

# Re: Need Graph Isomorphism Algorithm De-bunked

---

*Source:* <http://www.derkeiler.com/Newsgroups/sci.crypt/2006-09/msg01004.html>

---

- *From:* Mike Amling <nospam@xxxxxxxxxx>
  - *Date:* 28 Sep 2006 11:45:42 EDT
- 

fgrieu wrote:

Mike Amling wrote :

I've coded up something I think is equivalent to what you suggest. It works for up to 7 nodes (simply connected, undirected, no loopback), but for 8 it only found 10126 equivalence classes, not 11117. I'll look again at the code to see if something obvious is amiss.

My intent was to write an equivalent to the (second) Bill Cox algorithm that you verified up to 8 nodes, except not relying on a hash. I'd bet the problem (assuming it is not in your implementation) is in my rendering of Bill Cox's algorithm, not in the removal of the hash. Maybe I should have used Bill Cox's code as a basis, rather than his text. I'll check this in more detail.

The only data that you keep is the "result" concatenation of sorted unique "longhash"es (bad name; They're bit strings formed by concatenation, and not hashes.), right? I'm discarding final values assigned to the nodes. There is no marker between the list from one iteration and the appended list from the next iteration, but none should be necessary. And, mea culpa, I kept only a 32-bit hash of the results, not the results themselves (but I can hardly believe there've been 991 collisions).

It would help if you could give two graphs that are not distinguished by your new code, but are distinguished by the previous version (is it