Subject: Re: problem with call_external()
Posted by steinhh on Mon, 09 Oct 1995 07:00:00 GMT
View Forum Message <> Reply to Message

In article <4527dn$a8r@apakabar.cc.columbia.edu>, chs11@aloha.cc.columbia.edu (Carl H Sayres) writes:
|>
|> In article <450kgr$2tq@hermod.uio.no>,
|> Stein Vidar Hagfors Haugan <steinhh@amon.uio.no> wrote:
|> >In article <44va4a$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.
|> >|> 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).
|>
|> I'm passing the 2-d array to another function which is expecting a
|> float **. Can I create a *float[], and fill in the values ?
|> eg. for an m by n matrix:
|> float ** list;
|> list = (float **) malloc(sizeof(float *) * m);
|> for(i=0;i<m;i++) list[i] = (a + i*m);
|>
|> will that work?
|>

If I understand your description correctly, the answer is yes.

The way I understand it is that your function expects a pointer
to a table of pointers to floats. Each pointer in the table points
to one row or column in the two-dimensional array.

It could be, however, that your function expects a pointer
to *just one* pointer that points to the whole, contiguous table
of values. In that case, use (float **) &argv[0] as the
argument to the function.

Stein Vidar