

Κατανεμημένα Συστήματα

9ο εξάμηνο, Ακαδημαϊκή περίοδος 2022-2023

Μπάτσαρη Βασιλική - el18046

Μαρρές Χρήστος - el18912

Εξαμηνιαία Εργασία:

Noobcash

Noobcash-σύστημα.

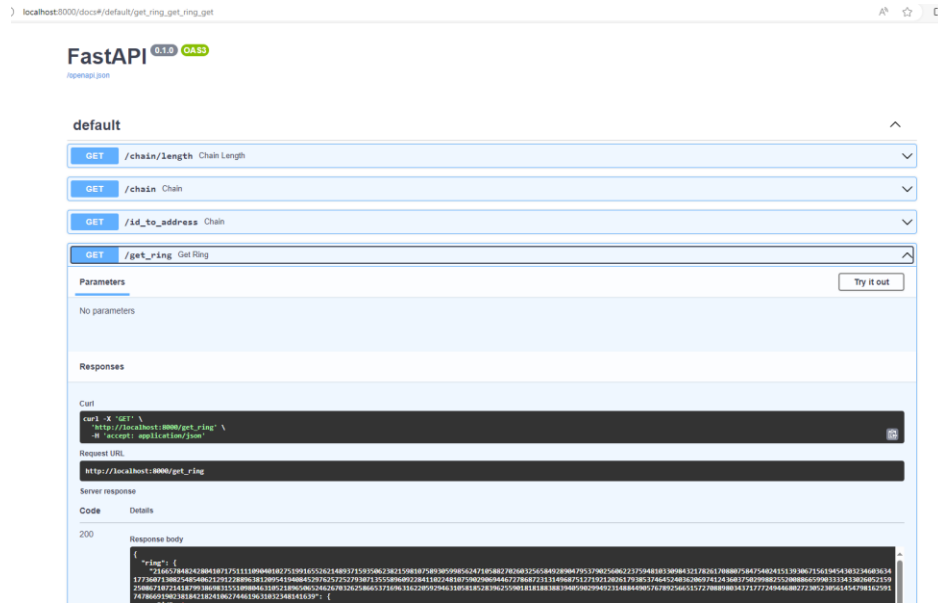
Το noobcash είναι ένα πλήρως αποκεντρωμένο, ψηφιακό σύστημα κρυπτονομισμάτων. Θεωρούμε ότι όλοι οι κόμβοι είναι miners και δεν αμείβονται για την συνδρομή τους στην εύρεση του κατάλληλου nonce. Κάθε συναλλαγή μεταξύ δύο χρηστών/κόμβων, αποστέλλεται στο δίκτυο, ώστε να γίνει γνωστή και τους υπόλοιπους κόμβους. Κάθε κόμβος, μόλις λάβει την συναλλαγή, την επαληθευτεί και την προσθέτει στο block του. Μόλις το block κάποιου κόμβου γεμίσει, ο κόμβος αυτός το αποστέλλει σε όλους τους υπόλοιπους κόμβους του συστήματος. Όταν ένα block γίνει αποδεκτό από τους υπόλοιπους κόμβους, προστίθεται στο blockchain.

Η γλώσσα προγραμματισμού που χρησιμοποιείται στην υλοποίηση του συστήματος είναι η python. Οι python κλάσεις που χρησιμοποιούνται στο πρόγραμμά μας είναι οι εξής: block, blockchain, node, transaction, transaction_Output και wallet.

Έχει δημιουργηθεί ένα cli για κάθε κόμβο, το οποίο μπορεί να εκτελέσει τις εντολές help, balance, view και t [recipient_address] [amount]. Παρατίθεται ένα παράδειγμα χρήσης του cli στον κόμβο client1, από τον οποίο ζητείται αρχικά να εκτυπώσει το υπόλοιπό του (το πόσα noobcash coins έχει στην κατοχή του την δεδομένη χρονική στιγμή), έπειτα, στέλνει στον client2 (του οποίου γνωρίζει το public key) δέκα (10) coins, και τέλος, επανεκτυπώνει το υπόλοιπό του.

