Homework4

October 3, 2020

[1]:
```
########## Preamble: Important functions from last homeworks

def textToInt(words):
    number = 0
    i = 0
    for letter in words:
        number += ord(letter)*(256**i)
        i+=1
    return number

def intToText(number):
    words = ""
    while number>0:
        nextLetter = number % 256
        words += chr(nextLetter)
        number = (number-nextLetter)/256
    return words

def fastPowerSmall(g,A,N):
    a = g
    b = 1
    while A>0:
        if A % 2 == 1:
            b = b * a % N
        A = A//2
        a = a*a % N
    return b

def extendedEuclideanAlgorithm(a,b):
    u = 1
    g = a
    x = 0
    y = b
    while true:
        if y == 0:
            v = (g-a*u)/b
            return [g,u,v]
```

```
        t = g%y
        q = (g-t)/y
        s = u-q*x
        u = x
        g = y
        x = s
        y = t

def findInverse(a,p):
    inverse = extendedEuclideanAlgorithm(a,p)[1] % p
    return inverse
```

[10]:
```
########## Problem #1

##### Parts (a) and (b)

def generatePublicKey(a):
    return fastPowerSmall(g,a,p)

def elgamalEncrypt(m,A):
    k = ZZ.random_element(2,p-2)
    c1 = fastPowerSmall(g,k,p)
    B = fastPowerSmall(A,k,p)
    c2 = m*B % p
    return [c1,c2]

def elgamalDecrypt(c1,c2,a):
    x = findInverse(fastPowerSmall(c1,a,p),p)
    return x*c2 % p

##### Part (c)

p = 787
g = 34

#secret key:
a = 99

#generate the public key
A = generatePublicKey(a)

#what's bob's message?
m = 314

#encrypt it using the public key A
cipherText = elgamalEncrypt(m,A)
```

```
#Then decrypt it:
decodedM = elgamalDecrypt(cipherText[0],cipherText[1],a)

#did it work?
print("Bob's message was",m,"and Alice decoded",decodedM,'\n')

###### Part (d)
#Here's the input

p= 753022235974397591242683563886842009117
g = 47393028462819284673
aliceSecret = 314159265358979323846
c1 = 449164960684688587557185888310931655332
c2 = 608713686463403616105013668689979824341

#let's decrypt it
decodedM = elgamalDecrypt(c1,c2,aliceSecret)

#and turn it into text:
decodedMText = intToText(decodedM)

#print it:
print("Decoded message from Gabriel in part (d):")
print(decodedMText,'\n')

##### Part (e)
#Here's the public key:
A = 418194837551245918495968754919547251501

messageAsText = "Secret Message!"

#turn it into a number
message = textToInt(messageAsText)

#then encode it
c = elgamalEncrypt(message,A)
print("Cipher text for Gabriel in part (e):")
print(c)
```

```
Bob's message was 314 and Alice decoded 314

Decoded message from Gabriel in part (d):
Can you hear me?

Cipher text for Gabriel in part (e):
[675399536838517445364979160971361258767,
115265361588385863423034203759220882351]
```

[0]: