Subject: Re: problem with call_external()
Posted by steinhh on Thu, 05 Oct 1995 07:00:00 GMT
View Forum Message <> Reply to Message

In article <44va4a$khr@apakabar.cc.columbia.edu>, chs11@inibara.cc.columbia.edu (Carl H Sayres) writes:
|> I'm trying to pass a two dimensional array of floats to an ansi C function.
|> I can successfully pass a 1-d array, but 2-d doesn't want to work.
|> Here's what I'm doing (in an abbreviated form)
|>
|> ;; idlprogram.pro ;;
|> pro idlprogram
|> a=fltarr(400,100)
|> b=fltarr(8)
|> m=long(400)
|> n=long(100)
|> l=long(8)
|> ;some code to put data into a and b...
|> dummy = call_external('cfunc.so','_cfunc',a,b,m,n,l)
|> end
|>
|> /* CFUNC.C
|> */
|> void cfunc(int argc, void **argv)
|> {
|> float ** a;
|> float * b;
|> int m,n,l;
|>
|> a = (float **) argv[0]; /* this doesn't work */

A two-dimensional array is still just an array -- no extra level of indirection is implied. You're treating a as if it was passed to your routine in the form of an array of pointers to each row (or column). This is not the case. Only one pointer is passed, and element a(i,j) can be found in your C program by the expression *(a + i + j*n).

Stein Vidar