**Cryptography and Coding Homework 6**

**Vedasagar Karthykeyan (vkarthy1)**

**Problem 2:**

**fast\_exponentiation.m**

function [ ans ] = fast\_exponentiation( n, a, k )

c = 0;

bin\_arr = [];

l = k;

while l ~= 0

    bin\_arr = [bin\_arr mod(l, 2)];

    l = floor(l/2);

end

m = mod(a, n);

arr = [m];

for i=bin\_arr(2:end)

    m = mod(m^2, n);

    arr = [arr m];

end

p = 1;

c = 1;

for j=bin\_arr

    if j == 1

        p = p \* arr(c);

    end

    c = c+1;

end

ans =  mod(p,n);

end

Output:

fast\_exponentiation(17, 5, 2631)

ans =

10

fast\_exponentiation(13, 7, 256)

ans =

9

fast\_exponentiation(19, 5, 117)

ans =

1

**Problem 4:**

**rsa.m**

n = 8439833;

e = 5711029;

c = 62472;

prime\_factors = [];

for i=2:n/2

    if mod(n, i) == 0

        prime\_factors = [prime\_factors i];

    end

end

phi\_n = (prime\_factors(1) - 1) \* (prime\_factors(2) - 1);

d = invmodn(e, phi\_n);

m = fast\_exponentiation(n, c, d)

Output:

rsa

m =

834300

**Problem 5:**

**mod\_square\_roots.m**

function [ m ] = mod\_square\_roots( c, p, q )

n = [p q];

c\_1(1) = fast\_exponentiation(p, c, (p+1)/4);

c\_1(2) = mod((p\*q) - c\_1(1), p);

c\_2(1) = fast\_exponentiation(q, c, (q+1)/4);

c\_2(2) = mod((p\*q) - c\_2(1), q);

m(1) = mod(crt(n, [c\_1(1), c\_2(1)]), p\*q);

m(2) = mod(crt(n, [c\_1(2), c\_2(1)]), p\*q);

m(3) = mod(crt(n, [c\_1(1), c\_2(2)]), p\*q);

m(4) = mod(crt(n, [c\_1(2), c\_2(2)]), p\*q);

end

**rabin.m**

c = 6245706;

n = 9353881;

m = factor(n);

mod\_square\_roots(c, m(1), m(2))

Output:

rabin

ans =

1443540 1234567 8119314 7910341

diary off