Name: Muhammad Sharjeel Akhtar day / date:

Roll No: 20p-0101

Assignment: Merkle-Damgard

Section: BCS-7B

I.S

Submitted To Respected Sir = Dr. Muhammad Amin

```
import hashlib

blocksize = 64

initial_hash = hashlib.sha256(b"").digest()

def merkle_damgard(message):
    message = message + b'\x80'
    while len(message) % block_size != (block_size -
                                        (block_size-8)):
        message += b'\x00'

    message += (8 * len(message)).to_bytes(8,
                                           byteorder = 'big')
    hash_state = list(initial_hash)
    for i in range(0, len(message), block_size):
        block = message[i:i+block_size]
```

KAGHAZ www.kaghaz.pk

Scanned with CamScanner

```
for j in range(len(hash_state)):
    block_word = int.from_bytes(block
        [j*4:(j+1)*4],
        byteorder='big')
    hash_state[j] = (hash_state[j] +
    block_word) & 0xFFFFFFFF

final_hash = bytes()
for word in hash_state:
    final_hash += word.to_bytes(4,
        byteorder='big')
return final_hash
message = b"This is a simple message for
    hashing using the Merkle-Damgard
    construction."
hashed_message = merkle_damgard(message)
print("Hashed Message:", hashed_message.hex())
```

So we have used hashlib library. We have generated initial hash using sha256. After it, we have written the merkle-damgard algorithm which takes message as variable and return hashed message.

KAGHAZ www.kaghaz.pk

Scanned with CamScanner