Subject: Re: Efficient comparison of arrays
Posted by William Clodius on Wed, 13 Aug 1997 07:00:00 GMT
View Forum Message <> Reply to Message

Andy Loughe wrote:
>
> Hi!
>
> I feel I should know the answer to this one, but I don't, so here goes.
>
> Given vectors of the type...
>
> a = [1,2,3,4,5]
> b = [3,4,5,6,7]
>
> What is the most efficient way to determine which values that occur in
> a also occur in b (i.e., the values [3,4,5] occur in both a and b).
>
> Presumably this needs to be done without loops (to be efficient), but an
> obvious solution escapes me.
>
> Thanks for your help.
> <snip>

It is not clear whether you want the values or the positions of the values (the second is harder). For the first case it is possible to this with an algorithm that approximately scales as of order N ln N.

Assume you have two vectors of length N and M respectively.

If unsorted, sort them, => operations of order N ln N and M ln M.

If duplicates within a vector can exists, delete duplicates. Operations of order N and M.

Concatenate arrays. An operation of order N + M

Sort concatenated array. An operation of order (N+M) ln (N+M)

Inspect adjacent elements of the sorted array to find duplicates. An operation of order N+M.

Done.

--

William B. Clodius Phone: (505)-665-9370
Los Alamos Nat. Lab., NIS-2 FAX: (505)-667-3815

PO Box 1663, MS-C323 Group office: (505)-667-5776
Los Alamos, NM 87545 Email: wclodius@lanl.gov