The RSA Encryption Scheme

RSA is one of the first practical __________ cryptosystems and is widely used in __________ __________. In such a cryptosystem the encryption key is ________ and differs from the decryption key which is ________. In RSA, this asymmetry is based on the practical difficulty of ______________.
RSA is made of the initial letters of ________, __________, and __________ ________, who first publicly described the algorithm in ______. ____________, an ________ mathematician, had developed an equivalent system in ______, but it was not declassified until ______.

In an RSA scheme any number of people can send each other messages, which ________ be read by ______ other than the __________. Each individual in the scheme finds two __________, p and q. Each person then calculates

$$n = \_\_\_\_\_\_.$$

Then an integer r is chosen that is __________ __________. Finally, s is calculated so that

$$\_\_\_\_\_\_ \equiv \_\_\_\_\_\_\_\_$$

The integers ___ and ___ are made ______ (perhaps each person's ___ and ___ are published in a directory) ______ while keeping ___, ___, and ___ secret.

Now suppose A wants to send a secret message to B let us say the message

JON SKATES

First A must turn the message into a ________.
It doesn't matter how so long as everyone does it the same way, let us simply let $A =$ __, $B =$ __, $C =$ __, ..., $Z =$ __, space = ___. The message is now the number

$$M = \ldots\ldots\ldots\ldots$$

Now ___ looks up ___'s ___ and ___ in the directory and calculates

$$E \equiv \ldots\ldots\ldots\ldots,$$

and sends ___ to ___ as the encoded message.
___ decodes the message E by calculating

$$\ldots\ldots\ldots\ldots$$

using his secret ___. As we shall see

$$\ldots\ldots \equiv M \ldots\ldots$$

so that B recovers the message M.

The description here only works if $M <$ ___; if $M \geq$ ___ then M must be ________ ________. The end ___ gets encoded as ________, and decoded by ________.

Example:

Suppose A finds B's entry in the directory to be $n = 1109635173$7 and $r = 68431529$7. This value of n is too small to use in practice (as we shall see), but it will do to illustrate the scheme. A breaks M into 10 digit pieces:

$M_1 =$ $\qquad\qquad$ $M_2 =$

In general, if the last piece does not contain 10 digits some ____________________. Then using $r = 684315297$ A calculates

$E_1 \equiv$

$E_2 \equiv$

Then A transmits to B the message

$E =$

Now B breaks this up into 10 digit pieces

$E_1 =$ $\qquad\qquad$ $E_2 =$

He then uses his secret ____ which is ____________. to calculate

$M_1 \equiv$

$M_2 \equiv$

Thus B has recovered

$$M =$$

which easily translates to

<u>Theorem</u> (Proof of The Decoding Procedure)
Suppose p and q are distinct _ _ _ _ _.

$$n = \quad ,$$

Suppose the positive integer r satisfies _ _ _ _ _ _ _.
and s is a positive integer such that

_ _ _ _ _ _ _ _ _ _ _ _ _ _.

Let M be a positive integer. Suppose $M <$ _ _
and

$$E \equiv \quad .$$

Then

$$\equiv M \quad (\text{mod} \quad).$$

<u>PROOF:</u> Suppose all the key hypotheses hold.
Since p and q are _ _ _ _ _ _ _,

$$\phi(n) = \quad .$$

Since $rs \equiv \quad (\text{mod} \quad)$, there is
a _ _ _ _ _. integer d such that

_ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _.

We consider 2 Cases.
We must show that

_ _ _ _ _ _ _ _ _ _ _.

Case 1. $(M, n) = 1$. Then

Case 2 $(M, n) > 1$. We may assume without loss of generality that ____ .

Security of the RSA Scheme

If a third party C can ________, that is if he can find ___ and ___, and then he can compute ________, and hence compute the secret key s using the ______________ since

$$\equiv$$

Thus the scheme is not secure unless __________ that C is _______________.

