Subject: Re: quick testing of string variables
Posted by steinhh on Wed, 24 Apr 1996 07:00:00 GMT
View Forum Message <> Reply to Message

In article <moninger-2304960900010001@zirkle.fsl.noaa.gov>, moninger@fsl.noaa.gov (Bill Moninger) writes:
|>
|> I have an array called station_name, dimensioned (6,n). Each item is a
|> string 6 characters long. I would like to quickly test station_name
|> against a particular string variable, find_this_station, another string of
|> dimension 6.
|>
|> Is there any way to do this without using loops?
|>
|> If I have to use loops, does anyone have a tip on the fastest way to do so?
|>
|> Is there are better way to configure the array station_name to make such
|> tests (against a particular station name) faster?
|>

It depends a little on what you mean my "testing A against B".
If it means, (as your variable names cleverly point to)
that you want to find out which entry - if any - in station_name
equals the value of find_this_station, then there's hope (I think).
For clarity assume that the strings have length "len", and station_name
has dimension (n1,n2). len, n1 and n2 should be LONGs for safety.

```
byt = byte(station_name)  ; Should be a byte array, dimension (len,n1,n2)

str = string(reform(byt,len*n1*n2,/overwrite)) ; Make a looong string.

pos = strpos(str,find_this_station)

; Of course, if pos eq -1 then nothing was found.
; But if station (0,0) is "AB" and station (1,0) is "AC", they
; will end up in str as "ABAC....".
; If you are looking for station "BA" then you could be in trouble,
; but the "pos mod len ne 0" test discovers this.

if pos eq -1 then print,"No such station found" $
else begin
   ;; We'll have to be careful here:
   if pos mod len ne 0 then print,"Bummer -- wrong number - indecisive"
   x = (pos/len) mod n1
   y = (pos/len)/n1
end

;; You'll find your station in station_name(x,y)
```

---

The problem with AB + AC = ABAC can be fixed by adding a separator character that's never used in the station names, e.g.:

byt = byte(station_name + '@')    ;; (and len = len + 1 !!)

This is of course at the cost of one extra string array copying operation. You might never need it given specific station names.

Regards,

Stein Vidar Hagfors Haugan

---