# POIS ASSIGNMENT 1

## TASK 4

## USE THE PRF TO BUILD A SECURE MAC

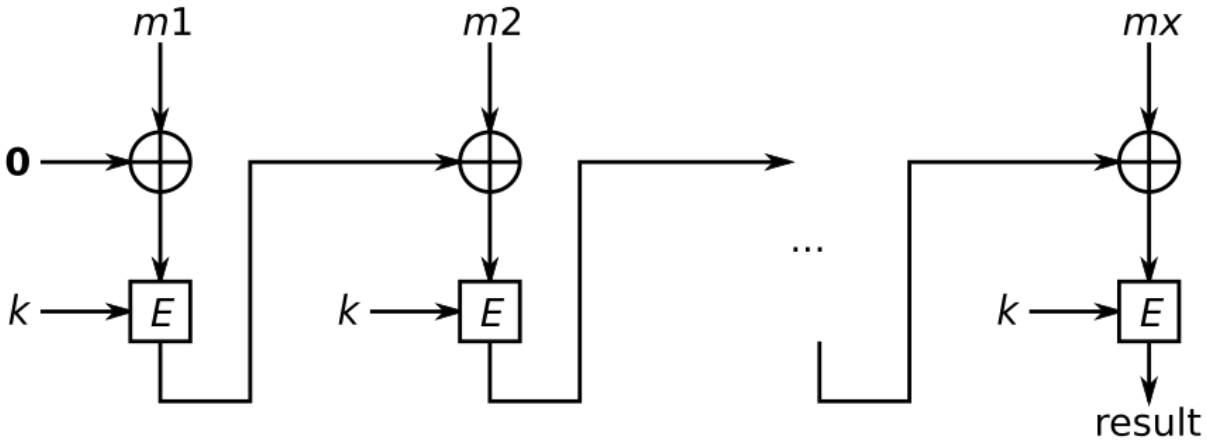
## CODE

```
def cbc_mac(msg, key=K):
    msg = msg_to_binary(msg)
    msg_len = len(msg)
    len_bin = dec_to_bin(msg_len).zfill(n)

    prf = PRF(len_bin, K)

    for i in range(0,msg_len,n):
        msg_block = msg[i:i+n]
        if len(msg_block)!=n:
            msg_block = msg_block.ljust(n,"0")
        xor = dec_to_bin(int(msg_block,2) ^ int(prf,2)).zfill(n)
        prf = PRF(xor, K)
    return prf
```

To calculate the CBC-MAC of message *m*, one encrypts *m* in CBC mode with zero initialization vector and keeps the last block. The following figure sketches the computation of the CBC-MAC of a message comprising blocks using a secret key *k* and a block cipher *E*




## OUTPUT :