A well chosen 2048 bit n is unlikely to be factored. This means that p and q should be about _____ bits; ie about ____ digits.

Is there a way to break the code without _________?

Nobody has found one (or, if they have, they aren't telling).

Advantages over traditional coding methods

In the usual coding schemes A and B have to agree on a secret key before they can start sending messages. If the communication of the secret key is ___________, then ____________.

In the RSA scheme, _________________.

Suppose k people wish to participate in an information network in such a way that no one can decode messages sent between any two others. Then each pair needs a secret key; each person needs to keep track of ____ different secret keys.

Thus the total number of keys needed is ______.
In the RSA scheme each person has only two keys, his ______ and ______; in all, only ___ keys.

Finding large primes

How do we find a 300 digit prime?
One way is to ______________ and ____________. If it proves composite then ________. How many numbers can you expect to test before finding a prime?
The Prime Number Theorem says that among the integers near x roughly one out of every _____ is prime.
Since $\log(10^{300}) =$ ______ $\approx$ ___, you would expect to test about ___ numbers to find one that is prime.

Some algorithms

To implement the RSA algorithm we need
(1) a method for computing ___;
(2) an efficient method for computing __________.

(1) Modular Inversion Algorithm

Input r and n

Let $s' = 0$ and $s = 1$

while $r > 0$ do

$t \leftarrow n$

$n \leftarrow r$

$q \leftarrow [t/n]$

$r \leftarrow t - rq$

$u \leftarrow s$

$s \leftarrow s' - sq$

$s' \leftarrow u$

Output s'

STOP

When applying the Euclidean Algorithm, we have

$$n = q_1 r + r_1$$
$$r = q_2 r_1 + r_2$$
$$r_1 = q_3 r_2 + r_3$$
$$\vdots$$
$$r_m = q_{m+1} r_{m+1} + r_{m+2}$$
$$r_{m+1} = q_{m+3} r_{m+2} + 0$$

Where $r_{m+2} = 1$

at each stage we have

$$r_1 = 1 \cdot n - q_1 r$$
$$\vdots$$
$$r_j = (\cdot)n + s'r$$
$$r_{j+1} = (\cdot)n + sr$$
$$r_{j+2} = r_j - q_{j+2} \cdot r_{j+1}$$
$$= (\cdot)n + s'r - q_{j+2}((\cdot)n + sr)$$
$$= (\cdot)n + (s' - q_{j+2} s) r$$

Initially $(\cdot) = (\cdot)n + \_\_\_ r$

$r = (\cdot)n + \_\_\_ r$

$r_1 = (\cdot)n + \_\_\_\_ r$

So initially $s' = \_\_\_$

$s = \_\_\_\_\_\_\_$

We find that when the algorithm finishes

$=$

and

The Power Algorithm

We describe an efficient method for computing _____.
First we describe an efficient method for computing powers.
As an example suppose we wish to compute a^{82} for some given a. The naive way would be to do __ multiplications but this is clearly inefficient.
The idea is to convert 82 to binary

$82 =$

so that

$82 =$

We compute powers of a by doing repeated ____:

$a^2,$

This takes __ multiplications

Then we complete the computation of a^{82} by doing __ more multiplications:

$a^{82} =$

This gives a total of __ multiplications instead of __. This leads to the following algorithm:

<u>The Power Algorithm</u>

```
Input a and N
Let j = 1, Q = 1, Y = 1 and n = N.
    while n > 0 do
        if n is odd then
            b ⟵ 1
            if n = N then
                Q ⟵ a
                Y ⟵ a
            else
                Q ⟵ Q^2
                Y ⟵ Q * Y
        else
            b ⟵ 0
            if n = N then
                Q ⟵ a
            else
                Q ⟵ Q^2
        n ⟵ [n/2]
Output Y
STOP
```

The input of this algorithm is ____

The output is ___.

