ASSIGNMENT 3 RSA

Thom Scholtens 500766849 & Vincent Hoogstra 500756820
T.scholtens@hva.nl & V.hoogstra@hva.nl

Table of contents

Encryption

Step 1

The first step is to fill in a number n. The program then calculates p and q which are two prime numbers that are equal to n when they are multiplied by each other. So: n = pq, where p and q are prime numbers.

The algorithm to calculate p and q works like this:
We first get the smallest prime number, which is 2, for p.
Secondly we get the smallest prime number for q, this is also 2.

We now check if n/q equals p and is not a decimal number.
If this is not true we increase q to the next prime number until it's not smaller then n anymore.
If the next prime number is bigger then n we reset q to 2 and increase p to the next prime number.
If n/q is smaller then q we also increase p and reset q for efficiency.

Step 2

Step 2 is to calculate e. To do this we first need to calculate phi:

$$phi = (p-1)(q-1)$$

Now we can calculate e which is a number between 1 and phi, is prime and the greatets common divider between e and phi is 1. So we generate a random number, until those conditions are true, and that number is e.

Step 3

We start by converting the string to an Integer list. We do this through unicode.

Next we walk over this list and do the item power of e. the result of this we do modulo n which gives us the encrypted number. So we use the following formula:

$$c^e \% n$$

Where:
c is the list item
we generated e in step 2

n was given in step 1.
We put those encrypted values into an array, which is the encrypted message.

Decryption

Step 1

We ask for N and E. we then again calculate p, q and phi as stated above.
after we get those we calculate d, which is done by the following formula:

$$(e^{-1} \% \, phi)$$

Step 2

First we translate the string we get from the input into an Integer arraylist.
then we walk over this array with the following formula.

$$c^{d} \% \, n$$

Where:
c is the list item
we generated d in step 1
n was given in step 1.

then we put the arraylist in a stringbuilder to create a normal output. This will be the decrypted text