# Bitcoin algoritam

**Radošević, Stefan**

**Master's thesis / Diplomski rad**

**2018**

*Degree Grantor / Ustanova koja je dodijelila akademski / stručni stupanj:* **Josip Juraj Strossmayer University of Osijek, Faculty of Electrical Engineering, Computer Science and Information Technology Osijek / Sveučilište Josipa Jurja Strossmayera u Osijeku, Fakultet elektrotehnike, računarstva i informacijskih tehnologija Osijek**

*Permanent link / Trajna poveznica:* https://urn.nsk.hr/urn:nbn:hr:200:209511

*Rights / Prava:* In copyright / Zaštićeno autorskim pravom.

*Download date / Datum preuzimanja:* **2025-03-21**

*Repository / Repozitorij:*

Faculty of Electrical Engineering, Computer Science and Information Technology Osijek

**SVEUČILIŠTE JOSIPA JURJA STROSSMAYERA U OSIJEKU**
**FAKULTET ELEKTROTEHNIKE,RAČUNARSTVA I INFORMACIJSKIH TEHNOLOGIJA**

**Sveučilišni studij**

# BITCOIN ALGORITAM

**Diplomski rad**

**Stefan Radošević**

**Osijek, 2018.**

FERIT
FAKULTET ELEKTROTEHNIKE, RAČUNARSTVA I INFORMACIJSKIH TEHNOLOGIJA OSIJEK

**Obrazac D1: Obrazac za imenovanje Povjerenstva za obranu diplomskog rada**

**Osijek, 12.07.2018.**

**Odboru za završne i diplomske ispite**

# Imenovanje Povjerenstva za obranu diplomskog rada

| | |
|---|---|
| **Ime i prezime studenta:** | Stefan Radošević |
| **Studij, smjer:** | Diplomski sveučilišni studij Računarstvo |
| **Mat. br. studenta, godina upisa:** | D 884 R, 27.09.2017. |
| **OIB studenta:** | 48283494050 |
| **Mentor:** | Doc.dr.sc. Ivan Aleksi |
| **Sumentor:** | |
| **Sumentor iz tvrtke:** | |
| **Predsjednik Povjerenstva:** | Doc.dr.sc. Tomislav Matić |
| **Član Povjerenstva:** | Izv. prof. dr. sc. Damir Blažević |
| **Naslov diplomskog rada:** | Bitcoin algoritam |
| **Znanstvena grana rada:** | **Programsko inženjerstvo (zn. polje računarstvo)** |
| **Zadatak diplomskog rada:** | (rezervirao Stefan Radošević) U ovom diplomskom radu je potrebno obraditi Bitcoin algoritam. Potrebno je napisati postupak rračunanja Bitcoina kao i pojmove hash rate, blockchain i kriptovaluta. Potrebno je objasniti kako rade ASIC mineri. U radu je potredno predočiti programsko rješenje i opisati njegove dijelove. |
| **Prijedlog ocjene pismenog dijela ispita (diplomskog rada):** | Izvrstan (5) |
| **Kratko obrazloženje ocjene prema Kriterijima za ocjenjivanje završnih i diplomskih radova:** | Primjena znanja stečenih na fakultetu: 2 bod/boda<br>Postignuti rezultati u odnosu na složenost zadatka: 2 bod/boda<br>Jasnoća pismenog izražavanja: 2 bod/boda<br>Razina samostalnosti: 3 razina |
| **Datum prijedloga ocjene mentora:** | 12.07.2018. |

| | |
|---|---|
| Potpis mentora za predaju konačne verzije rada u Studentsku službu pri završetku studija: | Potpis: |
| | Datum: |

FERIT
FAKULTET ELEKTROTEHNIKE, RAČUNARSTVA I INFORMACIJSKIH TEHNOLOGIJA OSIJEK

# IZJAVA O ORIGINALNOSTI RADA

**Osijek, 30.07.2018.**

| | |
|---|---|
| **Ime i prezime studenta:** | Stefan Radošević |
| **Studij**: | Diplomski sveučilišni studij Računarstvo |
| **Mat. br. studenta, godina upisa:** | D 884 R, 27.09.2017. |
| **Ephorus podudaranje [%]:** | 5 |

Ovom izjavom izjavljujem da je rad pod nazivom**: Bitcoin algoritam**

izrađen pod vodstvom mentora Doc.dr.sc. Ivan Aleksi

i sumentora

moj vlastiti rad i prema mom najboljem znanju ne sadrži prethodno objavljene ili neobjavljene pisane materijale drugih osoba, osim onih koji su izričito priznati navođenjem literature i drugih izvora informacija.
Izjavljujem da je intelektualni sadržaj navedenog rada proizvod mog vlastitog rada, osim u onom dijelu za koji mi je bila potrebna pomoć mentora, sumentora i drugih osoba, a što je izričito navedeno u radu.