To compute $a^N \pmod{m}$ requires a simple modification of the Power Algorithm. We need a function that does reduction mod m. One needs to define a function whose input is _ _ _ _ _ _ and whose output is the _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _:

$$\mathrm{mod}(a, m) = \text{\_ \_ \_ \_ \_ \_ \_ \_ \_ \_}.$$

The Power Algorithm mod m

```
Input a, N and m
Let j = 1, Q = 1, Y = 1 and n = N.
    while n > 0 do
        if n is odd then
            b ⟵ 1
            if n = N then
                Q ⟵ a
                Y ⟵ a
            else
                Q ⟵ mod (Q^2, m)
                Y ⟵ mod (Q * Y, m)
        else
            b ⟵ 0
            if n = N then
                Q ⟵ a
            else
                Q ⟵ mod (Q^2, m)
        n ⟵ [n/2]
Output Y
STOP
```

The input of this algorithm is _ _ _ _ _ _ _ _ _ _ _ _ _ _.
The output is _ _ _ _ _ _ _ _ _ _ _ _ _ _ _.

2014-08-01-125745

Dr.F.G. Garvan

8/2/2014

Contents

```
# RSA EXAMPLE
# Suppose Alice wants to send Bob the message
# JON SKATES
# First Alice converts this is a number using spce=00, A=01,
# B=02, C=03, ..., Z=26
# M = 10 15 14 00 19 11 01 20 05 19
# Then Alice looks up Bob's public n and r:

n=11096351737
r=684315297

# Then she breaks up M into blocks less than n:

M1=1015140019
M2=1101200519

# Alice next calculates M1^2 and M2^r mod n

mod(M1^r,n)
Error in lines 1-1
Traceback (most recent call last):
  File "/projects/17ab5038-8f40-4654-a1b9-176371ed4c00/.sagemathcloud/sage_server.py",
line 736, in execute
    exec compile(block+'\n', '', 'single') in namespace, locals
  File "", line 1, in <module>
  File "integer.pyx", line 1994, in sage.rings.integer.Integer.__pow__
(sage/rings/integer.c:14677)
  File "c_lib.pyx", line 73, in sage.ext.c_lib.sig_raise_exception (sage/ext/c_lib.c:872)
KeyboardInterrupt

# We see that using the mod function is not practical. We
# need to use the power_mod function (which naturally uses the power mod \
    algorithm)

E1=power_mod(M1,r,n);E1
9974872058

E2=power_mod(M2,r,n);E2
```

```
7148290747

# Also then encodes the message as
# E = E1 . E2 = 99748720587148290747
# Bob decodes the message by calculating E1^s and E2^s mod n using his \
   secret s.
#

# Instead we can find Bob's secret s by factoring n = p*q
# After finding p,q we find s    that r*s == 1 mod ( (p-1)*(q-1))
factor(n)
104729 * 105953

p=104729
q=105953

SL=xgcd(r,(p-1)*(q-1)); SL
(1, 657804897, -40567793)

s=mod(657804897,(p-1)*(q-1));s
mod(r*s,(p-1)*(q-1))
s=Integer(s)
657804897
1

# Hence we find Bob's secret s = 657804897

D1=power_mod(E1,s,n);D1
1015140019

D2=power_mod(E2,s,n);D2
1101200519

D1==M1
D2==M2
True
True

# Observe that M1 == E1^s mod n and M2 == E2^s mod n
#  and we are able to find M1, M2, M and the original message.
```

Extra Credit Assignment

(a) Using the usual alphabet encoding encode the phase

SEND MONEY

as three 8 digit numbers:

$M_1 = 19051404$, $M_2 =$, $M_3 =$

(b) Now use the RSA encryption scheme to encode M_1, M_2, M_3

with $n = 536813567$, and $r = 3602561$

to obtain three numbers ($< n$):

$E_1 = 463099189$, $E_2 =$, $E_3 =$

(c) Break the code by factoring n & obtaining the secret key s.

Explain clearly all your reasoning, work and detail of computations.

Check that E_1 decodes as M_1 etc showing details.