#Cicada
#RSA N=P*Q
In PyCharm or Python

Abstract

The purpose of this paper is to provide algorithm that is 4 lines of code and that finds P & Q when N is given. It will work for RSA-1024 & RSA-2018 if the computer can float large numbers in PyCharm or Python.

I. RSA-100

Given N below (A 100 digits). To find P and Q. First solve for P by multiplying A*A and then dividing it by A+A. Then continue by dividing by the square root of A that is divided by 2 and 1E48 is added to it. To solve for Q divide A by P. The algorithm will solve for P & Q and print (N,P,Q).

```python
import math

#RSA 100
A=152260502792253336055618378132637429718068114961380688657908494580122963258952897654000350692006139
P=((A*A)/(A+A))/((math.sqrt(A)/2)+1000000000000000000000000000000000000000000000000) #49 digits 99+1/2=49
Q=(A/P)
N=P*Q print 
(N,P,Q)
P=3.7118083904089913e+49 Q=4.102057185540126e+49 N=1.5226050279225333e+99
#Actual P=3797522793694367392280887275545627854565536638199 × Q=40094690950920881030683735292761468389214899724061
```

II. RSA-110

Given N below (A 110 digits). To find P and Q. First solve for P by multiplying A*A and then dividing it by A+A. Then continue by dividing by the square root of A that is divided by 2 and 1E53 is added to it. To solve for Q divide A by P. The algorithm will solve for P & Q and print (N,P,Q).
import math

# RSA 110

A = 3579423147972586877499180783256845540300377802422826193532908190484670252364677411513516111204504060317568667
P = ((A * A) / (A + A)) / ((math.sqrt(A) / 2) + 1000000000000000000000000000000000000000000000000000000000000000000)
# 55 digits 110+1/2=55
Q = A / P
N = P * Q
print (N, P, Q)

OverflowError: integer division result too large for a float P.

# Actual = Q = 61224210904935475769370373175614188422575855425310699 × P = 584641821440615467883655318297916238419861050560106233

# RSA - 2048

A = 251959084756579349402718322404483985714292821262040320277713783604366202070759556264018525880784406918290641249515082189298559149
17618450280848912007284499268739280728777673597141834372076218693675014971824691165077613379859095700973304597488084284017974291006424
586918171951187461215117265463228221686998754918242243363725908514186546204357679842338718477444792073993423658482382428119816381501
0674810451660377306056201619676256133844143603833904414952634432190114657544454178424020924616515723350778707749817125772467962926386
35637328991410156134381381688985040445364023527381951378636564391212019371122822120720357
P = ((A * A) / (A + A)) / ((math.sqrt(A) / 2) + 1000000000000000000000000000000000000000000000000000000000000000000)
# 617 digits/2=308
Q = A / P
N = P * Q
print (N, P, Q)

OverflowError: integer division result too large for a float P.

IV. Conclusion

If your computer can process P, one will get N=P*Q for RSA-2048. This algorithm is 4 lines long and can find P & Q when N is given. The algorithm does have a margin of error but given N, the algorithm will return the approximate P & Q.