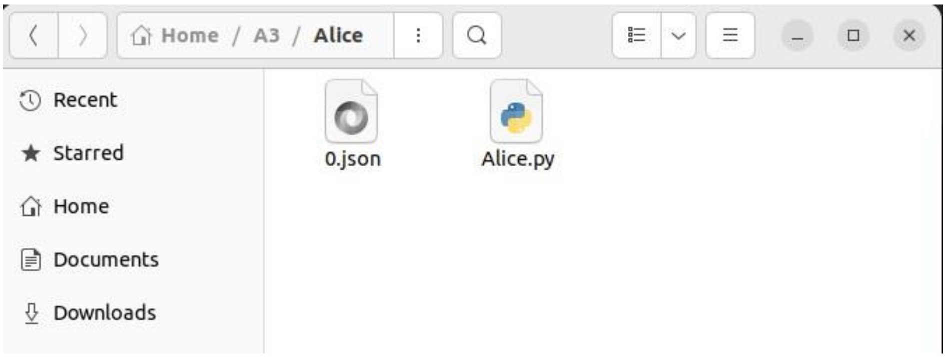
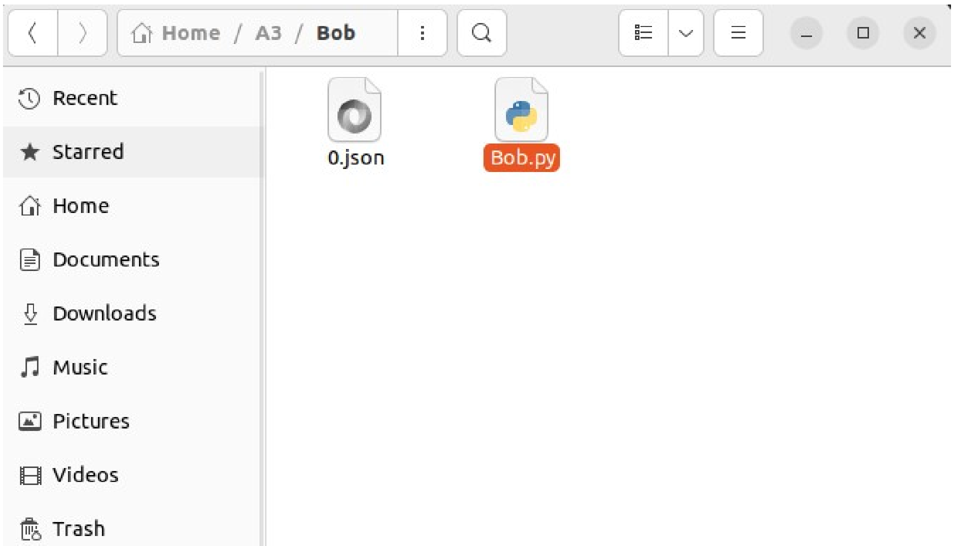
Bronson Chan
CSCI301 Assignment 3 Report
Alternative 2

Starting

2 different folders (Alice and Bob) both starting with each of their python program and also the genesis block 0.json

