[3.19] Test a list for identical elements

If you need to know if all the elements of a list are identical, try this function:

```
lstident(l)
Func
©(list) return true if all elements of 'list' are the same
©18jan01/dburkett@infinet.com
when(Δlist(l)=newlist(dim(l)-1),true,false)
EndFunc
```

The function returns *true* if all the list elements are equal, and *false* otherwise. The basic idea is to use `Δlist()` to create a list of the differences between the elements, then compare that result to a list of zeros.

This function works if the list elements are numbers (including complex numbers) or strings, but does *not* work if the list elements are expressions. If the elements are all the same expression, *lstident()* returns *true*, but if the elements are not the same, *lstident()* returns itself. It is a little odd that this method works for lists of strings, since *newlist()* returns a list with all zero elements, but

`Δlist("a","a","a")` returns `{0,0}`

and

`Δlist("a","a","b")` returns `{0,"b-a"}`

Since "b-a" is a string, and strings test as 'not equal' to numbers, it works.

Note that the *when()* function is necessary, otherwise the expression will return a list of *true* and *false* elements, instead of a single *true* or *false*.