Potpis studenta:

## Contents

1. UVOD ..... 1
1.1. Zadatak diplomskog rada ..... 1
2. BITCOIN MREŽA ..... 2
2.1. Bitcoin ..... 2
2.2. Blockchain ..... 3
2.3. Digitalni potpis ..... 8
2.4. Hash rate ..... 9
2.5. ASIC mineri ..... 10
3. BITCOIN ALGORITAM ..... 11
3.1. BLOCK.CPP i BLOCK.H ..... 11
3.2. TRANSACTION.CPP i TRANSACTION.H ..... 12
3.3. CHAIN.CPP i CHAIN.H ..... 15
3.4. HASH.CPP i HASH.H ..... 18
4. ZAKLJUČAK ..... 19
SAŽETAK ..... 20
ABSTRACT ..... 20
ŽIVOTOPIS ..... 21
LITERATURA ..... 22
PRILOG ..... 23

# 1. UVOD

Cilj ovog diplomskog rada je objasniti Bitcoin algoritam. Kako bi Bitcoin algoritam mogao biti pojašnjen, prvo je potrebno objasniti što je to Bitcoin. Stoga je rad podijeljen u dvije glavne cjeline: Bitcoin i Bitcoin algoritam. Kroz rad objasnit će se što je to zapravo bitcoin, kako ga stvoriti i koja je njegova uloga. Ostatak rada je posvećen analizi samog Bitcoin algoritma.

## 1.1. Zadatak diplomskog rada

U ovom diplomskom radu je potrebno obraditi Bitcoin algoritam. Potrebno je napisati postupak računanja Bitcoina kao i pojmove hash rate, blockchain i kripto valuta. Potrebno je objasniti kako rade ASIC mineri. U radu je potrebno predočiti programsko rješenje i opisati njegove dijelove.

# 2. BITCOIN MREŽA

Bitcoin je predstavljen 2008 godine u članku pod imenom „Bitcoin: A Peer-to-Peer electronic Cash System“, a s radom počinje u siječnju 2009. godine. Identitet tvorca bitcoina, koji se spominje u znanstvenom radu u kojem se Bitcoin prvi put pojavljuje, potpisan je sa „Satoshi Nakamoto“, tj. identitet tvorca bitcoina je nepoznat. Namjera je bila ostvariti potpunu kontrolu nad financijama stvaranjem stabilne, sigurne, svjetski prihvatljive i demokratske valute.

Sl. 2.1. Bitcoin

## 2.1.Bitcoin

Bitcoin (BTC) je digitalna, decentralizirana, anonimna platna mreža jer omogućuje plaćanje dobara i usluga te prijenos vrijednosti na daljinu. Transakcije u mreži Bitcoin se bilježe u javnoj knjizi koristeći vlastitu valutu bitcoin. Plaćanje je provođeno na principu peer to peer mreže, bez središnje institucije ili administracije. Jednostavnije rečeno, bitcoin je digitalni, virtualni novac čuvan elektronički. Sve transakcije obavljene pomoću bitcoina su anonimne, sigurne i povjerljive.

Bitcoin je prvi primjer kriptovalute. Kriptovaluta je vrsta digitalnog novca koja funkcionira na temelju kriptografskih algoritama koja ima osobine standardnih valuta, osim što iza njih ne stoji autoritet države i postoje samo u elektronskom obliku. Valuta bitcoin razmjenjuje se među korisnicima Bitcoin mreže primarno putem Interneta. Bitcoin je prenosiv tj. moguće ga je pokretati na raznim računalnim uređajima. Pomoću bitcoina je moguće obavljati gotovo sve što se može obavljati standardnim valutama. Transakcije preko Bitcoin mreže, u usporedbi s bankarskom mrežom, je brža, jeftinija, transparentna i anonimna.

Bitcoin transakcija je zapis u bitcoin mreži kojim se određeni iznos bitcoina prenosi sa jedne ili više adresa na drugu adresu (ili više njih). Sve transakcije u bitcoin mreži su javne, transparentne

i pseudo anonimne. Sve što se može saznati iz njih je to da je određena količina bitcoina prebačena s jedne adrese na drugu.