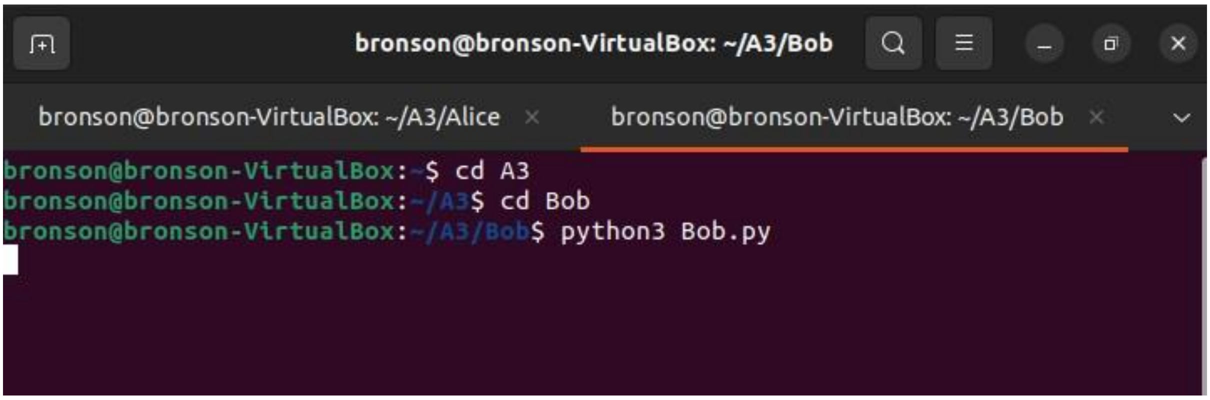




Running programs

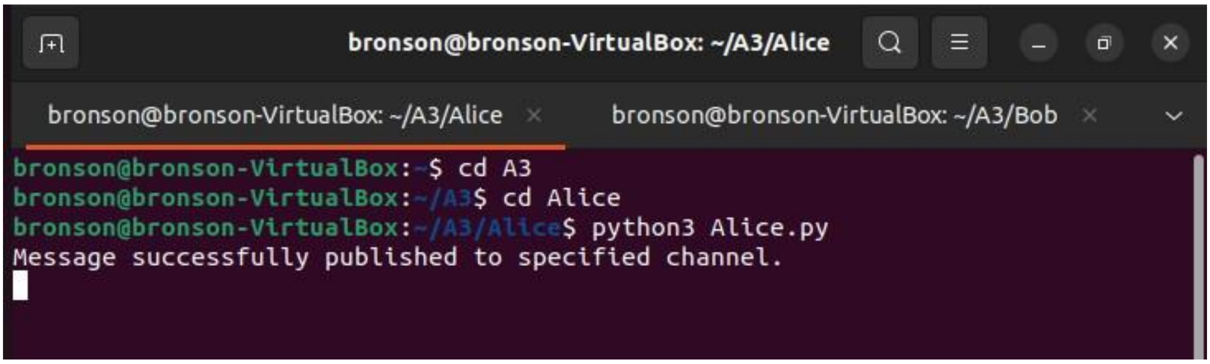
First, we will open 2 terminal and run Bob.py first because I have set Bob.py to start mining only after Alice has mined block 1.

1. Run python3 Bob.py


bronson@bronson-VirtualBox: ~/A3/Bob
bronson@bronson-VirtualBox: ~/A3/Alice
bronson@bronson-VirtualBox: ~/A3/Bob

```
bronson@bronson-VirtualBox:~$ cd A3
bronson@bronson-VirtualBox:~/A3$ cd Bob
bronson@bronson-VirtualBox:~/A3/Bob$ python3 Bob.py
```

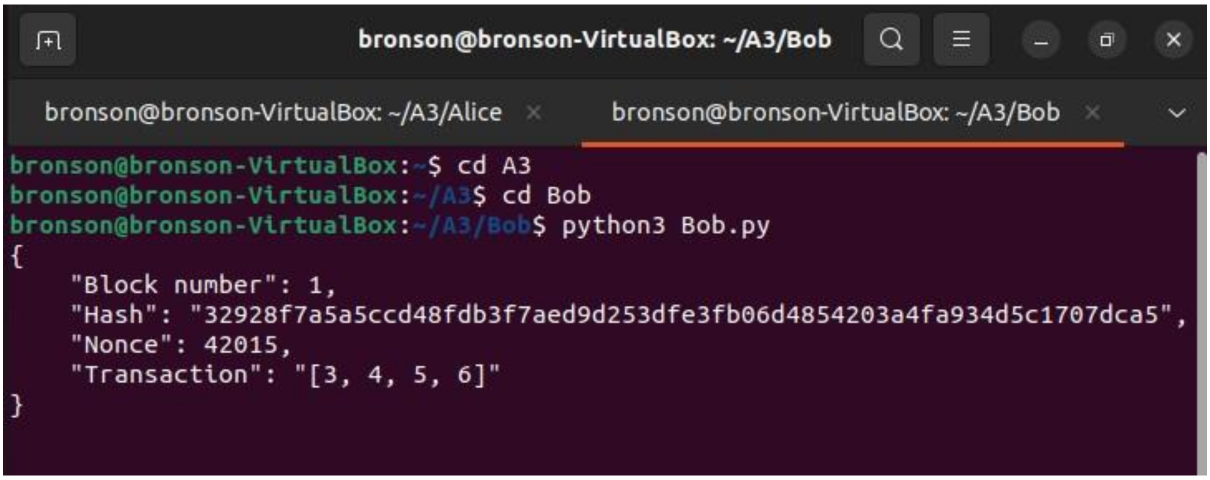
2. Run python3 Alice.py on a different terminal (Alice will immediately start mining block 1)
3. After Alice mined block 1, it will save it into 1.json and send block 1 over to Bob.


bronson@bronson-VirtualBox: ~/A3/Alice
bronson@bronson-VirtualBox: ~/A3/Alice
bronson@bronson-VirtualBox: ~/A3/Bob

```
bronson@bronson-VirtualBox:~$ cd A3
bronson@bronson-VirtualBox:~/A3$ cd Alice
bronson@bronson-VirtualBox:~/A3/Alice$ python3 Alice.py
Message successfully published to specified channel.
```

