RSA Public-Key Cryptosystem

Implementation and Assumptions

I have created a public-key cryptosystem that uses the RSA algorithm. I started my implementation by entering p and q prime numbers from the keyboard which allows the user to select at the beginning. Additionally, I have created a warning message for the user to correct their input if they don't select a prime number or they select prime numbers equal. After that, I followed the procedure of finding n which is multiplication of p and q by using nValue(). In my method, I used the multiply() method of BigInteger. Besides, I implemented phiValue() method to observe totient function that calculates (p-1)*(q-1). As before, I used subtract() and multiply() methods of BigInteger to obtain the desired result. To find e and d values in the cryptosystem, I performed calculateKeys() method which searches the e value in a for loop for the case e is smaller than phi and greatest common divisor of e and phi is 1. I found d value by taking the mod inverse of e and phi with the help of modInverse() method of BigInteger. I provided the user with the opportunity to choose from the pairs by listing all the pairs that meet the conditions I specified. After selecting the e and d values from the list by keyboard, I asked the user to enter the plaintext to the program. By the help of convertInt() method, I converted my plaintext (M) to ASCII. Then, I used the equation $C=M^e \pmod n$ to attain the ciphertext. Similarly, I calculated M by decrypting the ciphertext with the equation $M = C^d \pmod n$. In both operations, I used the modPow() method to take exponential and modulo of values. As a final step, I printed the message, message in ASCII, encryption result, and decryption result to the program. During my implementation, I assumed that the user would not enter large 4-digit prime numbers (etc. 6841, 5003), as the user would select the key pairs e and d from the list. Although the answer is correct, the numbers are so large that the key pairs take too long to calculate and make it difficult for the user to select. Additionally, I assumed that the user selects e and d values from the keyboard by benefiting the list.

Running the program

The user starts the program by entering two unequal prime numbers. Program warns the user in case of selecting a non-prime number. After that, users will be able to see the public (e,n) and private (d,n) key pairs. Then, the user chooses e and d values from the list by typing their values. When the user types the plaintext, all the calculated data is shown in the console such as plaintext message, mathematical representation of message, encrypted/decrypted values and message after decryption.

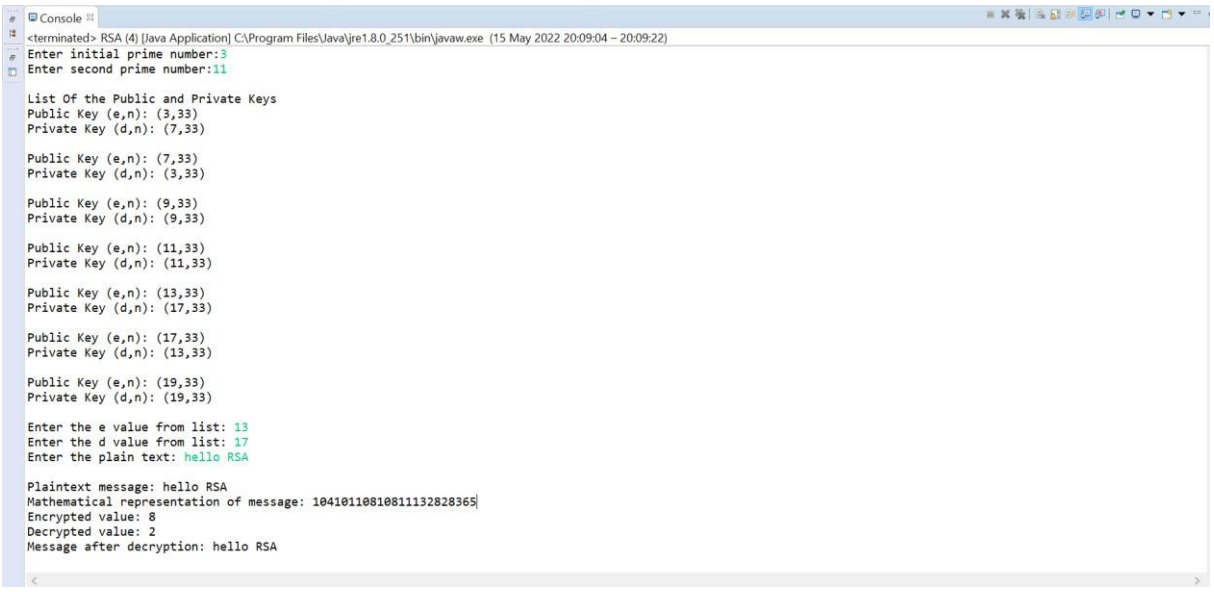

Figure-1: Example run of the program

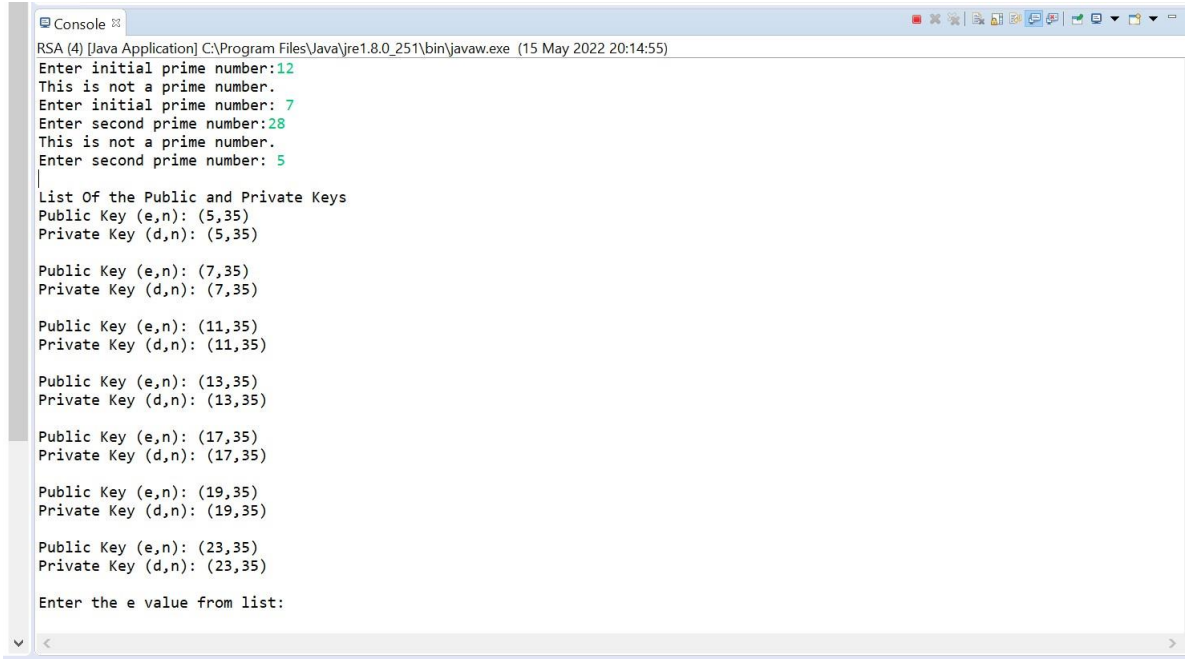

Figure-2: Entering non-prime numbers (produce list according to p=7 and q=5)