Bitcoin se najčešće upotrebljava u četiri svrhe: kupovini, slanje novaca, ulaganje, spekulacija. Bitcoin se može koristiti, na Internetu, kao i svaka druga valuta, samo jednostavnije i brže. Slanje novaca putem bankarske mreže traje neko vrijeme i koštaju. Slanje novaca bitcoin mrežom je puno povoljnije. Transakciju je moguće obaviti tako što se standardna valuta promjeni u bitcoin, pošalje primatelju, te primatelj promjeni bitcoin u željenu valutu. Jedini gubitak u toj transakciji je postotak ukupnog iznosa promjene iz prve valute u bitcoin i bitcoin u drugu valutu, ovisno o tome koliko je dobra ponuda za kupovinu i prodaju bitcoina. Prilikom ulaganja, bitcoin je kao i svaka druga valuta, s jednom razlikom, a to je da je vrijednost bitcoina daleko promjenjivija od standardnih valuta. Vrijednost bitcoina se može drastično promijeniti u samo jednom danu, kako je moguć nagli porast, tako je moguć i nagli pad njegove vrijednosti. Zbog toga je riskantno ulagati u Bitcoin. S obzirom da je vrijednost bitcoina nestabilna otvara se mogućnost spekulacije, kupnje bitcoina po što manjoj cijeni, te prodaja istog po što većoj. Međutim, spekulacije je iznimno riskantna i nije preporučljivo se upuštanje u istu.

Nakon porasta popularnosti Bicoin mreže, te porasta cijene samog bitcoina pojavile su se nove mogućnosti za mijenjanje bitcoina u standardne valute i obratno. Danas je moguće Bitcoin zamijeniti u posebnim mjenjačnicama za ostale standardne valute. Pored mjenjačnica, postoje i Bitcoin bankomati. To su internet mašine koje pružaju mogućnost korisnicima mijenjanje bitcoina za novac. Neki od Bitcoin bankomata pružaju mogućnost kupnje bitcoina.

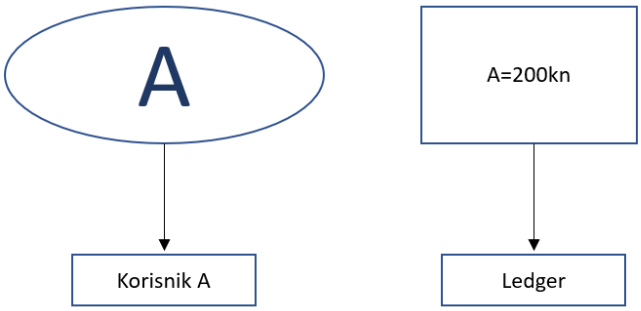
Osim pojma kripto valute, prije samog prelaska na analiziranje bitcoin algoritma, potrebno je pojasniti što je to blockchain, digitalni potpis i hash rate.

## 2.2.Blockchain

Blockchain je tehnologija, koju je izmislio tvorac Bitcoin mreže, koja omogućuje izvršavanje transakcije digitalne valute (u ovom radu bitcoina) ili dobara između dva pojedinca. Problem koji blockchain pokušava riješiti je prijenos novca. Glavna ideja je izvršiti prijenos novca s jedne lokacije na drugu bržim i jeftinijim putem, bez prisutnosti treće strane, odnosno posrednika. Prvi koncept koji je potrebno objasniti da bi blockchain bio razumljiv je Open Ledger. Open Ledger je, u principu, „lanac“ transakcija koji je izvršen unutra neke mreže, u ovom slučaju Bitcoin mreže, te su ti podaci javni i dostupni svima. Unutar „lanca“ su prikazane sve transakcije koje su se izvršile između svih korisnika mreže. Svaka legitimna transakciju upisana je u „glavnu knjigu“, odnosno Ledger, gdje se nalaze sve ranije izvršene transakcije. Ukoliko transakcija nije

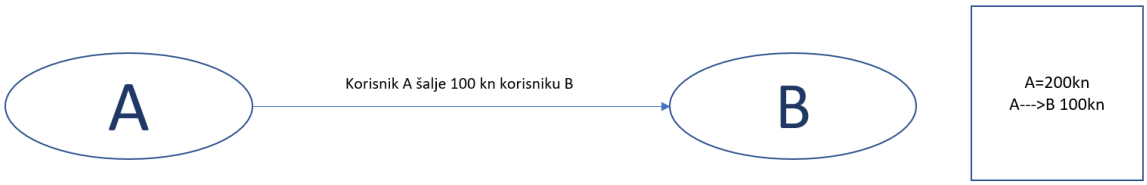
legitimna, npr. korisnik je pokušao poslati veću vrijednost dobara nego što posjeduje, transakcija se neće izvršiti, te se neće upisati u Ledger. Za lakše razumijevanje koncepta Open Ledger prikazan je sljedeći primjer:

Neka imovina korisnika A iznosi 200 kn.



