MATH4009 Project 2

October 28, 2021

This project 2 is a group work. You need to work in a group of 2 or 3. Submit your reports along with Python code through Moodle dropbox. Your goal is to create a script that will generate a large prime number.

1 Finding a large prime number

1. Pick a random integer smaller than 10,000 which is relatively prime to $2 \cdot 3 \cdot 5 \cdot 7 \cdot 11 = 2310$. You should do this by generating a random number, n, between 1 and 10,000 and then checking $\gcd(n, 2310) = 1$. If not, you should pick another random number between 1 and 10,000 and repeat the procedure until you find one.
2. Pick a random integer K such that $x = 2310K + n$ is 100-bit long (i.e. $2^{100} < x \leq 2^{101}$). Note that $2^{100} \sim 1.26 \cdot 10^{30}$ so x is about 30-digit long number. Think about the range of K carefully so that x is indeed 100-bit long.
3. Implement Miller-Rabin Primality test to test whether or not x is prime. If not, repeat step 3 with a different value of K until you find a prime. Record which K you used each time.

2 Discussion

1. In the script you wrote, comment each part to explain how the code works.
2. Write a brief report of the result (how many different Ks did you try? which prime number you found? etc.)
3. Using what we discussed in class, talk about the probability of finding a 100-bit long prime number of the form $2310K + n$. How many different K's do you expect to try before finding a prime number?
4. Is your answer from part 3 consistent with what you saw in practice? Explain why or why not briefly.
5. Why do we want to pick a random K amd n? Discuss the importance of picking a random K and n in terms of security assuming we are trying to find large prime numbers to use for RSA cryptosystem.

3 Extra credit

Try to find a very large prime number, at least 1000-bit long. I'll give a gift to whichever group that finds a largest prime number.