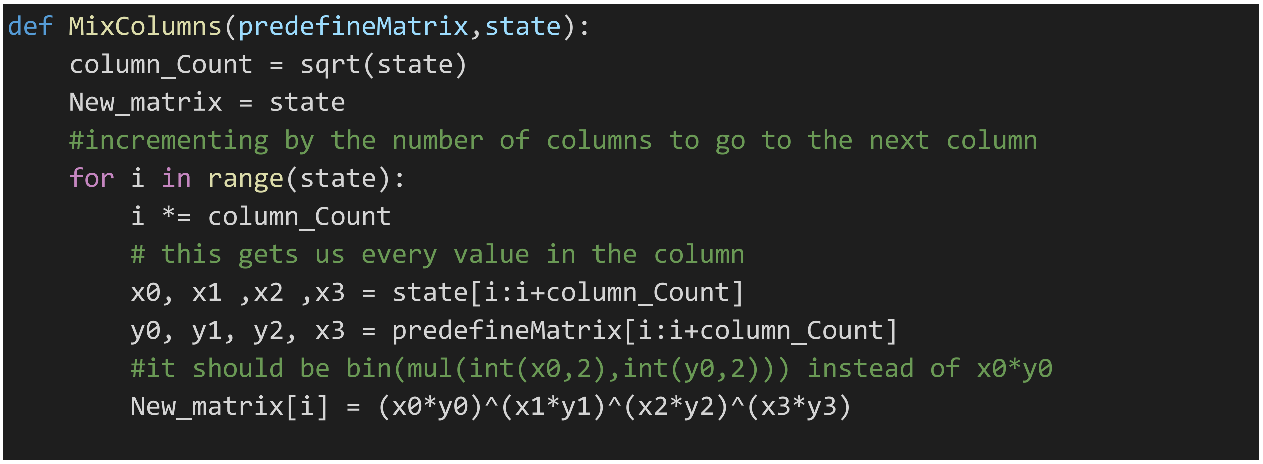
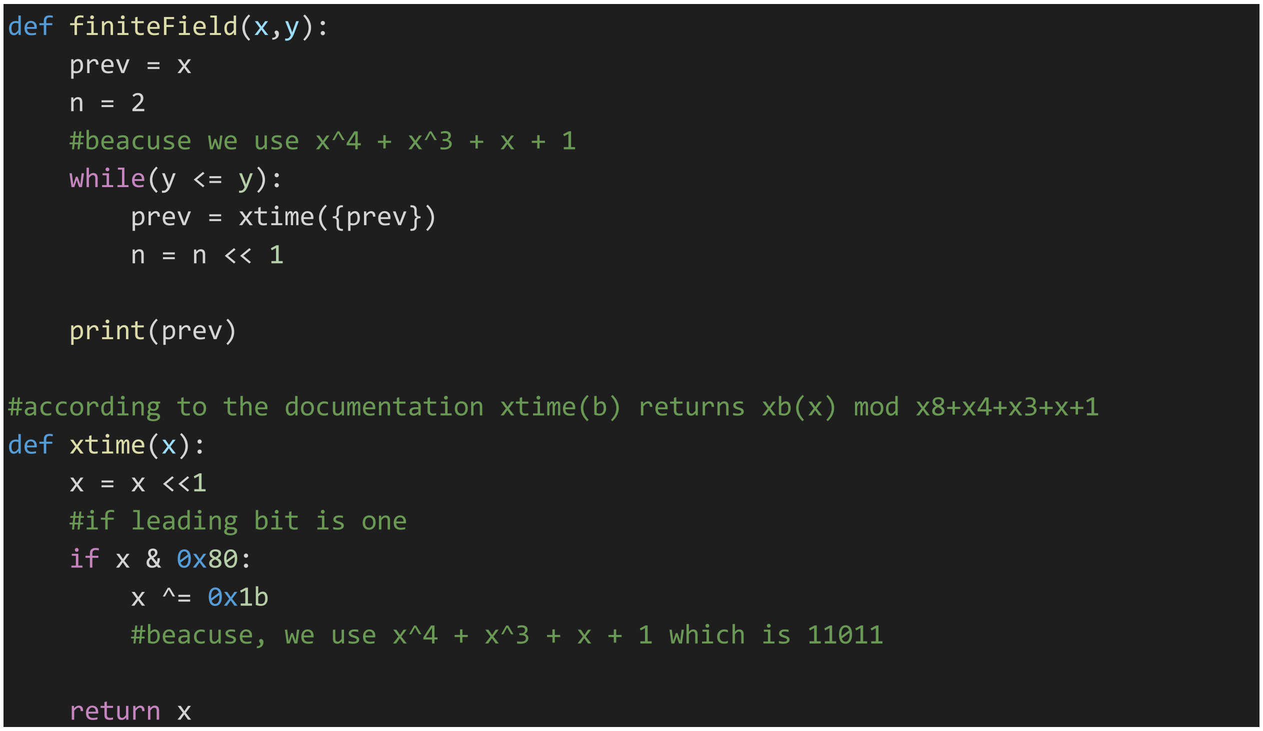
Abdurhman Bahour
Homework: AES
9/6/2022

Problem 1: Submit pseudocode for the MixColumns function in 5.1.3. Your submission should demonstrate that you understand how to implement the ideas in section 4. Study these sections and the lecture slides to be able to distinguish between the abstract mathematical ideas (e.g., polynomial representation) and the implementation methodology (i.e., bit shifting, MOD, AND, XOR).



```
def MixColumns(predefineMatrix,state):
    column_Count = sqrt(state)
    New_matrix = state
    #incrementing by the number of columns to go to the next column
    for i in range(state):
        i *= column_Count
        # this gets us every value in the column
        x0, x1 ,x2 ,x3 = state[i:i+column_Count]
        y0, y1, y2, x3 = predefineMatrix[i:i+column_Count]
        #it should be bin(mul(int(x0,2),int(y0,2))) instead of x0*y0
        New_matrix[i] = (x0*y0)^(x1*y1)^(x2*y2)^(x3*y3)
```

Problem 2: Submit pseudocode for a finite field multiply function that takes two bytes as input and produces a byte as output. Multiplication is done in the finite field used by AES. This process is described in section 4.2. Your code should use an xtime function that is described in Section 4.2.1.



```
def finiteField(x,y):
    prev = x
    n = 2
    #beacuse we use x^4 + x^3 + x + 1
    while(y <= y):
        prev = xtime({prev})
        n = n << 1

    print(prev)

#according to the documentation xtime(b) returns xb(x) mod x8+x4+x3+x+1
def xtime(x):
    x = x <<1
    #if leading bit is one
    if x & 0x80:
        x ^= 0x1b
        #beacuse, we use x^4 + x^3 + x + 1 which is 11011

    return x
```