Subject: Re: complex arithmetic

Posted by stl on Wed, 06 Apr 1994 06:49:29 GMT

View Forum Message <> Reply to Message

In article <1994Apr5.123735.8305@news.uit.no> royd@zapffe.mat-stat.uit.no (Roy Einar Dragseth) writes:

```
> Why isn't this supported:
> IDL> x = complex(0.,1.)
> IDL> print, x^(1./3.)
> % Operation illegal with complex type.
> % Execution halted at $MAIN$  .
>
> We are running IDL. Version 3.5.1 (hp-ux hp_pa) on a HP9000/755.
>
```

Hi,

neat little problem! I just tested this on my Sparc 10 running Solaris 4.1 with IDL version 3.5.1 and the problem seems even worse then you stated. The following works:

```
IDL> x = complex(0.,1.)
IDL> print,x^(3)
(     -0.00000,     -1.00000
```

but, as soon as you change the print to include a float things blow up:

```
IDL> print,x^(3.)
% Operation illegal with complex type.
% Execution halted at $MAIN$  .
```

does anyone understand this? SHould such an operation even be allowed?

-stephen

--

Stephen C Strebel   /   SKI TO DIE
stl@maz.sma.ch   /   and
Swiss Meteorological Institute, Zuerich   /   LIVE TO TELL ABOUT IT
01 256 93 85   /   (and pray for snow)

Subject: Re: complex arithmetic

Posted by landers on Wed, 06 Apr 1994 13:17:39 GMT

View Forum Message <> Reply to Message

First, let me agree that there's no reason that this kind of thing should not be supported. But...

Of course, you could do:

```
WAVE> x = complex( 0.,1.)
WAVE> print, exp( 3. * alog( x ) )
(  1.19249e-08,     -1.00000)
```

Just a bit of residual error there in the real part....
This kind of technique will handle complex exponents, too.

It would be pretty easy to write a "pow.pro" around this - test for combo of complex arg / non-int expo, and do the log thing.

(disclaimers - I use PV-WAVE - I tested this only lightly - YMMV - etc.)

```
function pow, arg, expo

on_error,2

if n_params() ne 2 then message, 'Usage: result = POW( argument, exponent )'

; argument sizes...
sa = size(arg)
se = size(expo)

; argument types...
ta = sa(sa(0)+1)
te = se(se(0)+1)

; test for structs/strings
if ta ge 7 or te ge 7 then message, 'Illegal data type.'

; check arg,expo combos - use hard way if complex^(float|double|complex)
; or anything^complex
if ( ta eq 6 and te ge 4) or te eq 6 then begin

   ans = exp( expo * alog( arg ) )

endif else begin

   ans = arg^expo

endelse

return, ans
end
```

Subject: Re: complex arithmetic

Posted by jjp on Wed, 06 Apr 1994 13:23:49 GMT
View Forum Message <> Reply to Message

In article <2ntm1pINN1ko@i32.sma.ch>, stl@sma.ch (Stephen Strebel) writes:
|> In article <1994Apr5.123735.8305@news.uit.no> royd@zapffe.mat-stat.uit.no (Roy Einar Dragseth) writes:
|> >Why isn't this supported:
|> >IDL> x = complex(0.,1.)
|> >IDL> print, x^(1./3.)
|> >% Operation illegal with complex type.
|> >% Execution halted at $MAIN$.
|> >
[snip]

This works:

```
 function cpower, z, p
 ; raise a complex number z to a power p
 return,exp(p*alog(z))
 end
```

Hope this helps.

—

Jim Pekar jp2d@nih.gov Disclaimer: Speaking only for myself.
  "A good public library has something to offend everyone."