Παρατήρηση: Το public key κάθε κόμβου μπορεί να βρεθεί από το get_ring στο url: <http://localhost:8000/docs#/>

```
root@d4cd49a98f1c:/source/cli# python main.py balance
71
root@d4cd49a98f1c:/source/cli# python main.py t 22507698235145027269507056312876992258203037517089601308110889594119667550206376495230519043475838628921604805677
76044899812236836822328819367899190892578383056496171252270102421937714617602923130213478656106907469655894363091179748890249351336262822569829336070132686108127
0531161673067544557731395055877703418838729151715750004403448569449352149766975654881238187325886687722381456018196543420640248716623009894450888091522932857975754
46354489415467489461294965809349971037731363291402130907109177100772725076064035077386228477161471116943041057683536819411309904312060349882172849944942738978164
53373091208512371 10
Transaction created successfully!
root@d4cd49a98f1c:/source/cli# python main.py balance
61
```



Απόδοση Συστήματος.

Η απόδοση του συστήματος έχει μετρηθεί για κάθε πιθανό συνδυασμό των δύο παραμέτρων capacity και difficulty, όπου οι τιμές για το capacity (δηλαδή το πόσες συναλλαγές χωράει κάθε block) επιλέγονται από το set: {1, 5, 10} και για το difficulty (το οποίο υποδηλώνει με πόσα μηδενικά -τουλάχιστον- πρέπει να αρχίζει το hash του block) επιλέγονται από το set: {4, 5}.

Οι μετρικές απόδοσης του συστήματος είναι το throughput (ρυθμαπόδοση) του συστήματος, δηλαδή το πόσα transactions εξυπηρετούνται στην μονάδα του χρόνου και το block time, δηλαδή, ο μέσος χρόνος που απαιτείται για να προστεθεί ένα νέο block στο blockchain.

Για 5 κόμβους, με 100 συναλλαγές ο κάθε κόμβος, τα αποτελέσματα είναι τα εξής:

```
vbatari@pcep850594 MINGW64 ~/Desktop/distributed_systems/NTUA-Distributed-Systems-Project (main)
$ python results.py

#-#-#-# FILE: p5-d4-c1.txt #-#-#-#

Throughput of the system: 0.15894015578116263
Average block time: 1.7320635300081186

#-#-#-# FILE: p5-d4-c10.txt #-#-#-#

Throughput of the system: 3.430502391723948
Average block time: 2.7912658330940068

#-#-#-# FILE: p5-d4-c5.txt #-#-#-#

Throughput of the system: 1.3902643902656862
Average block time: 1.275965280424828

#-#-#-# FILE: p5-d5-c1.txt #-#-#-#

Throughput of the system: 0.10502629272409461
Average block time: 2.1529532075327613

#-#-#-# FILE: p5-d5-c10.txt #-#-#-#

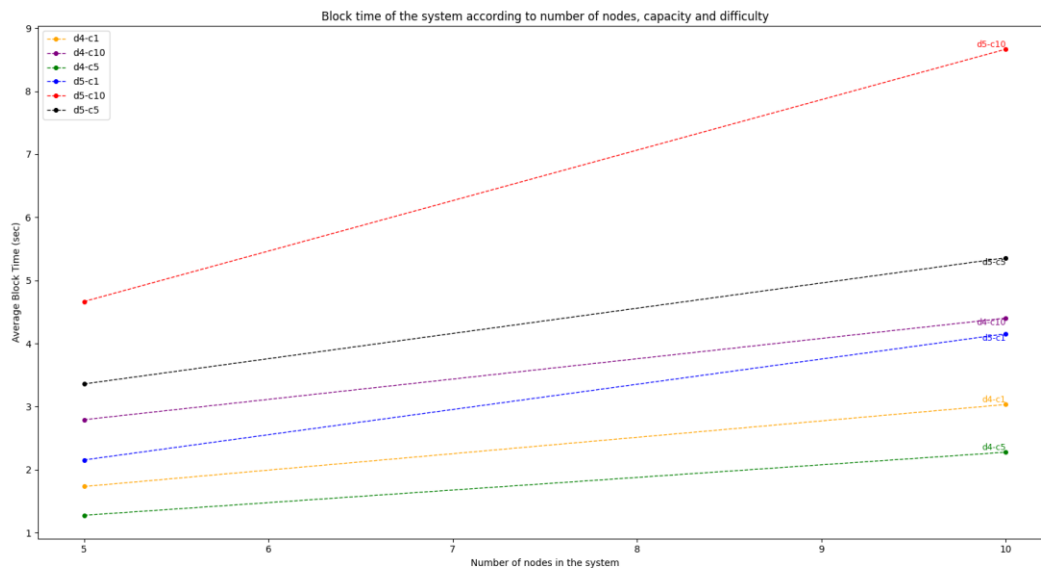
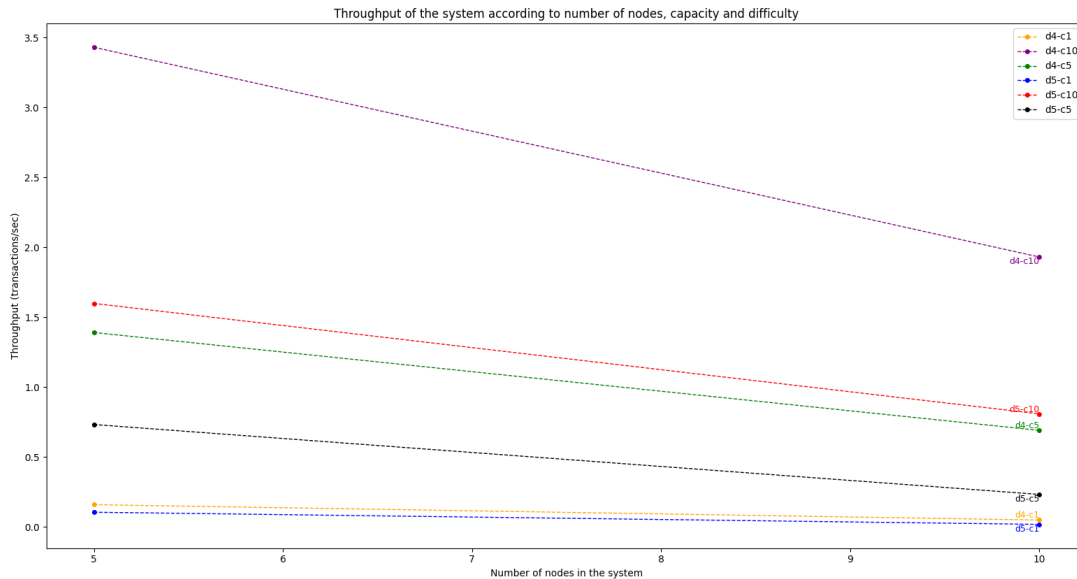
Throughput of the system: 1.5983972285353045
Average block time: 4.6646684945784305

#-#-#-# FILE: p5-d5-c5.txt #-#-#-#

Throughput of the system: 0.7322081821666063
Average block time: 3.358282789706025
```

