decomposition is valid for all odd $m > 1$ if and only if each of the three sequences that it defines for $m = 3$ is generalizable.*

Proof. If those three sequences aren't Hamiltonian, or if they don't generalize to Hamiltonian cycles for all odd $m > 1$, they certainly don't define a valid decomposition. Conversely, if they are Hamiltonian cycles that generalize to Hamiltonian cycles for all odd $m > 1$, they certainly are valid: Every vertex ijk appears in each of the three cycles, and its three outgoing arcs are partitioned properly because d is a permutation. ∎

By setting up an exact cover problem, using the 11502 Hamiltonian cycles for $m = 3$, we can deduce that there are exactly 4554 solutions to the $3 \times 3 \times 3$ decomposition problem. And in a similar fashion, if we study all ways to cover every arc by using just three of the 996 cycles that are generalizable, we find that exactly 760 of those 4554 solutions involve only generalizable cycles — about one in every 6. Consequently the theorem tells us that exactly 760 Claude-like decompositions are valid for all odd $m > 1$.

Maybe the decomposition that Claude found wasn't the "nicest" of those 760. Can we do better? I looked at several of them and found that they have somewhat different behaviors. Yet I didn't encounter any that were actually nicer.

The dependence on i, j, and s means that these decompositions don't have cyclic symmetry. I did notice, to my surprise, that 136 of the 760 generalizable $3 \times 3 \times 3$ cycles remain generalizable when we map ijk to jki. However, none are common to all three mappings $\{ijk, jki, kij\}$.

Filip told me that the explorations reported above, though ultimately successful, weren't really smooth. He had to do some restarts when Claude stopped on random errors; then some of the previous search results were lost. After every two or three test programs were run, he had to remind Claude again and again that it was supposed to document its progress carefully.

Delicious success for odd m, at exploration number 31, came about one hour after the session began.

This decomposition problem remains open for even values of m. The case $m = 2$ was proved impossible long ago [1]. As part of exploration number 24, Claude said that it had found solutions for $m = 4$, $m = 6$, and $m = 8$; yet it saw no way to generalize those results.

Filip also told me that he asked Claude to continue on the even case after the odd case had been resolved. "But there after a while it seemed to get stuck. In the end, it was not even able to write and run explore programs correctly anymore, very weird. So I stopped the search."

All in all, however, this was definitely an impressive success story. I think Claude Shannon's spirit is probably proud to know that his name is now being associated with such advances. Hats off to Claude!

Appendix. Claude's second cycle ($c = 1$) is governed by the following rules: "If $s = 0$, bump j. If $0 < s < m+1$, bump i. If $s = m-1$ and $i > 0$, bump k. If $s = m-1$ and $i = 0$, bump j."

We can show that the vertices with $s = 0$ are seen in the following order, for $k = 0, 1, \ldots, m-1$: $0k(-k)$, $(-2)(1+k)(1-k)$, $(-4)(2+k)(2-k)$, $\ldots$, $2(-1+k)(-1-k)$. And that will establish the order in which vertices with $s = 1, 2, \ldots, m-1$ are seen.

Claude's third cycle ($c = 2$) is governed by somewhat more complex rules: "If $s = 0$ and $j < m-1$, bump i. If $s = 0$ and $j = m-1$, bump k. If $0 < s < m-1$ and $i < m-1$, bump k. If $0 < s < m-1$ and $i = m-1$, bump j. If $s = m-1$, bump i."

The vertices with $s = 0$ and $j < m-1$ are seen in the following order: $0j(-j)$, $2j(-2-j)$, $4j(-4-j)$, $\ldots$, $(-2)j(2-j)$. And the successors of the latter vertex are $(-1)j(2-j)$, $(-1)(j+1)(2-j)$, $(-1)(j+2)(2-j)$, $\ldots$, $(-1)(j-2)(2-j)$, $0(j-2)(2-j)$.

But the vertices with $s = 0$ and $j = m-1$ are seen thus: $0(-1)1$, $1(-1)0$, $2(-1)(-1)$, $\ldots$, $(-1)(-1)2$. And the successors of the latter vertex are $(-1)(-1)3$, $(-1)03$, $(-1)13$, $\ldots$, $(-1)(-3)3$, $0(-3)3$.

Thus in every case the sequence of m vertices for $s = 0$ and a given j is followed by the sequence for $s = 0$ and $j-2$, modulo m.

Postscript. On March 3, Stappers wrote me as follows: "The story has a bit of a sequel. I put Claude Opus 4.6 to work on the $m =$ even cases again for about 4 hours yesterday. It made some progress, but not a full solution. The final program ... sets up a partial fiber construction similar to the odd case, then runs a search to fix it all up. ... Claude spent the last part of the process mostly on making the search quicker instead of looking for an actual construction. ... It was running many programs trying to find solutions using simulated annealing or backtrack. After I suggested to use the ORTools CP-SAT [part of Google's open source toolkit, with the AddCircuit constraint] to find solutions, progress was better, since now solutions could be found within seconds." This program is [4].

Then on March 4, another friend — Ho Boon Suan in Singapore — wrote as follows: "I have code generated by `gpt-5.3-codex` that generates a decomposition for even $m \geq 8$. ... I've tested it for all even m from 8 to 200 and bunch of random even values between 400 and 2000, and it looks good. Seems far more chaotic to prove correctness by hand here though; the pattern is way more complex." That program is [5]. (Wow. The graph for $m = 2000$ has 8 billion vertices!)

Postpostscript. Kevin Buzzard just told me that Kim Morrison, from the Lean community, quickly formalized my proof that Claude's construction is correct. Indeed, Kim had posted his verification online already on March 4 [6]. That's good to know, because I've been getting more errorprone lately.

On another front, Maximilian Reitbauer, aka "Exocija," has found a new construction for odd m that is probably the simplest possible, from the standpoint of computation, although it might perhaps not have the simplest proof: We can get a valid decomposition by replacing lines 8–10 of the C program by

if $(s == 0)$ $d = (j == m-1?$ `"201"` : `"021"`);
else if $(s == m-1)$ $d = (j == 0?$ `"102"` : `"120"`);
else $d =$ `"012"`;

notice that this solution uses only s and j, not i. Furthermore, the identity permutation `"012"` is used at almost every step! (I would have found this solution myself if I'd taken time to look carefully at all 760 of the generalizable solutions for $m = 3$, because this one is #369 on that list.) He told me on March 6 that he found his proof [7] by pasting text back and forth between GPT 5.4 (Extended Thinking) and Claude 4.6 Sonnet (Thinking).

Breaking news: The problem for *even* values of m is no longer in doubt! First, I heard again from Ho Boon Suan, who exercised the newly released GPT-5.4 Pro as follows:

> Your task is to prove rigorously that the algorithm given in [5] genuinely does always yield three cycles of length m^3 each when m is an even number ≥ 8. It would be good to give insight on why this algorithm works, as well as if there are simpler constructions. For context, see `https://cs.stanford.edu/~knuth/papers/claude-cycles.pdf`.

The result was a beautifully formatted and apparently flawless 14-page paper [8], containing the desired exposition and proof. Ho said this was entirely the machine's doing; he didn't have to edit the paper in any way. And GPT-5.4 Pro's fascinating record of the entire interaction is available online [9].

Finally, I heard from Keston Aquino-Michaels, who has brought these studies to a fitting conclusion and recorded them faithfully in [10]. Like Reitbauer, he interacted with two data-sharing LLM agents that have complementary skills, namely GPT and Claude. The result was yet another valid decomposition for the case of odd m, together with an elegant decomposition for the case of even m that's considerably simpler than [5]. Among other things, he turned up a relevant reference [11] that I'd missed (although it doesn't solve the decomposition problem).

Best of all, his paper [12] includes a careful analysis of how such joint interaction worked, with potentially significant implications for how new problems can be tackled and resolved in the future.

Over And Out. Dear reader, I hope you have enjoyed reading this story at least half as much as I've enjoyed writing it. We are living in very interesting times indeed.

But please do *not* write to me with further thoughts about the topics considered here. Please work with like-minded researchers as much as you can, but without putting me into the loop!

I absolutely must get back to writing [3], which will soon contain further stories of a completely different kind, stories that I'm much more qualified to write than stories about LLMs.

May the force be with you.

References.

[1] Jacques Aubert and Bernadette Schneider, "Graphes orientes indécomposables en circuits hamiltoniens." *Journal of Combinatorial Theory* **B32** (1982), 347–349.

[2] Donald E. Knuth, *The Art of Computer Programming*, Volume 4A: *Combinatorial Algorithms, Part 1* (Upper Saddle River, N. J.: Addison–Wesley, 2011), xvi+883 pp.

[3] Donald E. Knuth, preliminary drafts entitled "Hamiltonian paths and cycles," currently posted at `https://cs.stanford.edu/~knuth/fasc8a.ps.gz` and updated frequently as more material is being accumulated.

[4] `https://cs.stanford.edu/~knuth/even_solution.py`

[5] `https://cs.stanford.edu/~knuth/even_closed_form.c`

[6] `https://github.com/kim-em/KnuthClaudeLean/`

[7] `https://cs.stanford.edu/~knuth/alternative_hamiltonian_decomposition.pdf`

[8] `https://cs.stanford.edu/~knuth/even_closed_form_proof_final.pdf`

[9] `https://chatgpt.com/share/69aaab4b-888c-8003-9a02-d1df80f9c791`

[10] `https://github.com/no-way-labs/residue`

[11] Iren Darijani, Babak Miraftab, and Dave Witte Morris, "Arc-disjoint Hamiltonian paths in Cartesian products of directed cycles," *Ars Mathematica Contemporanea* **25** (2025), P2.10:1–32.

[12] Keston Aquino-Michaels, "Completing Claude's cycles: Multi-agent structured exploration on an open combinatorial problem," 20 pages. (In [10].)