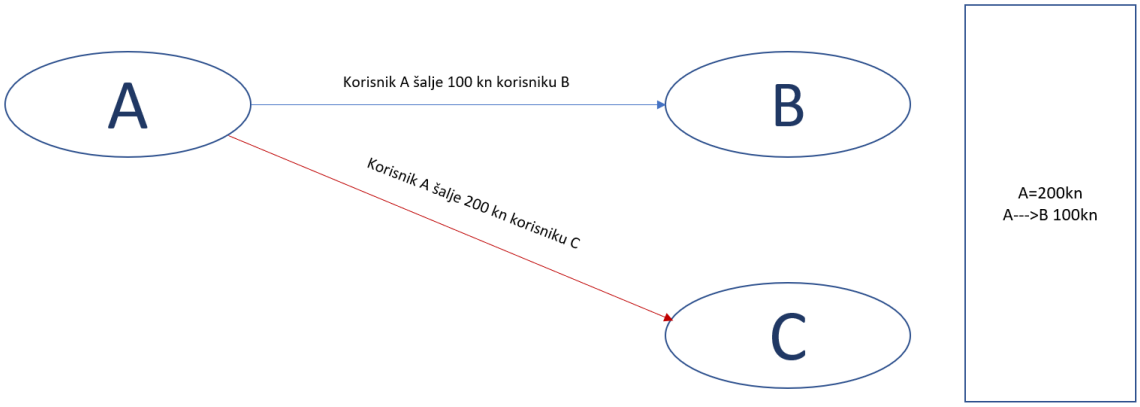

Sl. 2.2. Prikaz korisnik A, A=200kn je upisano u Ledger

Korisnik A je odlučio poslati 100 kn udaljenom korisniku B.




Sl. 2.3. Izvršena legitimna transakcija između korisnika A i B u iznosu od 100 kn

Nakon izvršene transakcije u Ledger se upisuje A--->B 100kn što znači da je Korisnik A uspješno poslao Korisniku B imovinu u vrijednosti od 100 kn. Nakon prikazane transakcije na slici 2.3. Korisnik A odluči poslati drugom korisniku. npr. korisniku C imovinu u vrijednosti od 200 kn.

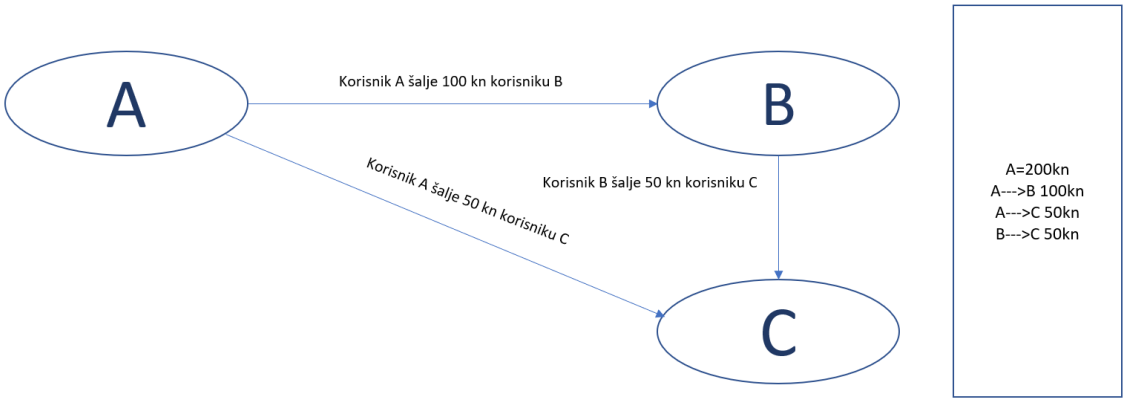



Sl. 2.4. Nelegitimna transakcija između korisnika A i korisnika C

Budžet Korisnika A, nakon prve transakcije, iznosi 100kn. To znači da je nemoguće izvršiti transakciju prema korisniku C. Takva transakcije se registrira kao nelegitimna, te se takva

transakcije ne upisuje u Ledger i ne izvršava se. Isti korisnik ne može dva puta potrošiti isti novac i time su korisnici osigurani od krivotvorenja novca.

Korisnik A i korisnik B odlučili su poslati korisniku C imovinu u vrijednosti od 50 kn svaki.



