

## **Pi.bas**

This is a small routine to print pi, the ratio of a circle's circumference to its diameter, to as many places as you wish.

It uses integer math for all values and calculations.  
It'll crank out the first 100 digits in a few seconds.  
After a minute it is upto about 200 digits, and reaches  
300 digits after about 4 minutes but it can go much much  
longer. I've let it run for an hour without crashing  
and it was still churning out digits. :-)

It's output has been compared to John Fisher's pi program  
and they are the same to 1500 digits. John's, much faster,  
trigonometry program can be found at his website:

[As mentioned above, this program does slow down.](#)  
[To see why, change desiredprecision = 50, then](#)  
[print a, a1, b, b1 or better yet, just print their sizes.](#)  
[Each one will be more than a googul \(10^100\).](#)  
[The longer you let it run, the larger they get.](#)  
[After printing 300 digits a, b, a1, b1 are over](#)  
[1000 digits long. That's insane.](#)

[LB's ability to handle such enormous integers is truly remarkable!](#)

```
t0 = time$("ms")  
k = 2 : a = 4 : b = 1 : a1 = 12 : b1 = 4  
digits = 0 : desiredprecision = 300  
-  
while digits<desiredprecision  
' Next approximation  
  p = k*k  
  q = 2*k+1  
  k = k+1  
-  
  olda1 = a1 : oldb1 = b1  
  a1 = p*a+q*a1 : b1 = p*b+q*b1  
  a = olda1 : b = oldb1  
-  
' Print common digits  
  d = int(a / b)  
  d1 = int(a1 / b1)
```

```
while d = d1  
print d;  
digits = digits + 1  
if digits mod 50=0 then print  
a = 10*(a mod b) : a1 = 10*(a1 mod b1)  
d = int(a/b) : d1 = int(a1/b1)  
wend  
wend  
t1 = time$("ms")  
print  
print "Elapsed time was "; (t1-t0)/1000; " seconds."  
end
```

Take care.

- ---  
harmony