Subject: Re: Double precision
Posted by Allan Whiteford on Thu, 19 Mar 2009 10:51:56 GMT
View Forum Message <> Reply to Message

plim.dreaming@gmail.com wrote:

> Well, I can't make sense of it.
> I read the article and if my case is in there I can't find it.
> Seems simple my problem.
> I am reading in a bunch of numbers, the input file has those numbers
> as
> 100.489418 10.512547
> 100.489718 10.512558
> and so on
> I read in those numbers with DOUBLE
> i remove some pairs which don't interest me and the rest I print them
> to an output file and they turn up as
> 100.48942 10.512547
> 100.48942 10.512558
>
> i just want the precision to be the same in the output as in the
> input. same number. that's all i ask. I notice that if I read in
> those numbers as FLOAT instead of DOUBLE then it was even worse. So
> it is true I am assuming that the DOUBLE is truncating/rounding-off/
> whatever.
>
> thanks for the help thus far.
> P
>

Plim,

The numbers you give should be 'good enough' even in single precision, given the following:

pres.txt:

```
100.489418 10.512547
100.489718 10.512558
```

and IDL commands of:

```
        openr,unit,'pres.txt',/get_lun
        readf,unit,a,b
        readf,unit,c,d
        free_lun,unit
        print,a,b,format='(2F9.5)'
        print,c,d,format='(2F9.5)'
```

I get:

```
100.48942 10.51255
100.48972 10.51256
```

which looks fine to me. You are working at the seventh significant figure which is where you'd expect to move to having to use doubles but my implementation of IDL doesn't have a problem with these particular numbers - this issue shouldn't really be implementation dependent. Can you try the above and see what you get?

Maybe the algorithm where you're removing some of the pairs isn't doing exactly what you think it is?

As also pointed out by Chris, you need to be careful that you are actually reading in the numbers as doubles rather than reading them as floats then converting to doubles. Specifically:

```
readf,unit,a
a=double(a)
```

is different from:

```
a=0d0
readf,unit,a
```

(the second example is correct). This issue is covered in David's article in the section headed "[Question:] But, why doesn't x2=Double(443496.984) produce the correct result?" although he doesn't phrase the question specifically in terms of reading from a file.

However, I don't think it's a precision thing causing your problem - try to narrow the problem down by printing back the numbers as soon as you've read them so see if it really is the reading step going wrong.

Thanks,

Allan

> On Mar 18, 10:31 pm, David Fanning <n...@dfanning.com> wrote:
>
>> plim.dream...@gmail.com writes:
>>
>>> Later in the program I calculate the separation between points (x1,y1)
>>> (x2,y2)
>>> And for some of those points the program says that the pairs are the
>>> same. But they are only the same if they are rounded off, the
>>> difference often only shows up in the last 2 decimal places.
>>> ya, i read that link, most of it at least.

>>
>> I hope you aren't comparing floats with the EQ operator.
>> Maybe you should read that article again. All the way
>> to the end this time. :-)
>>
>> You might try Floats_Equal. You will get better results,
>> probably:
>>
>> http://www.dfanning.com/program/floats_equal.pro
>>
>>
>>> One other thing is: lets say the print out is the issue; a case like
>>> you pointed out above, then why is it that if I do:
>>> b=3Dstring(num)
>>> print,b   will give me the rounded off number?
>>
>> It isn't rounding off the number. The number is the
>> number. It is printing the number in 8 significant
>> figures, which is the default format. Give it another
>> format and it will do something else.
>>
>> Cheers,
>>
>> David
>> --
>> David Fanning, Ph.D.
>> Fanning Software Consulting, Inc.
>> Coyote's Guide to IDL Programming:http://www.dfanning.com/
>> Sepore ma de ni thui. ("Perhaps thou speakest truth.")
>
>