Κλιμακωσιμότητα Συστήματος.

Για την αξιολόγηση της κλιμακωσιμότητας του συστήματος, επαναλάβαμε το πείραμα με 10 κόμβους, αντί για 5, πάλι με 100 συναλλαγές ο καθένας. Παρακάτω παρατίθεται η σύγκριση των αποτελεσμάτων, μέσω δύο γραφημάτων, ένα για το throughput και ένα για το block time.



Περιβάλλον ανάπτυξης - Docker.

Για την εξαμηνιαία εργασία χρησιμοποιήθηκε Docker. Οι λόγοι της επιλογής αυτής είναι οι εξής: πρωτίστως, με χρήση docker, όλο το setup μπορεί να επαναληφθεί εύκολα και γρήγορα, με μία απλή εντολή, σε οποιοδήποτε μηχάνημα, αρκεί το τελευταίο να έχει κατεβασμένο το docker app και δευτερευόντως, δεν απαιτούνται ssh sessions με κάποιο απομακρυσμένο Virtual Machine (VM), ούτε σύνδεση στο internet, για την εκτέλεση του προγράμματος.

Παρατήρηση: Το γεγονός ότι στην περίπτωση των 10 κόμβων, το throughput μειώνεται και το block time αυξάνεται, σε σχέση με την περίπτωση των 5 κόμβων -διατηρώντας τις υπόλοιπες παραμέτρους σταθερές- οφείλεται στην χρήση docker και όχι αληθινών VMs. Αν το παραπάνω έτρεχε σε 5 ή 10 διαφορετικά VMs, τότε αναμενόμενο αποτέλεσμα θα ήταν να αποδειχθεί αποδοτικότερο το σύστημα με 10 κόμβους, έναντι 5.

GitHub Repository Link.

Το link για το github repo της εργασίας, με όλους τους κώδικες, καθώς και οδηγίες για αναπαραγωγή των αποτελεσμάτων, είναι το εξής: <https://github.com/chris-marres/NTUA-Distributed-Systems-Project> .