[8.6] Convert strings to upper- and lower-case

The two functions below convert an alphanumeric string to all upper-case or all lower-case characters. This is useful, for example, to test strings entered by a user in a program.

Upper-case and lower-case character codes differ by 32, but converting the case is not as simple as adding or subtracting 32, because the string may also include characters which should not be changed. Only characters with lower-case equivalents are converted. In addition, the character set includes international characters, so the conversion routines handle those characters as well

The function to convert a string to lower-case:

```
casel(s)
Func
© (string) convert to lower case
© 7mayØ2/dburkett@infinet.com

local ä,ï,ÿ

""→ÿ

for ä,1,dim(s)
 ord(mid(s,ä,1))→ï
 ÿ&char(when(ï≥65 and ï≤9Ø or ï≥192 and ï≤214 or ï≥216 and ï≤223,ï+32,ï))→ÿ
endfor

return ÿ

EndFunc
```

The function to convert a string to upper-case:

```
caseu(s)
Func
© (string) convert to upper case
© 7mayØ2/dburkett@infinet.com

local ä,ï,ÿ

""→ÿ

for ä,1,dim(s)
 ord(mid(s,ä,1))→ï
 ÿ&char(when(ï≥97 and ï≤122 or ï≥224 and ï≤246 or ï≥248 and ï≤256,ï-32,ï))→ÿ
endfor

return ÿ

EndFunc
```

Some examples:	`caseu("abc")`	returns	`"ABC"`
	`casel("ABC")`	returns	`"abc"`
	`caseu("a+b=C")`	returns	`"A+B=C"`

Both functions work by getting the code of each character in the string, with *ord()*, then adjusting the code by 32 only if the character has a lower-case equivalent, as determined by its code. Note that the *and* operator has higher priority than *or*, so no parentheses are needed.

The execution time in seconds can be estimated as 0.097c + 0.081, where *c* is the number of characters in the string. For example, a 10-character string converts in about 1.05 seconds.