Re: Hashing for big list of last names

Source: http://coding.derkeiler.com/Archive/General/comp.programming/2006−09/msg00377.html

- *From*: CBFalconer <cbfalconer@xxxxxxxxx>
- *Date*: Thu, 14 Sep 2006 02:00:49 −0400

*** top−posting fixed ***
sprash wrote:

> Ben Pfaff wrote:
>
> > sprash25@xxxxxxxxx writes:
> >
> > > I would like to design a hash table for a big list of last names
> > > of patients.
> > >
> > > Any recommendations on the hashing type (chaining, open
> > > addressing, double hashing) and the function I should use?
> >
> > It entirely depends on your constraints. If you describe your
> > application in some more detail, perhaps we can give
> > recommendations.
>
> To put it simply, the module I am writing essentially reads the
> list of last names of Patients from a CSV file which is a very
> large list. Next, based on the inputs I get from another
> application, I need to manipulate this list by searching and
> inserting new last names and also delete certain last names from
> my list.
>
> There is existing legacy code that is implementing this using
> chaining (function is based on the first character of the last
> name). This causes a lot of collision for common characters such
> as A, D etc and very few for X, Q etc. Basically the distribution
> is understandably uneven.
>
> I am wondering if there are any better ideas out there.

To all practical purposes it is all done for you, and available
under GPL. See:

<http://cbfalconer.home.att.net/download/>

and look for the hashlib and id2id−20 packages.

−−
"I have a creative mind. You (singular) are eccentric.
He is insane. We are losing sight of reality.
You (plural) are smoking crack. They are certifiable."
Declension of verbs, per Lewin Edwards

−−

.