[6.48] Use R▸Pθ() for four-quadrant arc tangent function

The built-in arc tangent function, *tan^{-1}()*, cannot return the correct quadrant of an angle specified by x- and y-coordinates, because the argument does not contain enough information. Suppose we want the angle between the x-axis and a ray with origin (0,0) and passing through point (1,1). Since

$\tan(\theta) = \frac{y}{x}$ we have $\tan(\theta) = \frac{1}{1}$ or $\theta = \tan^{-1}(1)$ so $\theta = \frac{\pi}{4}$

However, if the ray instead passes through (-1,-1), we get the same result since (-1/-1) is also equal to 1, but the angle is actually $(-3\pi)/4$. This difficulty was addressed by the designers of the Fortran programming language, which includes a function called *atan2(y,x)* to find the arc tangent of y/x correctly in any of the four quadrants, by accounting for the signs of *x* and *y*.

It is a simple matter to accomplish this in TI Basic with the built-in *R▸Pθ()* function, if we account for the special case (0,0). In fact, it can be done with a single line of TI Basic code:

```
when(x=0 and y=0,undef,R▸Pθ(x,y))
```

undef is returned for (0,0), otherwise *R▸Pθ(0,0)* returns itself in Radian angle mode, and this expression in Degree angle mode:

```
180*R▸Pθ(0,0)/π
```

Neither of these results are useful.

A more elaborate function can be written which also handles list and matrix arguments:

```
atan2(αx,αy)
Func
©(x,y) 4-quadrant arctan(y/x)
©Must be installed in math\
©6jan02/dburkett@infinet.com

local αt,εm,τx                                  © Function name, error message, αx type

"atan2 error"→εm                                © Initialize error message

define αt(α,β)=func                             © Function finds atan2() of simple elements
 when(α=0 and β=0,undef,R▸Pθ(α,β))
endfunc

getType(αx)→τx                                  © Save argument type for later tests

if τx≠getType(αy):return εm                     © Return error if arguments not same type

if τx="LIST" then                               © Handle list arguments
 if dim(αx)≠dim(αy):return εm
 return seq(αt(αx[k],αy[k]),k,1,dim(αx))

elseif τx="MAT" then                            © Handle matrix arguments
 if rowdim(αx)≠rowdim(αy) or coldim(αx)≠coldim(αy):return εm       © Validate dimensions
 return list▸mat(math\atan2(mat▸list(αx),mat▸list(αy)),coldim(αx))

elseif τx="NUM" then                            © Handle numeric arguments
 return αt(αx,αy)

else                                            © Return error for all other arg types
 return εm

endif

EndFunc
```

Both arguments of *atan2()* must be the same type, and must be numbers, lists or expressions. *atan2()* does not work with symbolic arguments.

Typical calls and results in Degree angle mode are:

atan2(1,1)	returns	45
atan2(-1,-1)	returns	-135
atan2({1,-1},{1,-1})	returns	{45,-135}

$$\operatorname{atan2}\left(\begin{bmatrix} 1 & -1 \\ 0 & 3 \end{bmatrix}, \begin{bmatrix} 1 & 1 \\ 0 & \sqrt{3} \end{bmatrix}\right) \quad \text{returns} \quad \begin{bmatrix} 45 & 135 \\ \text{undef} & 30 \end{bmatrix}$$