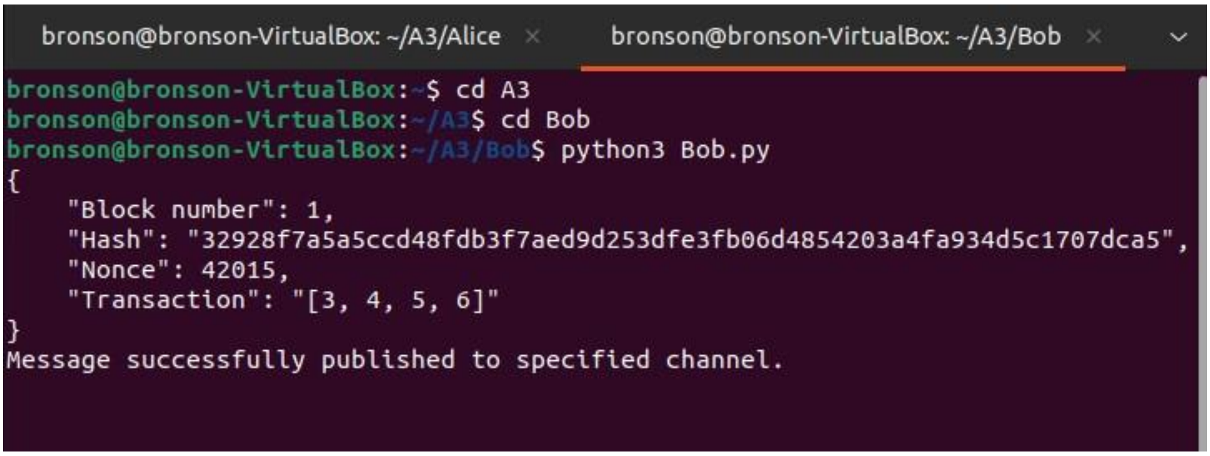
“Message successfully published to specified channel” means the block is mined and published as shown below.

4. Bob will then save it as 1.json in his own folder too.


bronson@bronson-VirtualBox: ~/A3/Bob
bronson@bronson-VirtualBox: ~/A3/Alice
bronson@bronson-VirtualBox: ~/A3/Bob

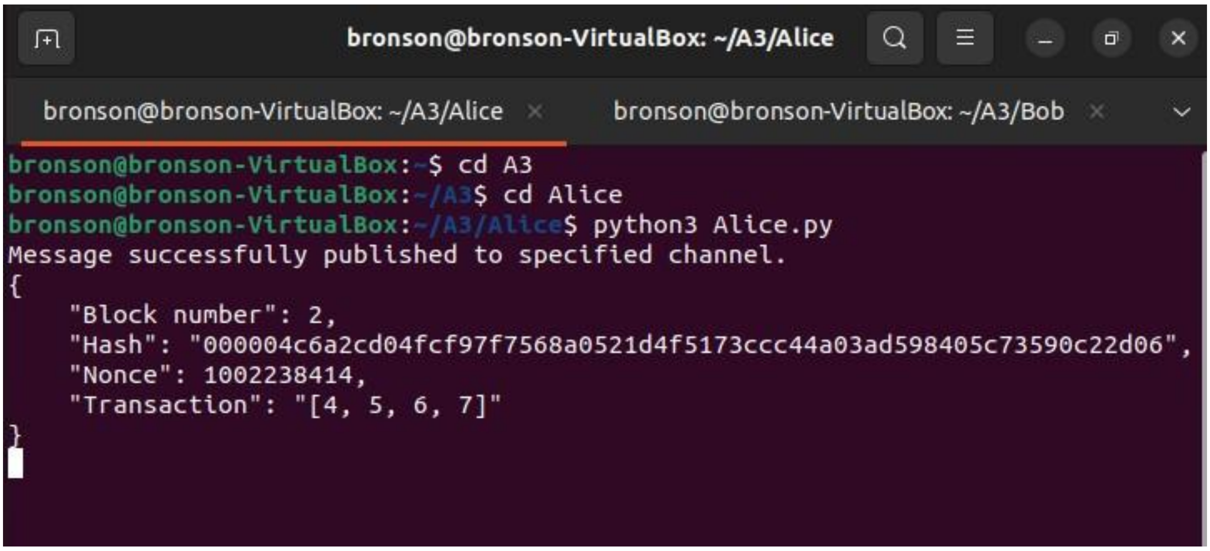
```
bronson@bronson-VirtualBox:~$ cd A3
bronson@bronson-VirtualBox:~/A3$ cd Bob
bronson@bronson-VirtualBox:~/A3/Bob$ python3 Bob.py
{
    "Block number": 1,
    "Hash": "32928f7a5a5ccd48fdb3f7aed9d253dfe3fb06d4854203a4fa934d5c1707dca5",
    "Nonce": 42015,
    "Transaction": "[3, 4, 5, 6]"
}
```

