

Using Liberty Basic to Calculate Pi with the Bailey-Borwein-Plouffe Algorithm

An interesting exercise (and useful in technical calculations) is the determination of pi to a high degree of precision. One of the simplest of the algorithms for the calculation of pi is the Bailey-Borwein-Plouffe Algorithm.

Editor's note: for more detailed discussion, see Wikipedia [Bailey-Borwein-Plouffe formula](#)

This code segment calculates the value of pi with accurate precision to 100 places. The more iterations performed, the deeper into the irrational number the precision may extend. With 100 places of precision the pi value may be considered accurate for any form of practical engineering or technical calculation.

```
'Program to calculate PI using the Bailey-Borwein-Plouffe Algorithm
Print "How many iterations to calculate (0-100)"
input iterations
pi = 0
if iterations > 100 then goto [end]
for n = 0 to iterations
    pi = pi + (4/((8*n) + 1) - 2/((8*n)+4) - 1/((8*n)+5) -
              1/((8*n)+6)) * (1/16)^n
next n
print using ("#.#####", pi)
[end]input done
```

Frank West
frankdwest@yahoo.com

See also: [\[\[basic:Pi+a+la+modulus|Pi a la modulus\]\]](#) by - [harmony](#)