Date: 17th September, 2017

Project Partners:
Harshit Shah [UFID: 1211-6976, Email ID: hbshah@ufl.edu]
Pradeep Rajan [UFID: 1995-3544, Email ID: pradeepnr@ufl.edu]

Project Implementation:

- *Local:*
  ./project1 k -> where k is the no. of leading zeroes required in the bitcoin
- Remote:
  ./project1 [server_ip_addr] -> specify the server ip address to connect

Project Details:

1-> Size of Work Unit:

100,000 Strings per worker. We are giving a range of values and base64 of each value is calculated to which UFID is prefixed to it. Finally, SHA256 hash is calculated from it.
Reason: We selected these values as we wanted to keep the actors busy for not too long or too less and found this to be appropriate.

We are maintaining a bucket at the client side which will accumulate all the bitcoins mined. Once the bucket is filled, all the values will be sent at once. By doing this, we will ensure that the server's receive thread is not getting overwhelmed by all the messages sent by actors on the client side.

2-> Result of running the program:

./project1 4

Server started at IP -> bitcoin_server@128.227.248.174
hbshahMTUxNDg2
000033E4A0C622A3DC87DDB689BE14B9B434B832AC9777B5D04FC8E1C15DFBB0
hbshahMTkyMTI2
000024095C76ED3B25666423171A03B3C90F7B08BD3DBAD114C011697637F438

3-> Runtime of the program:

Server started at IP -> bitcoin_server@128.227.248.174

real 0m30.005s
user 3m51.716s
sys 0m0.840s

Ratio: 7.7 on a machine with 8 cores

4-> The coin with the most 0s

hbshahMzE2MDY3MzMw
00000000D78C72D31A3733A2D9B50960600286AD782EC512D36E17CFECF1693E

Number of zeroes: 8

5-> The largest number of working machines the code was able to run with was 5 machines (4 miners and 1 master)