5. Bob will verify block 1 before starting to mine block 2 based on the hash of block 1.
6. After Bob mined block 2, it will save it into 2.json and send block 2 over to Alice.


bronson@bronson-VirtualBox: ~/A3/Alice
bronson@bronson-VirtualBox: ~/A3/Bob

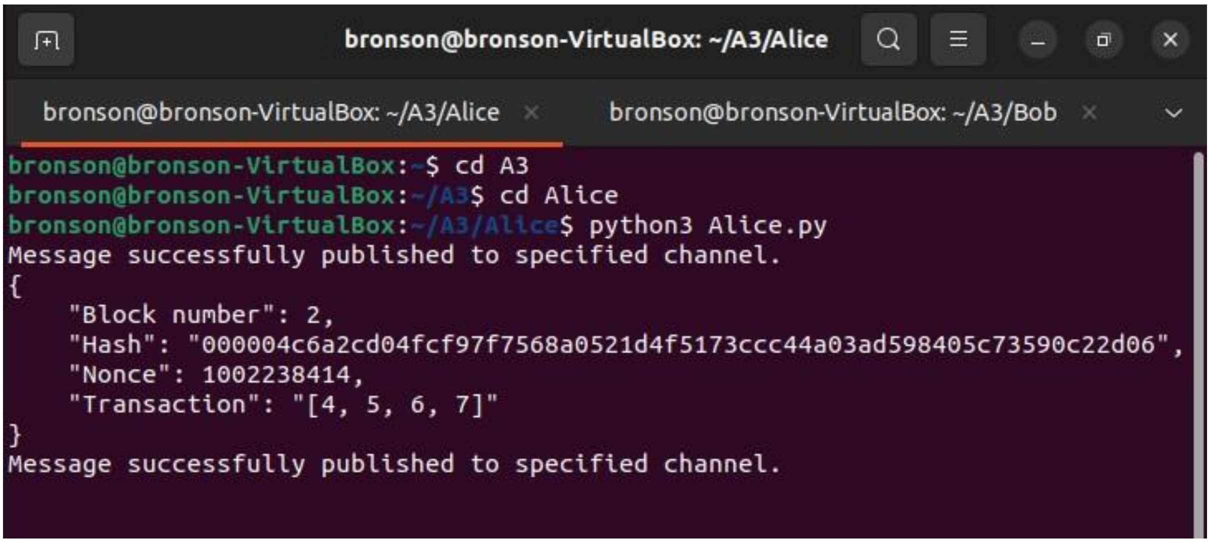
```
bronson@bronson-VirtualBox:~$ cd A3
bronson@bronson-VirtualBox:~/A3$ cd Bob
bronson@bronson-VirtualBox:~/A3/Bob$ python3 Bob.py
{
    "Block number": 1,
    "Hash": "32928f7a5a5ccd48fdb3f7aed9d253dfe3fb06d4854203a4fa934d5c1707dca5",
    "Nonce": 42015,
    "Transaction": "[3, 4, 5, 6]"
}
Message successfully published to specified channel.
```

7. Alice will then save it as 2.json in her own folder too and then verify it before starting mining block 3.


bronson@bronson-VirtualBox: ~/A3/Alice
bronson@bronson-VirtualBox: ~/A3/Alice
bronson@bronson-VirtualBox: ~/A3/Bob

```
bronson@bronson-VirtualBox:~$ cd A3
bronson@bronson-VirtualBox:~/A3$ cd Alice
bronson@bronson-VirtualBox:~/A3/Alice$ python3 Alice.py
Message successfully published to specified channel.
{
    "Block number": 2,
    "Hash": "000004c6a2cd04fcf97f7568a0521d4f5173ccc44a03ad598405c73590c22d06",
    "Nonce": 1002238414,
    "Transaction": "[4, 5, 6, 7]"
}
```

8. Alice finish mining block 3 and sending over.

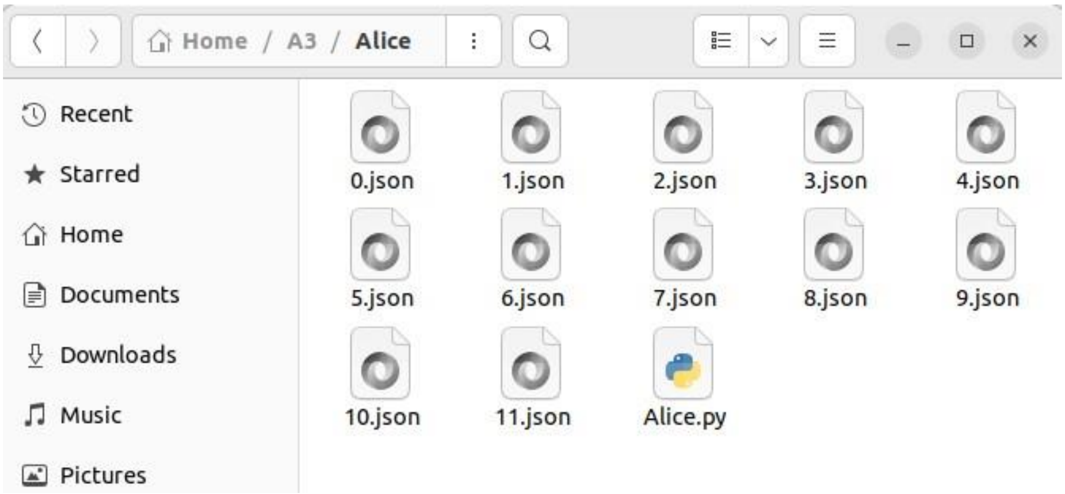

bronson@bronson-VirtualBox: ~/A3/Alice
bronson@bronson-VirtualBox: ~/A3/Alice
bronson@bronson-VirtualBox: ~/A3/Bob

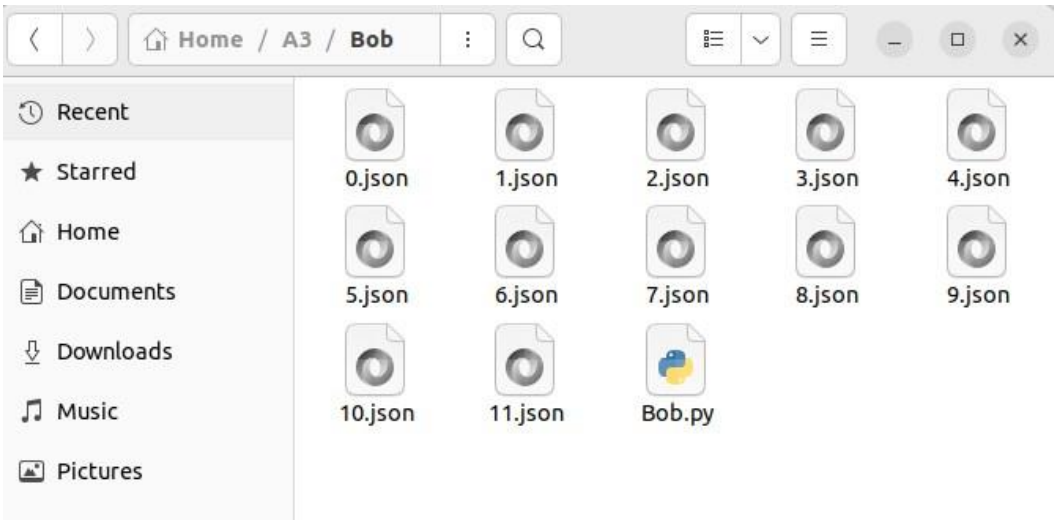
```
bronson@bronson-VirtualBox:~$ cd A3
bronson@bronson-VirtualBox:~/A3$ cd Alice
bronson@bronson-VirtualBox:~/A3/Alice$ python3 Alice.py
Message successfully published to specified channel.
{
    "Block number": 2,
    "Hash": "000004c6a2cd04fcf97f7568a0521d4f5173ccc44a03ad598405c73590c22d06",
    "Nonce": 1002238414,
    "Transaction": "[4, 5, 6, 7]"
}
Message successfully published to specified channel.
```

Result

Both programs need to be stopped manually at the end after block 11 using ctrl + C.

In the end, both Alice's and Bob's folder will both have 0.json to 11.json and their respective python program as shown below. All .json files will be identical in both Alice and Bob's folder.

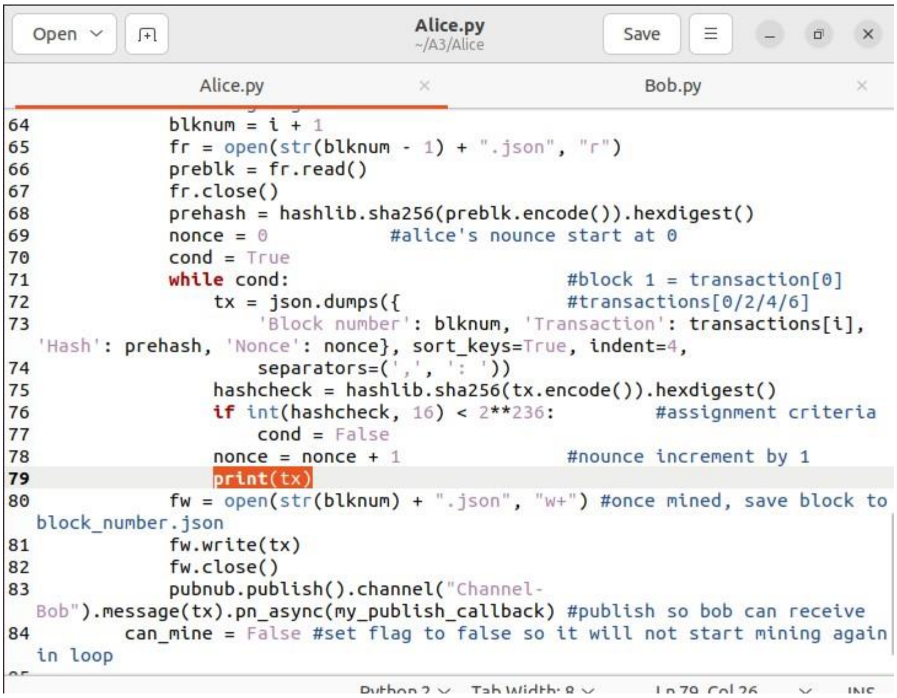




Running with printed block to show the process

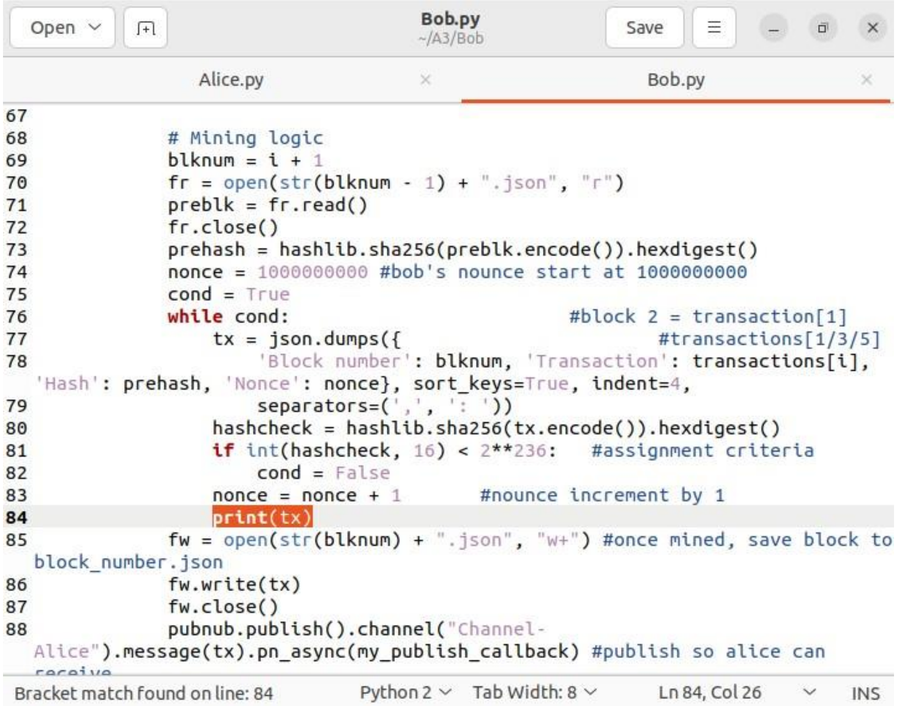
To see the process of mining of blocks with the increment of nonce, we would need to un-comment away “print(tx)” so the program would show the mining.

For Alice.py, it will be at Line 79



```
64          blknum = i + 1
65          fr = open(str(blknum - 1) + ".json", "r")
66          preblk = fr.read()
67          fr.close()
68          prehash = hashlib.sha256(preblk.encode()).hexdigest()
69          nonce = 0           #alice's nounce start at 0
70          cond = True
71          while cond:                             #block 1 = transaction[0]
72              tx = json.dumps({                   #transactions[0/2/4/6]
73                  'Block number': blknum, 'Transaction': transactions[i], 'Hash': prehash, 'Nonce': nonce}, sort_keys=True, indent=4,
74                  separators=(',', ': '))
75              hashcheck = hashlib.sha256(tx.encode()).hexdigest()
76              if int(hashcheck, 16) < 2**236:         #assignment criteria
77                  cond = False
78              nonce = nonce + 1                   #nounce increment by 1
79              print(tx)
80          fw = open(str(blknum) + ".json", "w+") #once mined, save block to block_number.json
81          fw.write(tx)
82          fw.close()
83          pubnub.publish().channel("Channel-Bob").message(tx).pn_async(my_publish_callback) #publish so bob can receive
84      can_mine = False #set flag to false so it will not start mining again in loop
```

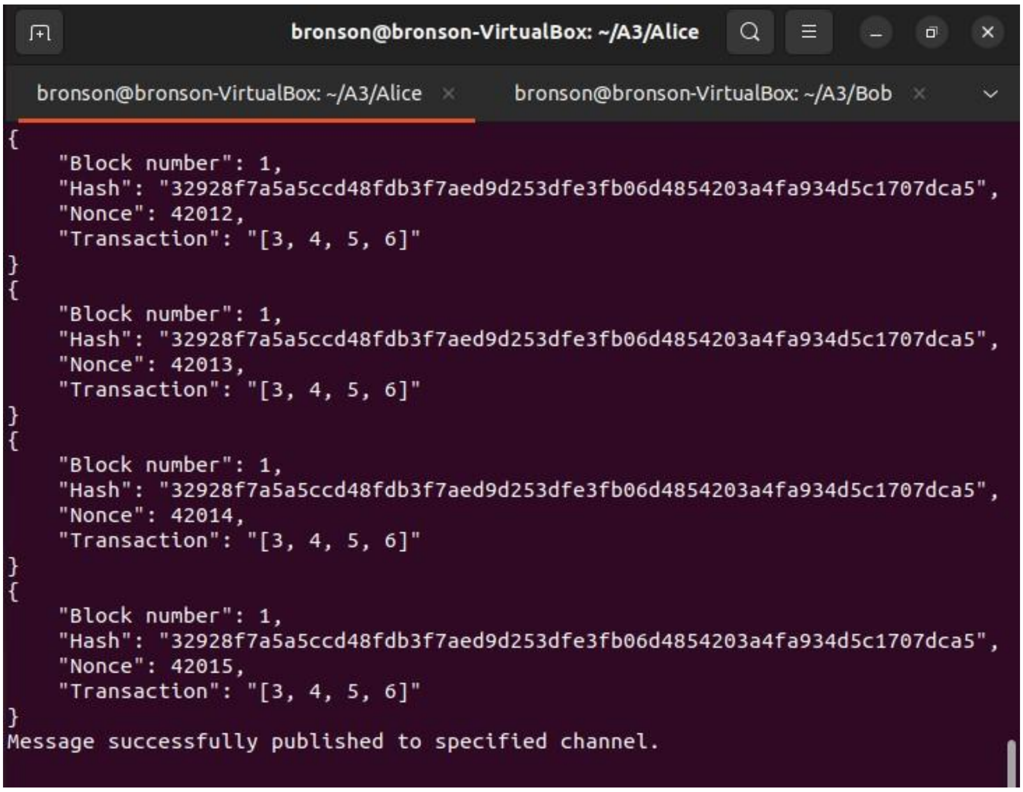
For Bob.py, it will be at Line 84



```
67
68          # Mining logic
69          blknum = i + 1
70          fr = open(str(blknum - 1) + ".json", "r")
71          preblk = fr.read()
72          fr.close()
73          prehash = hashlib.sha256(preblk.encode()).hexdigest()
74          nonce = 1000000000 #bob's nounce start at 1000000000
75          cond = True
76          while cond:                             #block 2 = transaction[1]
77              tx = json.dumps({                           #transactions[1/3/5]
78                  'Block number': blknum, 'Transaction': transactions[i], 'Hash': prehash, 'Nonce': nonce}, sort_keys=True, indent=4,
79                  separators=(',', ': '))
80              hashcheck = hashlib.sha256(tx.encode()).hexdigest()
81              if int(hashcheck, 16) < 2**236:   #assignment criteria
82                  cond = False
83              nonce = nonce + 1       #nounce increment by 1
84              print(tx)
85          fw = open(str(blknum) + ".json", "w+") #once mined, save block to block_number.json
86          fw.write(tx)
87          fw.close()
88          pubnub.publish().channel("Channel-Alice").message(tx).pn_async(my_publish_callback) #publish so alice can receive
```