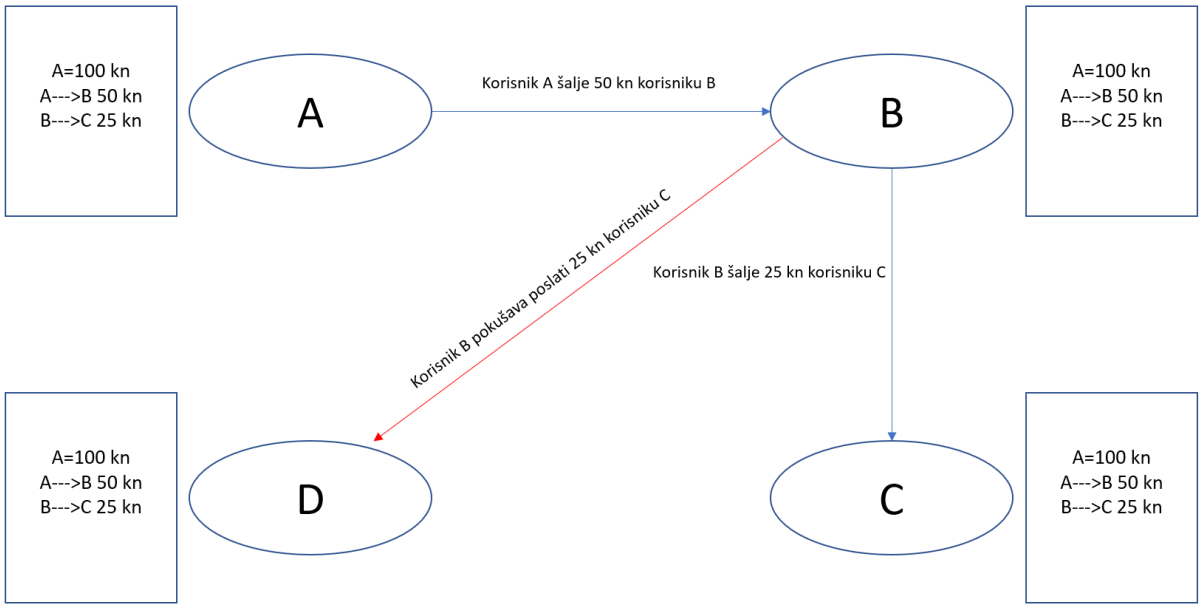

Sl. 2.5. Transakcija u vrijednosti od 50 kn između korisnika A i B s korisnikom C je uspješno izvršena

Bez obzira što je probana nelegitimna transakcija, mreža nastavlja funkcionirati nesmetano. Svaka 1 kn, u ovom primjeru, ima svoj univerzalan ID, te se pomoću tog ID prati njen položaj u mreži. Open Ledger omogućava svim korisnicima uvit u sve transakcije koje su se dogodile u mreži, te svaki korisnik može odlučiti da li je neka transakcije legitimna ili ne. Na primjeru sa slike 2.4. probana je nelegitimna transakcija. U momentu kada je korisnik pokušao izvršiti takvu transakciju, svi unutar mreže imaju uvid u to. Isti je slučaj i kada je transakcije legitimna. Jedina razlika između te dvije transakcije je treba li ju upisati u Ledger ili ne.

Na primjerima prikazanim ranije u tekstu vidljivo je da postoji centralizirana jedinica koja nadgleda sve transakcije, Ledger. Rečeno je da je Bitcoin mreža necentralizirana, stoga je cilj blockchaina skloniti treću stranu kako ne bi bilo posrednika pri izvršavanju transakcije. To dovodi do drugog koncepta blockchaina, „distributed“ Ledger. Kako bi se otklonio posrednik, blockchain uzima centralizirani Ledger i isporučuje ga svim korisnicima mreže. Sada svaki korisnik ima svoju kopiju Ledgera. Ledger je distribuiran i više nije potreban centraliziran Ledger.

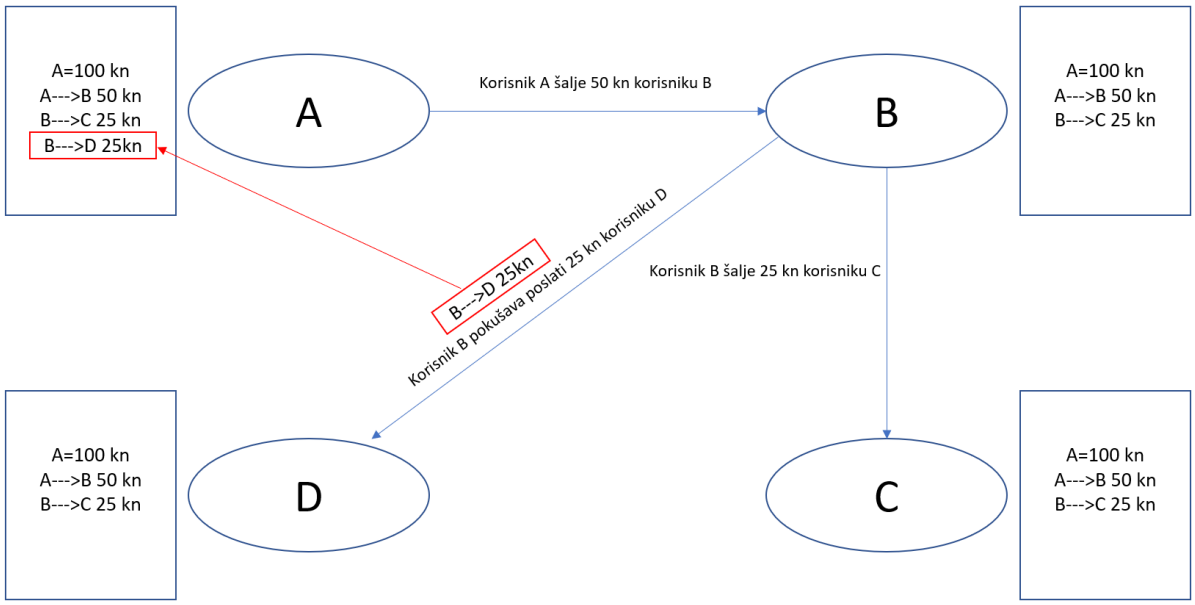
To dovodi do novog problema. S obzirom da svaki korisnik ima svoju kopiju Ledgera potrebno ih je sinkronizirati, tako da svaki korisnik vidi istu verziju Ledgera. Navedeni problem je rešen trećim principom blockchain algoritma koji je objašnjen pomoću sljedećeg primjera.

