 Factorization Batch Algorithm and Prime Numbers Distribution

Shaimaa said soltan1

**1. Algorithm (1) Generate any prime number or its composite using only number [7].**

In Figure (5) we showed that all composite prime numbers are based on branch 7 numbers

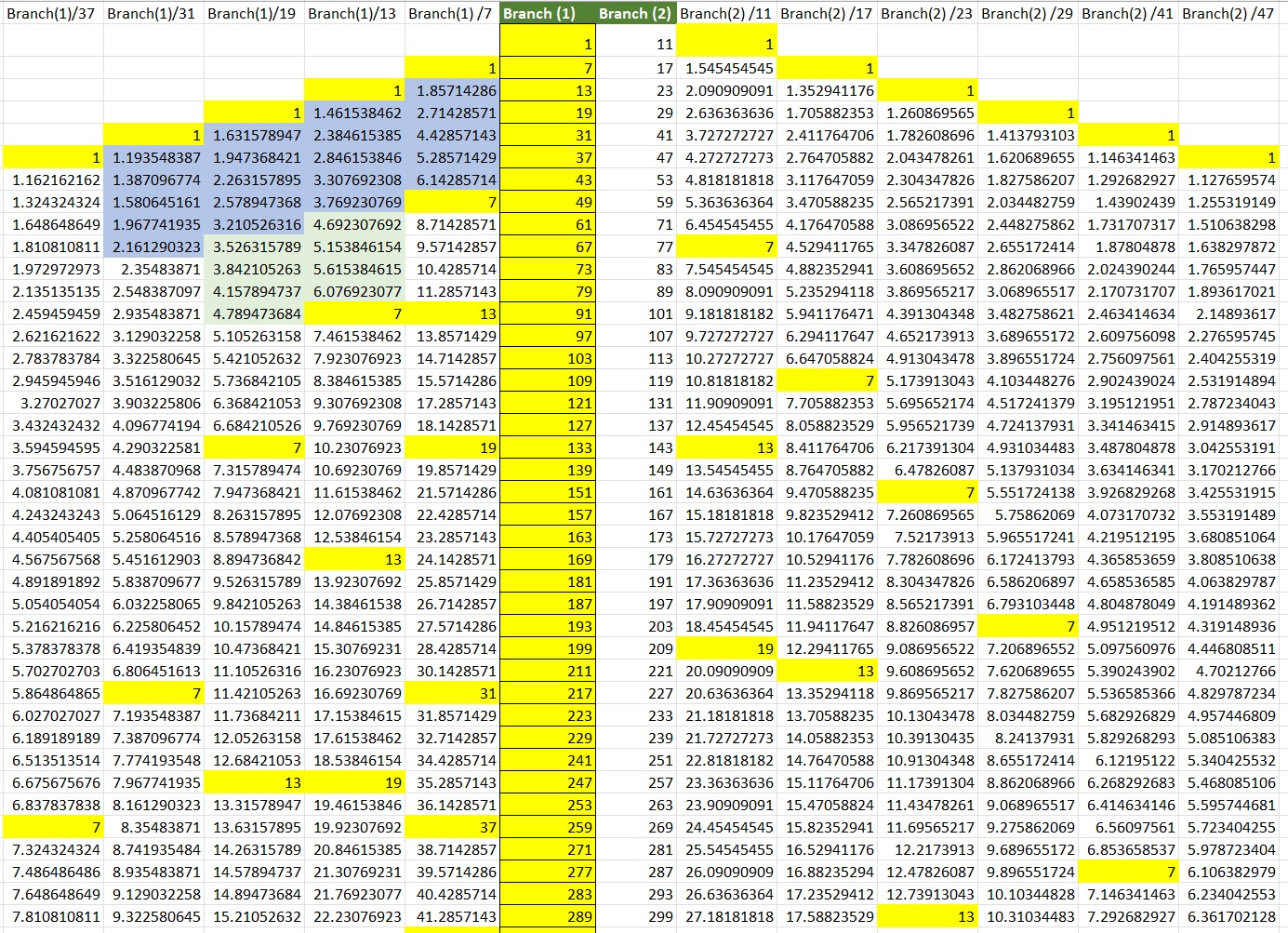
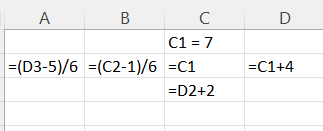
Here we will show why this is the case, and how the branch 7 list is the main list for all rime numbers and how we can reproduce all the branch list from only number [7] and simple operations.