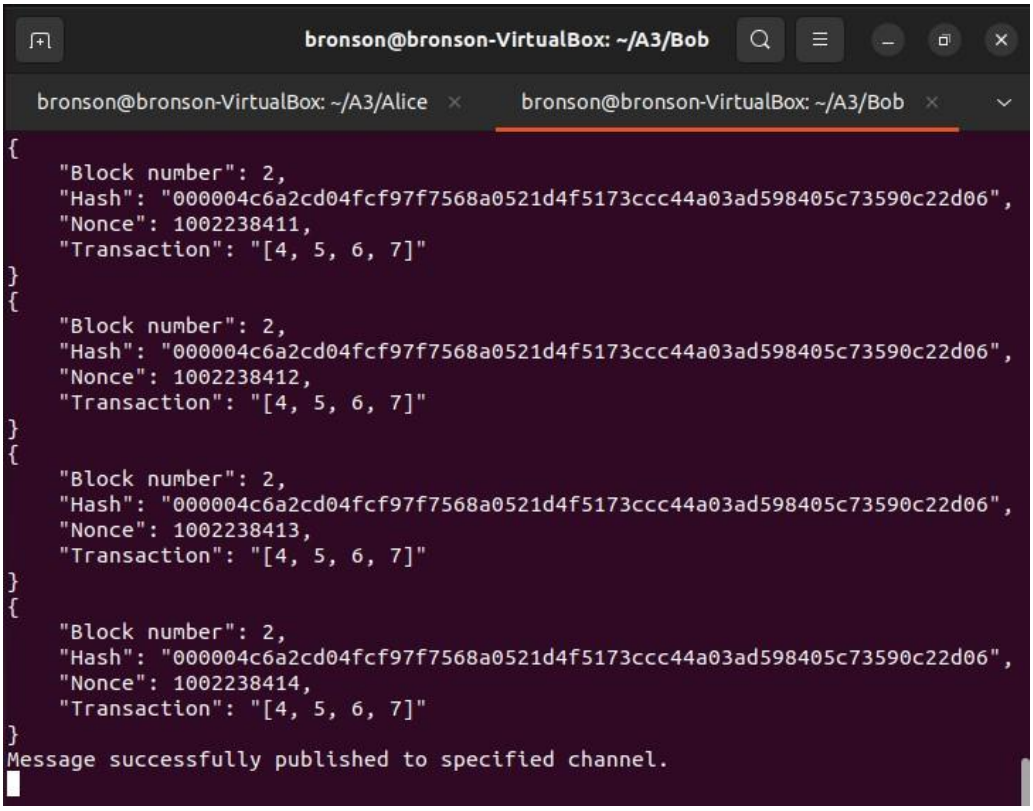
This will be the process of the mining after un-commenting the print(tx)

On Alice’s side:


bronson@bronson-VirtualBox: ~/A3/Alice
bronson@bronson-VirtualBox: ~/A3/Alice
bronson@bronson-VirtualBox: ~/A3/Bob

```
{
    "Block number": 1,
    "Hash": "32928f7a5a5ccd48fdb3f7aed9d253dfe3fb06d4854203a4fa934d5c1707dca5",
    "Nonce": 42012,
    "Transaction": "[3, 4, 5, 6]"
}
{
    "Block number": 1,
    "Hash": "32928f7a5a5ccd48fdb3f7aed9d253dfe3fb06d4854203a4fa934d5c1707dca5",
    "Nonce": 42013,
    "Transaction": "[3, 4, 5, 6]"
}
{
    "Block number": 1,
    "Hash": "32928f7a5a5ccd48fdb3f7aed9d253dfe3fb06d4854203a4fa934d5c1707dca5",
    "Nonce": 42014,
    "Transaction": "[3, 4, 5, 6]"
}
{
    "Block number": 1,
    "Hash": "32928f7a5a5ccd48fdb3f7aed9d253dfe3fb06d4854203a4fa934d5c1707dca5",
    "Nonce": 42015,
    "Transaction": "[3, 4, 5, 6]"
}
Message successfully published to specified channel.
```

On Bob’s side:


bronson@bronson-VirtualBox: ~/A3/Bob
bronson@bronson-VirtualBox: ~/A3/Alice
bronson@bronson-VirtualBox: ~/A3/Bob

```
{
    "Block number": 2,
    "Hash": "000004c6a2cd04fcf97f7568a0521d4f5173ccc44a03ad598405c73590c22d06",
    "Nonce": 1002238411,
    "Transaction": "[4, 5, 6, 7]"
}
{
    "Block number": 2,
    "Hash": "000004c6a2cd04fcf97f7568a0521d4f5173ccc44a03ad598405c73590c22d06",
    "Nonce": 1002238412,
    "Transaction": "[4, 5, 6, 7]"
}
{
    "Block number": 2,
    "Hash": "000004c6a2cd04fcf97f7568a0521d4f5173ccc44a03ad598405c73590c22d06",
    "Nonce": 1002238413,
    "Transaction": "[4, 5, 6, 7]"
}
{
    "Block number": 2,
    "Hash": "000004c6a2cd04fcf97f7568a0521d4f5173ccc44a03ad598405c73590c22d06",
    "Nonce": 1002238414,
    "Transaction": "[4, 5, 6, 7]"
}
Message successfully published to specified channel.
```