1(i)

(3, 8) works because C = aM + b (mod 26).

And 3 and 26 are coprimes

(ii)

(8,3) does not work as 8 and 26 are not coprimes

(iii)

(5,0) works as 5 and 26 are coprimes

(iv)

(0,5) does not work as 0 and 26 are not coprimes

(v)

(13,1) does not work as 13 and 26 are not coprimes

2

for(int i=0;i<cipher.length();i++){

int n = 0;

char decrpt = 0;

char q = cipher.charAt(i);

int m = (int)q;

//System.out.println("int m: " + m);

if (cipher.charAt(i) != ' ')

{

if(m>96&&m<123)

{

n=m-97;

}

if(m>64&&m<91)

{

n=m-65;

}

int x=0;

int inv = 0;

for(int j=0;j<26;j++)

{

x=(a\*j)%alphabet;

if(x==1)

{

inv=j;

}

}

int cipherNum = mod((inv)\*(n-b),alphabet);

//System.out.println("int mod: " + mod);

if(m>96 && m<123)

{

int combined = cipherNum+97;

System.out.println(combined);

decrpt= (char)(combined);

plaintext = plaintext + decrpt;

}

if(m>64 && m<91)

{

int combine = cipherNum+65;

System.out.println(combine);

decrpt=(char)(combine);

//System.out.println(decrpt);

plaintext = plaintext + decrpt;

}

}

else

{

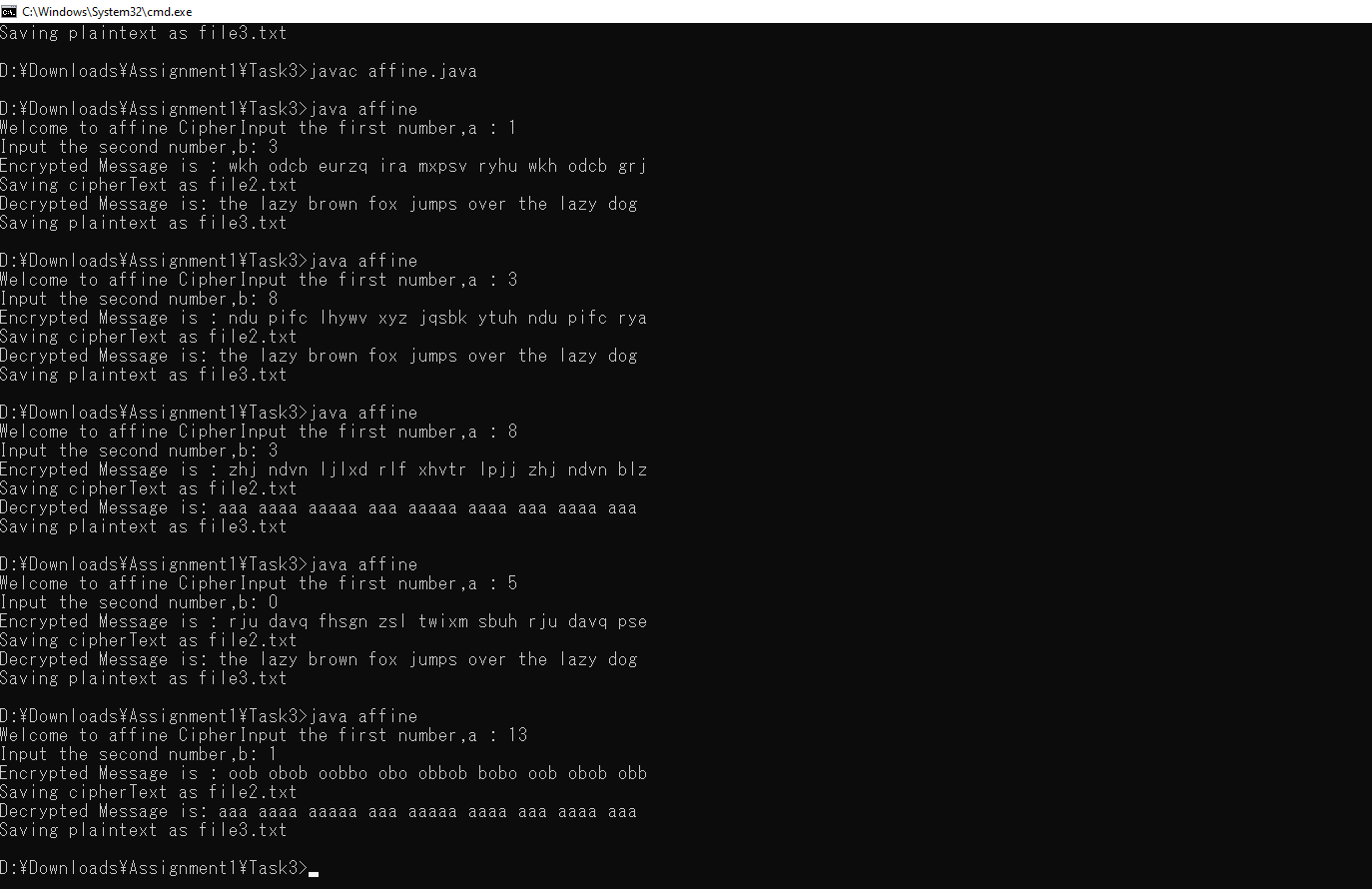
plaintext += cipher.charAt(i);

}

3

Inside affine.java

4

s