Figure 5. Primary List distributions.



We only need 7 to generate both branches (branch 7 and Branch 11).

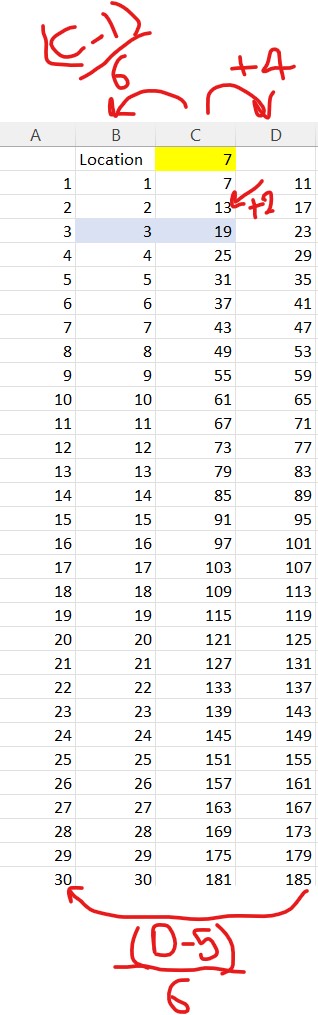


Figure 6. Natural number 7 is the main number for Prime numbers in both branches ( 7 and 11)

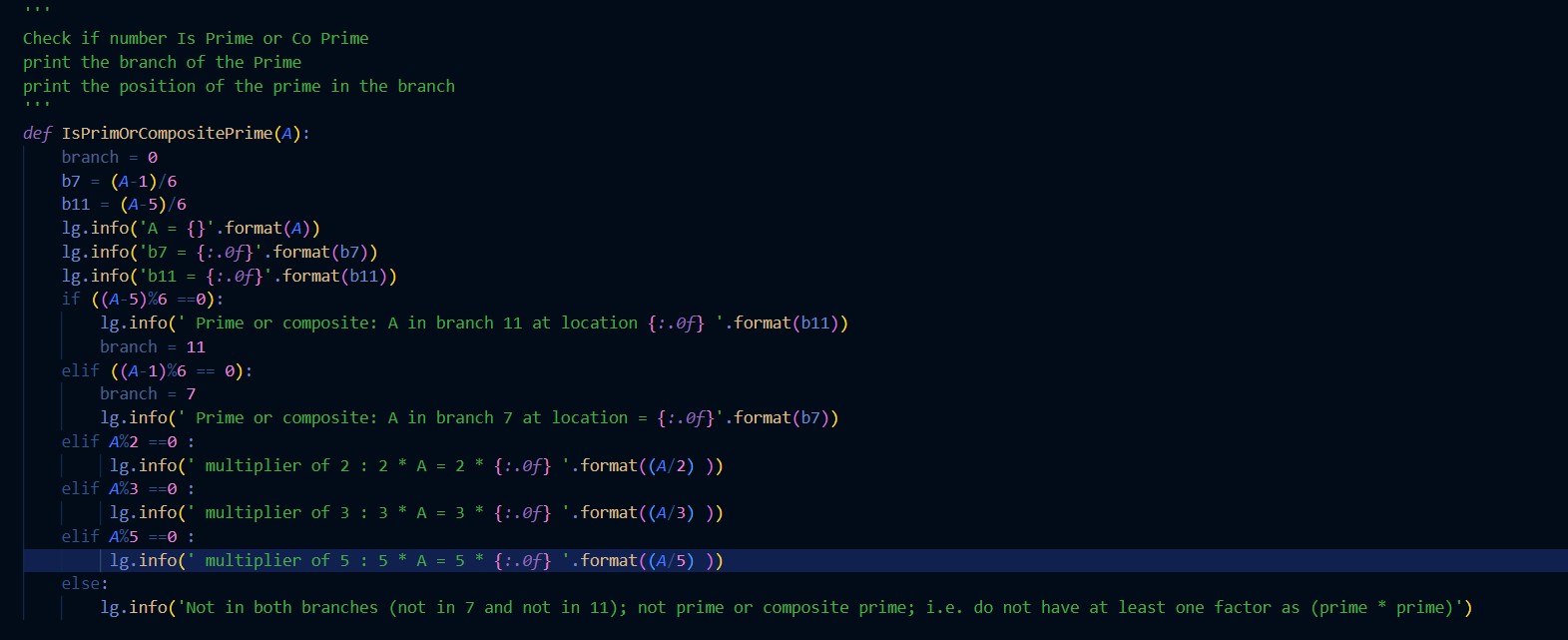
**2. Algorithm (2) Check if number is Prime or not using O (1)**

