Erikson Sodergren CS462-01

1.
    a. $4^{-1} = 4^5$ mod 7 = $4*2^2$ mod 7 = 2
    b. $\Phi(12) = 4$, 5^3 mod 12 = 5*1 mod 12 = 5 mod 12
    c. $6^{-1} = 6^{11}$ mod 13 = $6*10^5$ mod 13 = $6*10*9^2$ mod 13 = 6*10*3 mod 13 = 11
2. $\Phi(6) = 2\{5, 1\}$, $\Phi(9) = 6\{8, 7, 5, 4, 2, 1\}$
    a. $a^2 = 1$ mod 6 for{1, 5}.
        i. 5*5 = 25 = 1mod6, 1*1 = 1mod6
        ii. 2*2 = 4mod6, 3*3=3mod6, 4*4=2mod6
    b. $A^6 = 1$ mod 9 for {1, 2, 4, 5, 7, 8}
        i. 1^6=1mod9, 2^6=64=1mod9, $4^6=(2^6)^2$=1mod9, $5^6=25^3$=7*4=1mod9, $7^6=4^3$=54=1mod9, $8^6=(2^6)^3$=1mod9
        ii. 3^6=0mod9, 6^6=0mod9
3. $a^{-1} = a^{\Phi(26)-1}$mod26 = a^{12-1}mod26 = a^{11}mod26
4. 39^{39} mod 773
    a. $39_{10}=100111_2$
    b.

step	Square	Mul	bit
1	1	39	1
2	39^2=748		0
3	748^2=625		0
4	625^2=260	260*39=91	1
5	91^2=551	551*39=618	1
6	618^2=62	62*39=**99**	1

5. result of 1234567^2345678 mod 3333337 is: 3078688

```
# -*- coding: utf-8 -*-
"""
Created on Mon Apr  1 17:16:23 2019

@author: Erikson
"""

def sqAndMul(base, exp, mod):
    result = 1
    binlist = [int(x) for x in '{:b}'.format(exp)]
    for x in binlist:
        result**=2
        result %=mod
        if x is 1:
            result *= base
            result %=mod
    return result


def main():
    base = 1234567
    exp = 2345678
    mod = 3333337
    print("result of %s^%s mod %s is: %s"%(base, exp, mod, sqAndMul(base, exp, mod)))



if __name__ == "__main__":
    main()
```