Budžet korisnika A iznosi 100 kn i napravljene su dvije transakcije: korisnika A prema korisniku B u vrijednosti od 50 kn i transakcija korisnika B prema korisniku C u vrijednosti od 25 kn.



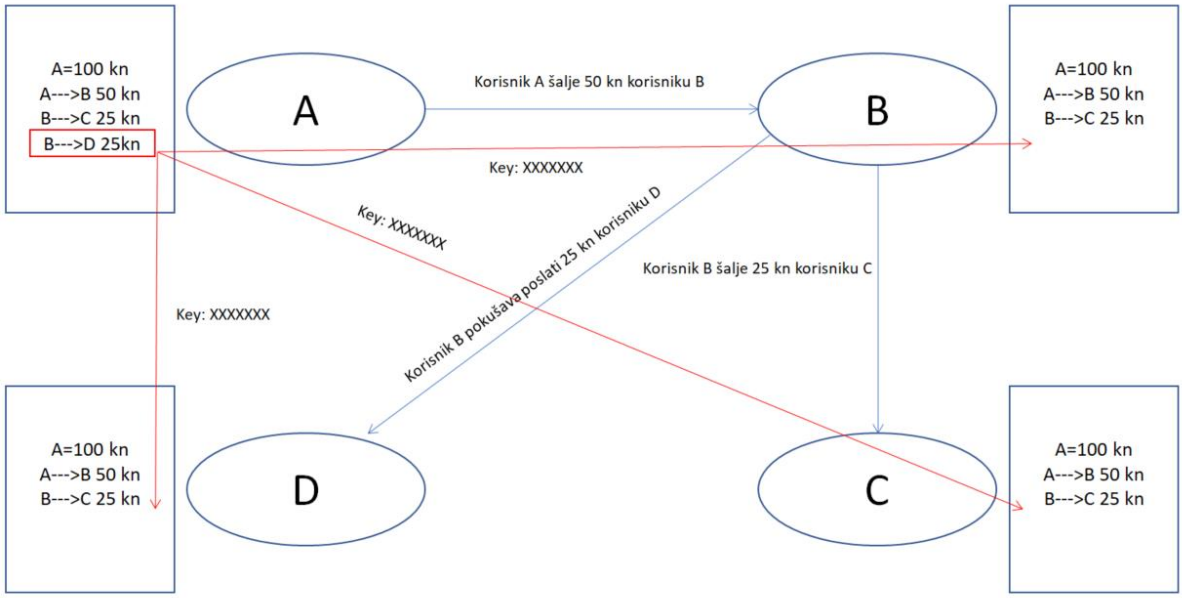

Sl. 2.6. Korisnik B pokušava poslati korisniku D 25 kn

U primjeru kao što je prikazan na slici 2.6. korisnik B će napraviti broadcast ostalim korisnicima u Bitcoin mreži o njegovoj željenoj transakciji. Svaki korisnik će vidjeti da korisnik B želi izvršiti pomenutu transakciju s korisnikom D. Navedena transakcija je još uvijek smatrana nelegitimnom što znači da još uvijek neće biti upisana u Ledger. Da bi bila upisana u Ledger potrebni su mineri. Mineri su specijalni node-ovi u kojima se, u ovom slučaju, nalazi Ledger. Mineri se međusobno natječu tko će biti nelegitimnu transakciju pretvoriti u legitimnu i upisati je u svoj Ledger. Prvi miner koji to napravi dobija financijsku nagradu, u ovom slučaju bitcoin. Da bi miner upisao transakciju u svoj Ledger, mora učiniti dvije stvari. Prvo mora legitimirati novu transakciju. S obzirom da je Ledger dostupan svima i svako ima isti Ledger, može se izračunati da li korisnik B ima dovoljno sredstava da ostvari željenu transakciju prema korisniku D. Druga stvar koju miner mora učinit je pronaći specijalni ključ koji će omogućiti povezivanje posljednje ostvarene transakcije u „lanac“ i trenutne željene transakcije. Da bi miner pronašao potreban specijalni ključ mora uložiti svoju računalnu snagu i vrijeme jer je pretraživanje za ključem nasumično, što bi značilo da miner uzastopno pogađa novi ključ sve dok ne pronađe pravi ključ. Prvi miner koji pronađe taj ključ dobit će financijsku nagradu. Neka je miner korisnika A prvi uspješno pronašao traženi ključ i upisao transakciju korisnika B prema korisniku D u Ledger.