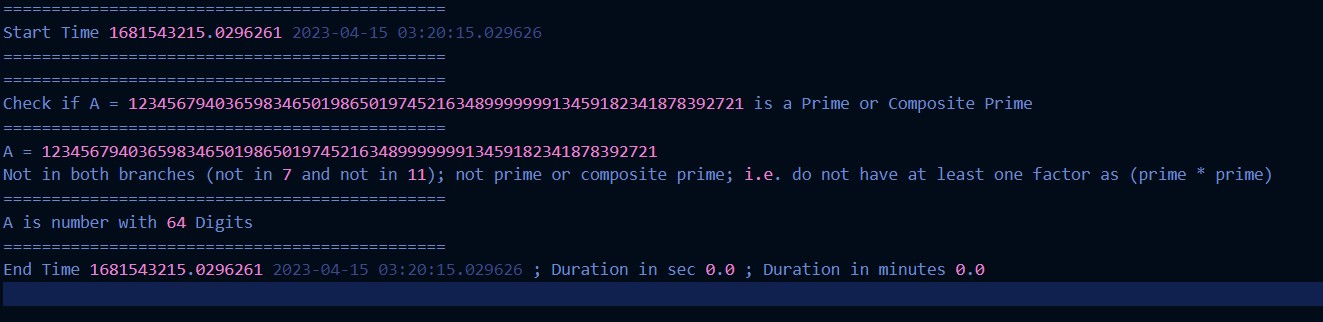
Rule (1):

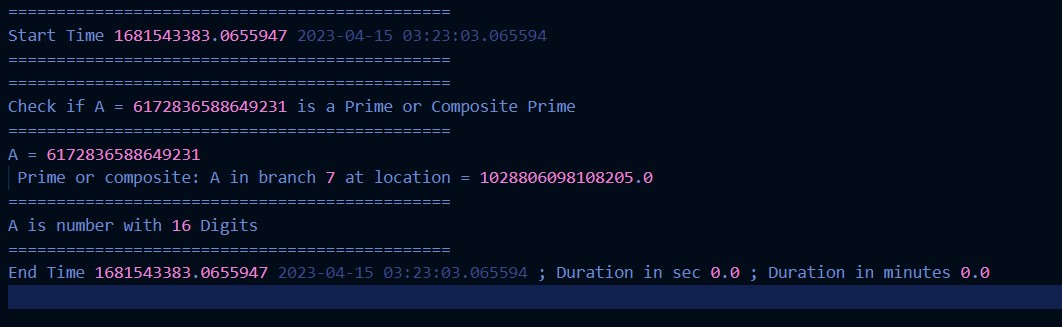
If any number not in both branches (7 and 11) means the number can be divided by (2 or 3 or 5 or 6 or 8 or 9). So, the number is not prime 100%.

And, if number in branches (7 or 11) means the number either Prime of have at least one of its factors will be (prime \* prime).

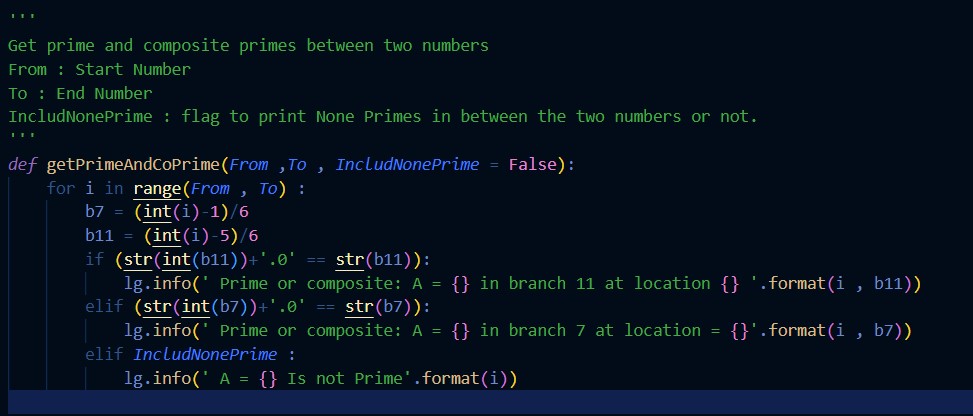
Using this rule, we are going to build algorithm that return accurate results for checking if a number is prime or not with complexity = O (1).

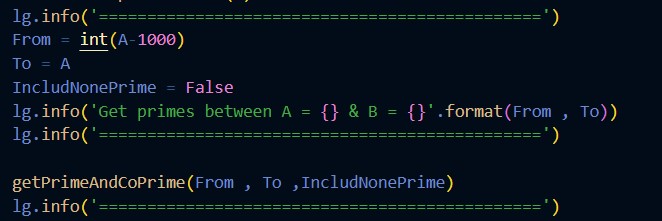
and gives us the location of this number and which branch this number are located in.

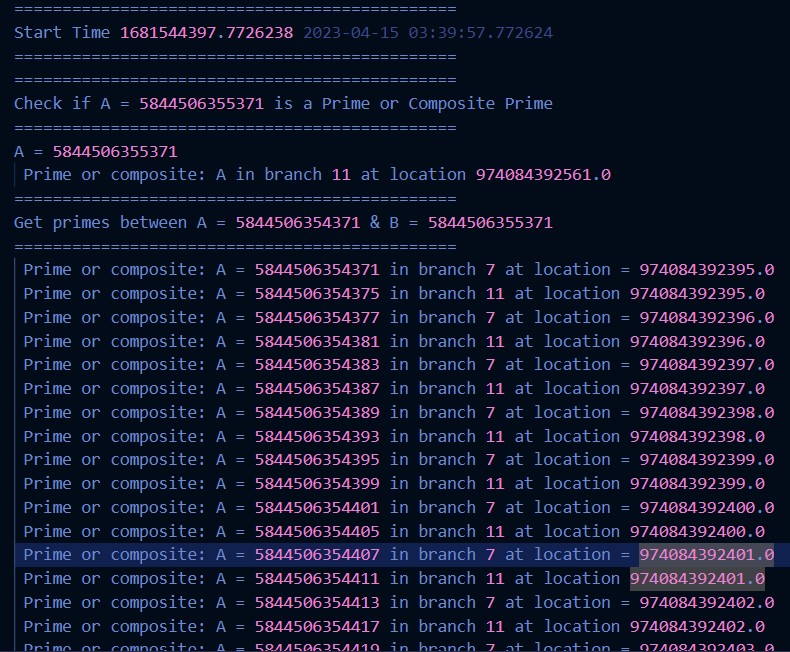
The result for running this algorithm took almost zero sec to check if number with 64 digit is prime or not.

