1.) 233987973x + 41111687y = 1
x = -11827825
y = 67318298

3.) From the jump we can at least decrypt the message because we have access to Joe Bidens private key. The message states “Put your mask on! There is a deadly pandemic outside!”

```
Welcome to DrRacket, version 7.8 [3m].
Language: R5RS; memory limit: 128 MB.
> (RSA-decrypt received-mystery-message joe-biden-private-key)
"Put your mask on! There is a deadly pandemic outside.   "
>
```

Then I realized, that I would never be able to know who sent it using this function so another function I have that decrypts messages and take a signature as a parameter is the authenticate and decrypt function. So I used make-key-pair as an example and made a signed message using the message from Biden and the signature in the pdf.

```
> (define Biden-signed-message
    (signed-message
     received-mystery-message
     received-mystery-signature))
```

Using this it guess and checked all the public keys until one of them displayed the message I was looking for.

```
> (authenticate-and-decrypt Biden-signed-message donald-trump-public-key  joe-biden-private-key)
Cannot Authenticate
> (authenticate-and-decrypt Biden-signed-message ivanka-trump-public-key  joe-biden-private-key)
Cannot Authenticate
> (authenticate-and-decrypt Biden-signed-message nancy-pelosi-public-key  joe-biden-private-key)
Cannot Authenticate
> (authenticate-and-decrypt Biden-signed-message aoc-public-key  joe-biden-private-key)
Cannot Authenticate
> (authenticate-and-decrypt Biden-signed-message mike-pence-public-key  joe-biden-private-key)
Cannot Authenticate
> (authenticate-and-decrypt Biden-signed-message michael-cohen-public-key  joe-biden-private-key)
Cannot Authenticate
> (authenticate-and-decrypt Biden-signed-message bernie-sanders-public-key  joe-biden-private-key)
Cannot Authenticate
> (authenticate-and-decrypt Biden-signed-message kamala-harris-public-key  joe-biden-private-key)
"Put your mask on! There is a deadly pandemic outside.   "
```

I was able to find that **Kamala Harris** was the sender of the message.

4.)

In order to forge messages on RSA we need both the public and private keys of the people we want to send messages between.

```
(define bernie-sanders-private-key (crack-rsa bernie-sanders-public-key))
(define mike-pence-private-key (crack-rsa mike-pence-public-key))
(define donald-trump-private-key (crack-rsa donald-trump-public-key))
(define ivanka-trump-private-key (crack-rsa ivanka-trump-public-key))
```

Next, we use encrypt and signed with our cracked private key and the known public key, and to read the message from the forged sender we need to crack the recipients private key and use the senders known public key.

The following is a forged message from Donald Trump to Ivanka Trump.

```
> (define trump-signed-message
    (encrypt-and-sign "This is your Dad, Donald Trump. Ivanka, I am so mad at you for making that
horrible investment. Call me now!" donald-trump-private-key ivanka-trump-public-key))
> (authenticate-and-decrypt trump-signed-message donald-trump-public-key ivanka-trump-private-key)
"This is your Dad, Donald Trump. Ivanka, I am so mad at you for making that horrible investment.
Call me now!"
```

The following is a forged message from Bernie Sanders to Mike Pence.

```
> (define sanders-signed-message
    (encrypt-and-sign "Hi Mike Pence, This is Bernie Sanders! if anyone has a real shot at being
president it is me" bernie-sanders-private-key mike-pence-public-key))
> (authenticate-and-decrypt sanders-signed-message bernie-sanders-public-key
mike-pence-private-key)
"Hi Mike Pence, This is Bernie Sanders! if anyone has a real shot at being president it is me"
>
```

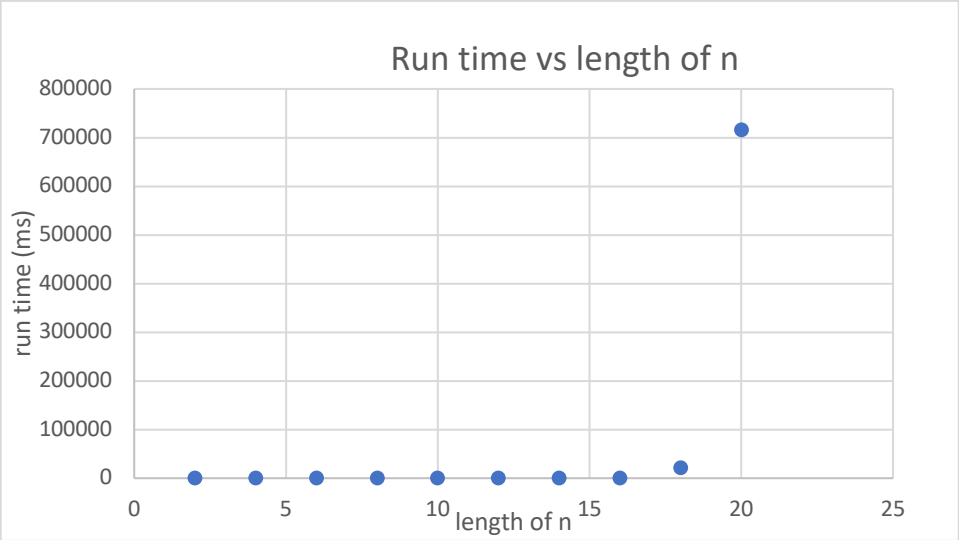
The following is forged message from Ivanka Trump to Bernie Sanders

```
> (define ivanka-signed-message
    (encrypt-and-sign "Hi Bernie Sanders, This is Ivanka Trump, If you let my dad win the
election I will give you 10 million dollars" ivanka-trump-private-key bernie-sanders-public-key))
> (authenticate-and-decrypt ivanka-signed-message ivanka-trump-public-key
bernie-sanders-private-key)
"Hi Bernie Sanders, This is Ivanka Trump, If you let my dad win the election I will give you 10
million dollars  "
>
```

5.) This is the essence of RSA encryption and the reason that it is so hard to crack. To show how long it would take to find the smallest I started with n such, n = pq. The length of n is p + q

```
> (timed smallest-divisor (* 92269 19867))
(time: 1)
19867
> (timed smallest-divisor (* 669913 368363))
(time: 19)
368363
> (timed smallest-divisor (* 4388609 1731913))
(time: 73)
1731913
> (timed smallest-divisor (* 40541309 22845007))
(time: 945)
22845007
> (timed smallest-divisor (* 532153519 632133031))
(time: 22043)
532153519
> (timed smallest-divisor (* 7317115529 4264019483))
(time: 715644)
4264019483
```

The following is a graph and a table that displays the results from the graph, where the length of n is the x-axis and y is the runtime in milliseconds.



length of n	milliseconds
2	0
4	0
6	0
8	1
10	1
12	19
14	73
16	945
18	22043
20	715644

For 18 digits the runtime in milli-seconds is 22,043 which can be converted to about roughly 22 seconds. For 20 digits the runtime is 715644 milliseconds which is roughly about 11 min 20 seconds. The growth is exponential and will continue to rise this way. As you can see there is a huge increase in runtime by only increasing it 2 digits. If we increase p to 12 I suspect runtime would rise past an hour based on the trends above.