Subject: Re: complex arithmetic
Posted by thompson on Wed, 06 Apr 1994 14:23:13 GMT
View Forum Message <> Reply to Message

stl@sma.ch (Stephen Strebel) writes:

> In article <1994Apr5.123735.8305@news.uit.no> royd@zapffe.mat-stat.uit.no (Roy Einar Dragseth) writes:
>> Why isn't this supported:
>> IDL> x = complex(0.,1.)
>> IDL> print, x^(1./3.)
>> % Operation illegal with complex type.
>> % Execution halted at $MAIN$.
>>
>> We are running IDL. Version 3.5.1 (hp-ux hp_pa) on a HP9000/755.
>>
> Hi,

> neat little problem! I just tested this on my Sparc 10 running Solaris

```
> 4.1 with IDL version 3.5.1 and the problem seems even worse then you
> stated.  The following works:
> IDL> x = complex(0.,1.)
> IDL> print,x^(3)
> (     -0.00000,     -1.00000

> but, as soon as you change the print to include a float things blow up:
> IDL> print,x^(3.)
> % Operation illegal with complex type.
> % Execution halted at $MAIN$  .

> does anyone understand this?  SHould such an operation even be allowed?
```

I think the problem is that such problems are degenerate--there is more than one correct answer. For example, if we define A and B to be

```
 IDL> A = COMPLEX(1,1)
 IDL> B = COMPLEX(-1,-1)
```

and C to be

and B to be

```
 IDL> C = A^2
 IDL> PRINT, C
 (      0.00000,      2.00000)
```

then A can be thought of as the square root of C. However, so can B, becaus A^2 and B^2 resolve to the same value. Thus, which is the correct answer for C^(0.5)?

Evidently, IDL gets around this ambiguity by not allowing one to calculate a complex number to a non-integer power, even if the floating point number could be simplified to an integer such as in your example above.

Bill Thompson

Subject: Re: complex arithmetic
Posted by salchegg on Wed, 06 Apr 1994 14:36:08 GMT

View Forum Message <> Reply to Message

I tried

```
IDL> z = complex(0.0,1.0)
IDL> print,exp(alog(z)/3)
(     0.866025,     0.500000)
IDL>
```

This is correct $z^{1/3} = e^{\ln(z)/3}$ but unfortunately incomplete.

Because:

Let n be an integer and z a complex number then
$z^{1/n}$ has n solutions in the complex plane (de Moivre !!)

This would mean that IDL should have to make a new array with the solutions:
v = exp(alog(z)/3). This does not happen.
Out of v(0),..,v(n-1) only v(0) is computed.

On the other hand the original question in the first posting told us about problems with the more general problem:

Let u,v be complex numbers.
If we want to compute $z = u^v$ = exp(v Ln(u)), we have to handle the problem with the complex logarithm (Ln):

Ln(u) = ln |z| + i (\varphi_0 + 2k\pi) with k = {0,+/- 1, +/- 2, ...}

and -\pi < \varphi_0 \le \pi

BUT: IDL's alog(z) only computes one value.

```
IDL> z = complex(0.,1.)
IDL> print, exp(alog(z)/3.0)
(     0.866025,     0.500000)
IDL> v = z
IDL> print, exp(v*alog(z))                      i        -pi/2
(     0.207880,      0.00000)      which is    i       = e
IDL>
```

Markus

| Markus Salchegger   University of Salzburg, Austria |
| Research Institute f. Software Technology   (RIST++) |
| email: salchegg@coma.sbg.ac.at |
| <A HREF="http://www.coma.sbg.ac.at/~salchegg/pers.html">WWW</A> |

Subject: Re: complex arithmetic
Posted by isaacman on Wed, 06 Apr 1994 15:37:00 GMT

In article <thompson.765642193@serts.gsfc.nasa.gov>, thompson@serts.gsfc.nasa.gov (William Thompson) writes...

> I think the problem is that such problems are degenerate--there is more than
> one correct answer. For example, if we define A and B to be
>
> IDL> A = COMPLEX(1,1)
> IDL> B = COMPLEX(-1,-1)
>
> and C to be
>
> and B to be
>
> IDL> C = A^2
> IDL> PRINT, C
> (0.00000,   2.00000)
>
> then A can be thought of as the square root of C. However, so can B, because
> A^2 and B^2 resolve to the same value. Thus, which is the correct answer for
> C^(0.5)?
>
> Evidently, IDL gets around this ambiguity by not allowing one to calculate a
> complex number to a non-integer power, even if the floating point number could
> be simplified to an integer such as in your example above.

I don't agree with this at all. IDL has no problem taking the square
root of positive real numbers, even though (-2.)^2 = (2.)^2

Rich Isaacman