

[OT] Linked list fixup [was: Fast DES IP]

Source: <http://www.derkeiler.com/Newsgroups/sci.crypt/2004-12/0412.html>

From: Francois Grien (*fgrien_at_francenet.fr*)

Date: 12/09/04

Date: Thu, 09 Dec 2004 11:52:32 +0100

In article <873byfvmlc.fsf@nonospaz.fatphil.org>, Phil Carmody <thefatphil_demunged@yahoo.co.uk> wrote:

> "Douglas A. Gwyn" <DAGwyn@null.net> writes:
> > *There is a related XOR trick sometimes used to*
> > *traverse a linked list in either forward or*
> > *reverse direction with only a single link field*
> > *in each node, with one extra field of information*
> > *for the whole list needed to get started (that*
> > *can be stored in a list header). If you aren't*
> > *familiar with this trick, it is a nice little*
> > *exercise to figure it out.*
>
> Now that (as far as I know) *_is_ Knuth's.*

While I fail to identify which trick / Knuth's algo this is, I can't resist a little [non-crypto] puzzle on linked lists.

We have a non-empty linked list in RAM, of unknown size n , and a pointer to the head node. Instead of nicely ending with a node having a NULL "next" field, the list is "loopy" and the last node in the list has a "next" field that points to some unknown node in the list. Find an $O(n)$ algorithm using $O(1)$ memory that fixes the list, changing the "next" field of the last element to NULL.

Note: solutions that store an extra bit of info in each node do not count as $O(1)$ memory.

Francois Grien