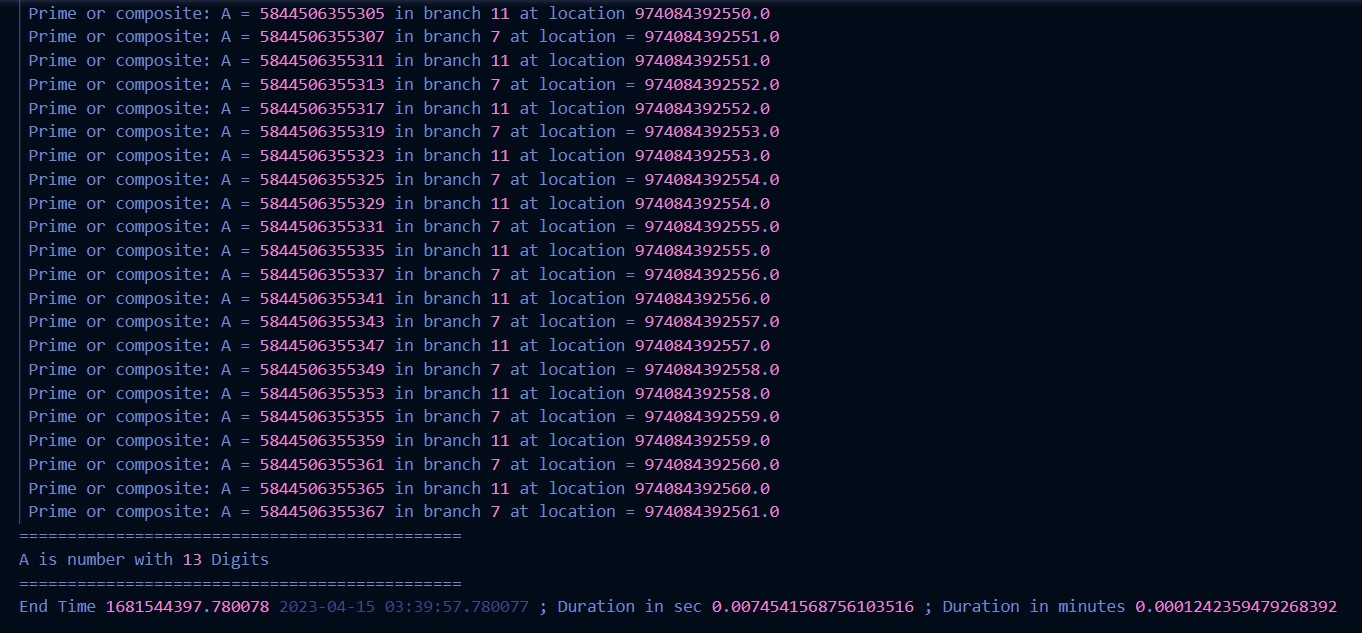
Another example to check number with 16 digits took almost 0 sec and gives us in which branch this number is and the location of this number inside this branch (i.e., the order of this prime inside the branch itself).

**3. Algorithm (3) getting the prime numbers between any two numbers and the locations of these numbers inside the branches (7 and 11)**

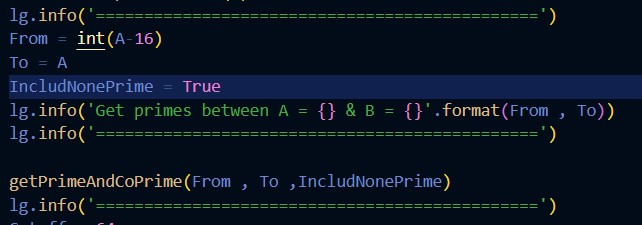


This is an example for the algorithm running with writing only prime and composite primes for number with 13 digits.

