Subject: Re: How to send a CHARACTER to a C/C++/FORTRAN function via CALL_EXTERNAL?
Posted by Brian Daniel on Wed, 13 Feb 2013 18:44:34 GMT
View Forum Message <> Reply to Message

A character is one byte. You could try the byte data type defined with the appropriate ASCII value.

On Tuesday, February 12, 2013 8:36:16 PM UTC-5, Bogdanovist wrote:
> I'm having trouble getting a CALL_EXTERNAL call to work. I think the issue is the requirement to send a CHARACTER*1. To be specific, I'm trying to call dgels from LAPACK, see here
>
>
>
> http://www.netlib.org/lapack/explore-html/d7/d3b/group__doub
le_g_esolve.html#ga225c8efde208eaf246882df48e590eac
>
>
>
> I know from previous experience that you have to be careful to get all arguments to CALL_EXTERNAL exactly right, making IDL INTs into LONGs, making sure to get single/double precision etc correct. However this one I can't seem to crack. It doesn't surprise me that the IDL STRING is not equivalent to CHARACTER, but how can I send this correctly?

Subject: Re: How to send a CHARACTER to a C/C++/FORTRAN function via CALL_EXTERNAL?
Posted by Matt Francis on Thu, 14 Feb 2013 05:08:24 GMT
View Forum Message <> Reply to Message

On Thursday, 14 February 2013 05:44:34 UTC+11, Brian J. Daniel wrote:
> A character is one byte. You could try the byte data type defined with the appropriate ASCII value.
>
>
>
> On Tuesday, February 12, 2013 8:36:16 PM UTC-5, Bogdanovist wrote:
>
>> I'm having trouble getting a CALL_EXTERNAL call to work. I think the issue is the requirement to send a CHARACTER*1. To be specific, I'm trying to call dgels from LAPACK, see here
>
>>
>
>>
>
>>
>
>> http://www.netlib.org/lapack/explore-html/d7/d3b/group__doub

le_g_esolve.html#ga225c8efde208eaf246882df48e590eac
>
>>
>
>>
>
>>
>
>> I know from previous experience that you have to be careful to get all arguments to CALL_EXTERNAL exactly right, making IDL INTs into LONGs, making sure to get single/double precision etc correct. However this one I can't seem to crack. It doesn't surprise me that the IDL STRING is not equivalent to CHARACTER, but how can I send this correctly?

Thanks! I've tried using:

trans='N'
c_trans=BYTE(N[0])

but my IDL session still dies when I send this to the routine. There may be other problems with the other arguments though. Do you know if BYTE() will assign the correct ASCII Byte code or do I need to look that up and assign it directly (numerically)?

Subject: Re: How to send a CHARACTER to a C/C++/FORTRAN function via CALL_EXTERNAL?
Posted by Lajos Foldy on Thu, 14 Feb 2013 09:25:10 GMT
View Forum Message <> Reply to Message

On Thursday, February 14, 2013 6:08:24 AM UTC+1, Bogdanovist wrote:
> On Thursday, 14 February 2013 05:44:34 UTC+11, Brian J. Daniel wrote:
>
>> A character is one byte. You could try the byte data type defined with the appropriate ASCII value.
>
>>
>
>>
>
>>
>
>> On Tuesday, February 12, 2013 8:36:16 PM UTC-5, Bogdanovist wrote:
>
>>
>
>>> I'm having trouble getting a CALL_EXTERNAL call to work. I think the issue is the requirement to send a CHARACTER*1. To be specific, I'm trying to call dgels from LAPACK, see here
>

>>
>
>>>
>
>>
>
>>>
>
>>
>
>>>
>
>>
>
>>>   http://www.netlib.org/lapack/explore-html/d7/d3b/group__doub
le_g_esolve.html#ga225c8efde208eaf246882df48e590eac
>
>>
>
>>>
>
>>
>
>>>
>
>>
>
>>>
>
>>
>
>>> I know from previous experience that you have to be careful to get all arguments to CALL_EXTERNAL exactly right, making IDL INTs into LONGs, making sure to get single/double precision etc correct. However this one I can't seem to crack. It doesn't surprise me that the IDL STRING is not equivalent to CHARACTER, but how can I send this correctly?
>
>
>
> Thanks! I've tried using:
>
>
>
> trans='N'
>
> c_trans=BYTE(N[0])
>
>
>

> but my IDL session still dies when I send this to the routine. There may be other problems with the other arguments though. Do you know if BYTE() will assign the correct ASCII Byte code or do I need to look that up and assign it directly (numerically)?

Fortran arguments are passed by reference, try an IDL byte array: BYTE('N').

regards,
Lajos