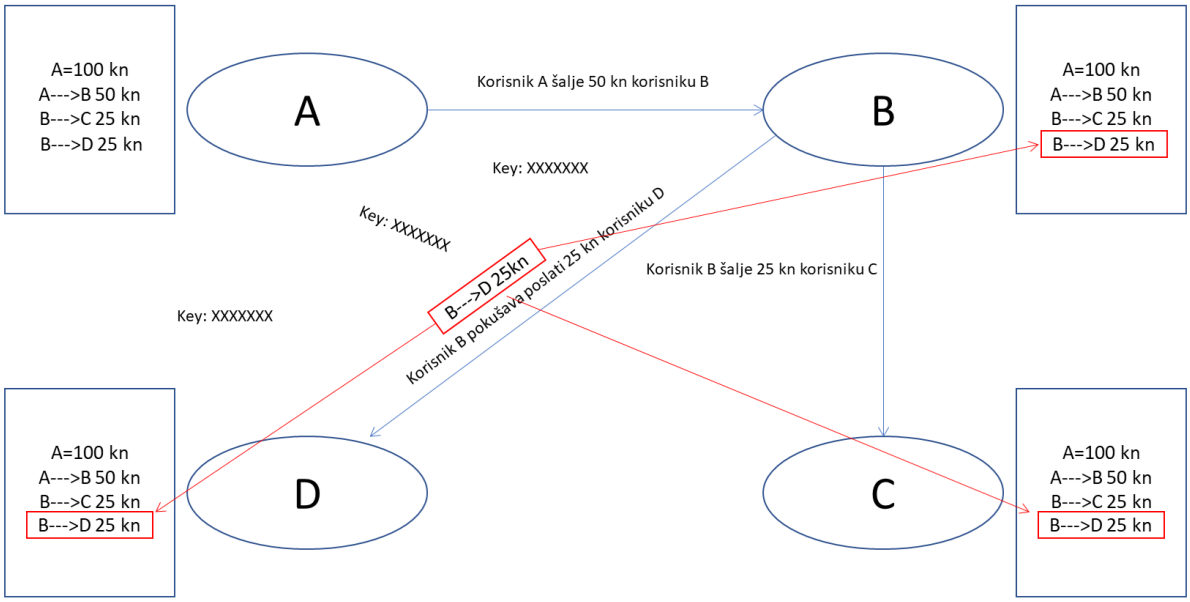

Sl. 2.7. Miner korisnika A uspješno pronalazi specijalni ključ koji je potreban da upiše navedenu transakciju u svoj Ledger

Miner korisnika A nakon pronalaska specijalnog ključa i upisivanja transakcije u svoj Ledger šalje broadcast cijeloj mreži rješenje koje je pronašao tj. vrijednost specijalnog ključa za povezivanje posljednje transakcije i trenutne u Ledger.




Sl. 2.8. Miner korisnika A šalje vrijednost specijalnog ključa ostatku Bitcoin mreže

Ostali mineri će stati s pretraživanjem ključa i s primljenim ključem upisati transakciju u svoj Ledger jer je nepotrebno dalje pretraživati ključ iz razloga što samo miner koji je prvi pronašao ključ prima financijsku nagradu.



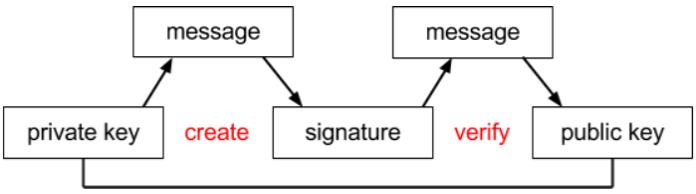

Sl. 2.9. Preostali korisnici Bitcoin mreže upisuju transakciju u svoj Ledger nakon što su dobili vrijednost specijalnog ključa

Nakon što je cijeli proces gotov, svi mineri traže novu transakciju za koju je potrebno naći specijalni ključ te je upisati u Ledger kako bi dobili financijsku nagradu.

## 2.3.Digitalni potpis

Svaka transakcija u blockchainu, svaki blok, sadrži jedinstveni digitalni potpis. Digitalni potpis, jednostavnije rečeno, je jedinstvena lozinka koja otključava bitcoin wallet i omogućava izvršenje transakcija. Svaka transakcija ima potpuno drugačiji digitalni potpis. Digitalni potpis je dokaz da je poruka autentična. S obzirom da svaka transakcija ima drugačiji digitalni potpis koji je jedinstven isključivo za tu transakciju, otklonjena je mogućnost kopiranja lozinke potrebne za otključavanje bitcoin wallet. Digitalni potpis je u principu jedna od zaštitnih mjera Bitcoin mreže koja sprečava pristup tuđem „budžetu“.