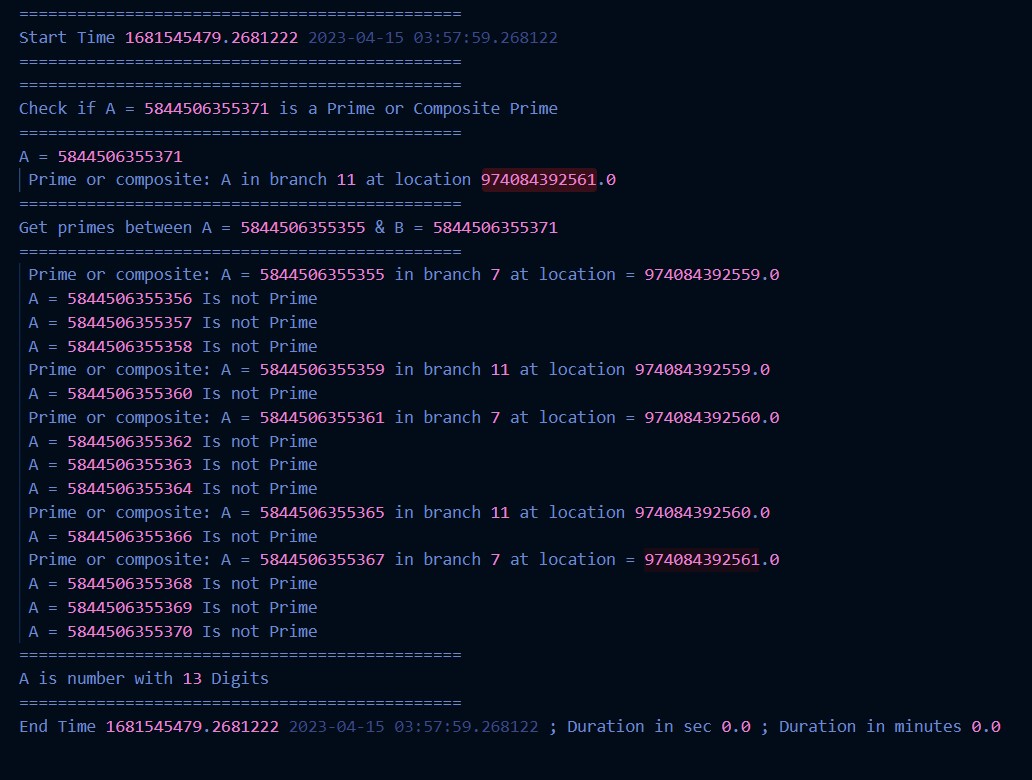


As it shows here to get the list of primes for number with 13 digits and their locations inside each branch it tool less than one second to give us the list using this algorithm.

Another example if we will print all numbers in between including the nonprime numbers. By setting IncludNonePrime = True.

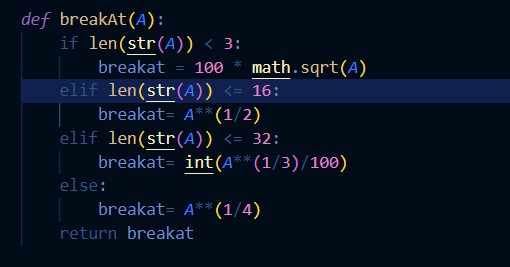


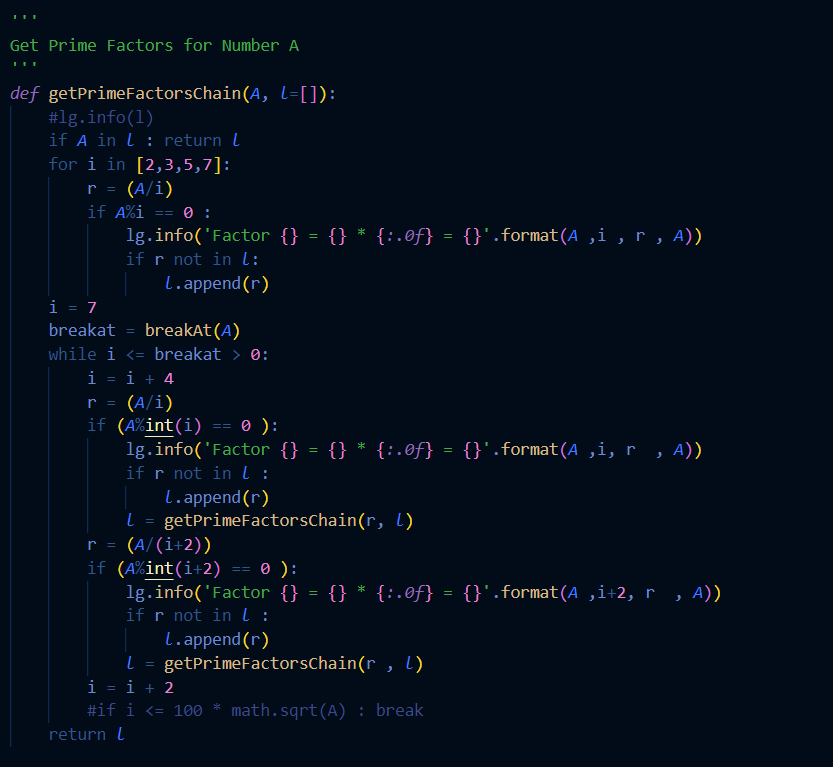
The result took almost Zero second



**4. Algorithm (4) Factorization for prime and composite primes using order O (**

We used a break point based on the number of digits in the number and the root used.

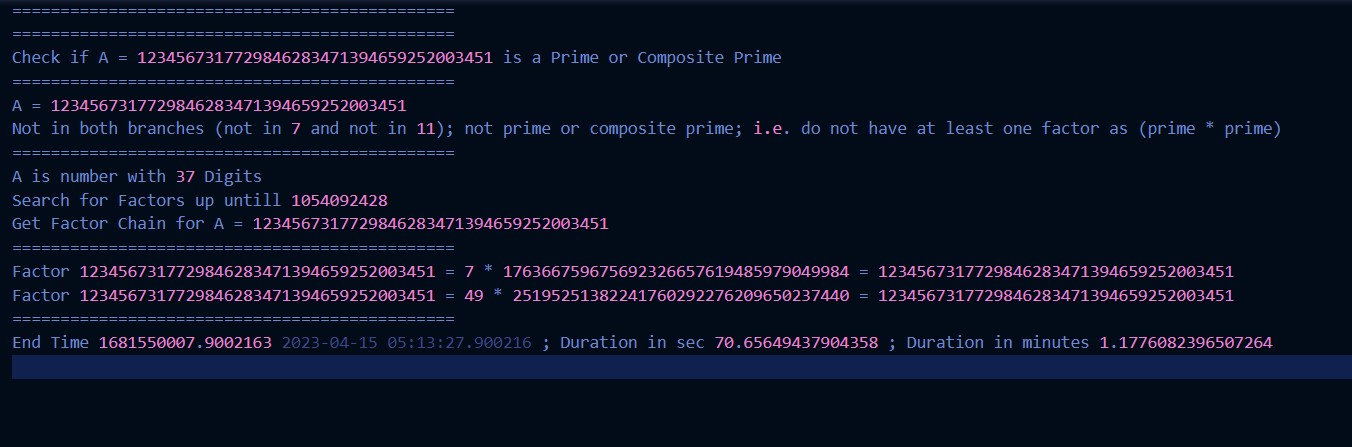
If length <= 16 we used square root and if length <= 32 we used the cubic root other wise we used the fourth root. And these break points can be adjusted based on the ability of the compute power of your machine to get factors up until a break point that can work with you compute machine.



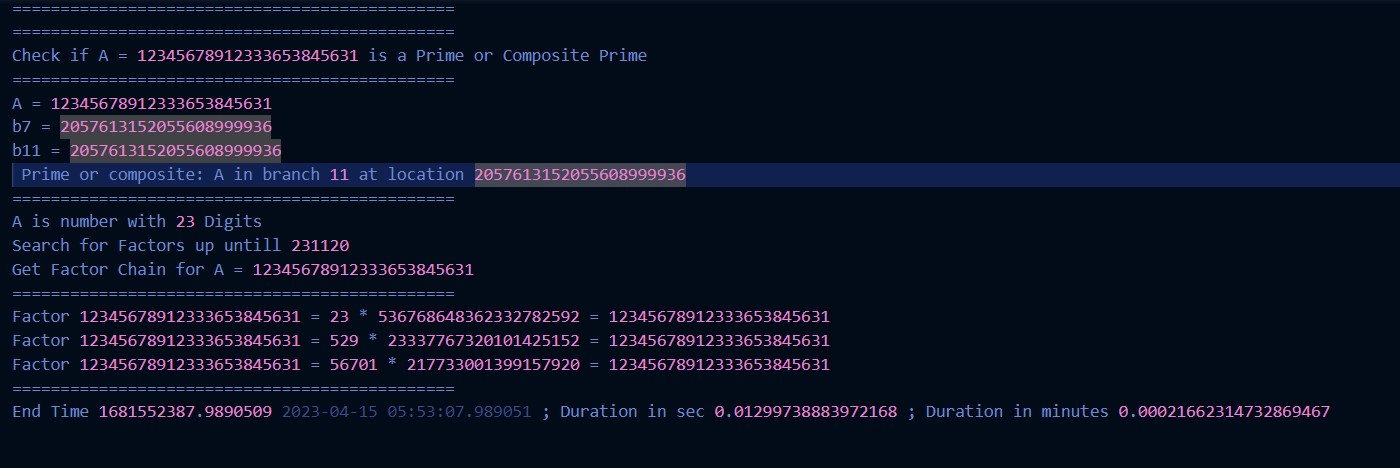
Example 1: Number with 35 digits took 21 sec to get the factors up until its quadratic root



Another Example for number with 37 digits up until its quadratic root in 70 seconds.



Example (3)



Example (4)