Digitalni potpis koristi dva ključa, privatni i javni. Privatni se koristi za kreiranje potpisa, a javni kako bi ostali učesnici u mreži mogli provjeriti legitimnost transakcije.




Sl. 2.10. Digitalni potpis

Može se reći da je privatni ključ prava lozinka korisnika, a potpis da je posrednik koji sprečava otkrivanje korisnikove lozinke. Za javni ključ se može reći da je ustvari dolazna adresa neke transakcije. Da bi korisnik mogao trošiti novac, prvo mora dokazati da je pravi vlasnik adrese, tj. javnog ključa na kojeg je novac poslan. To je ostvareno tako što se generira digitalni potpis iz transakcije poruke i svog privatnog ključa. Time se dolazi do zaključka da je digitalni potpis funkcija poruke i privatnog ključa. Zato što je digitalni potpis funkcija privatnog ključa i poruke, svaki potpis je drugačiji i ne može se ponovno koristiti za drugu transakciju. Time se također sprečava modificiranje poruke dok se prosljeđuje kroz mrežu. Svaka promjena poruke učinit će je nelegitimnom.

Prilikom obavljanja nove transakcije, kako bi transakcija bila autentična, učesnici u transakciji generiraju novi privatni i javni ključ, te se poruka kodira hash funkcijom. Hash funkcija kreira 32 bitnu riječ bilo koje proizvoljne dužine poruke koristeći SHA256 (secure hash algorithm) hash funkciju.

SHA256(“short sentence”)
0x
0acdf28f4e8b00b399d89ca51f07fef34708e729ae15e85429c5b0f40329
5cc9
SHA256("The quick brown fox jumps over the lazy **dog**")
0x
d7a8fbb307d7809469ca9abcb0082e4f8d5651e46d3cdb762d02d0bf37c
9e592
SHA256("The quick brown fox jumps over the lazy **dog.**") **(extra period added)**
0x
ef537f25c895bfa782526529a9b63d97aa631564d5d789c2b765448c86
35fb6c

Sl. 2.11. Primjer poruka kodiranih SHA256

Privatni i javni ključ u kombinaciji s porukom kodiranom SHA256 hash funkcijom čine nemogućim predviđanje enkriptiranog outputa. Jedini način da se poruka dekodira je „pogađanjem“ kako je ranije spomenuto. Moguće je da se traženi ključ pogodi iz prvog pokušaja, ali prosječno „pogađanje“ traje oko 10 minuta. Ključna stvar prilikom „pogađanja“ ključa je koliki je vaš hash rate.

## 2.4.Hash rate

Hash rate ili hash power jedan od ključnih, kritičnih pojmova koje je potrebno razumjeti prije nego što se pristupi Bitcoin mreži. Hash rate je mjerna jedinca kojom se mjeri koliko snage Bitcoin mreža koristi da bi bila funkcionalna, odnosno koliko je snage potrebno da se pronađe dobra kombinacija za otključavanje transakcijskog bloka. Srednje vrijeme koje je potrebno za pronalazak dobre kombinacije u prosjeku iznosi 10 minuta.

Bitcoin mreža uzima mnogo računalne snage jer rješava složene matematičke funkcije za pronalazak prave kombinacije za otključavanje bloka. Praktično, svaki blok je „matematička slagalica“ koju je potrebno riješiti, koju se ne može riješiti bez mnogo računanja. S većom uloženom snagom, moguće je obaviti više proračuna, što znači da veći hash rate omogućava brži pronalazak ključa. Veličina hash rate-a je proporcionalna s zaradom koju je moguće ostvariti mining-om.

## 2.5.ASIC mineri

Bitcoin mining je proces u kojem se Bitcoin transakcija, blok, dodaje u Ledger prijašnjih transakcija, block chain. Mining se može vršiti pomoću procesora, grafičkih kartica ili pomoću specijalne opreme za minanje, ASIC minera. U početku su se koristile grafičke kartice. Međutim, minanje grafičkim karticama ima manu, predugo traje zapisivanje dobivenih rezultata u spremište. To je omogućilo pojavu ASIC minera i njihovu uporabu. Sve više i više su popularne farme koje se bave minanjem kripto valuta. Na tim farmama ASIC mineri dolaze do najvećeg izražaja.

ASIC (application-specific integrated circuit) mineri su integrirani krugovi namjenjeni za obavljanje isključivo jednog posla, mininga. Važno je zapamtiti da su Bitcoin ASIC čipovi isključivo namijenjeni za bitcoin minanje i ni za što drugo. Naravno postoje čipovi kojima se mogu minati i dvije kripto valute, ali to je zato što jedan čip sadrži dva ASICa (jedan za jednu i drugi za drugu kripto valutu). ASIC čipovi odlučuju o kvaliteti minera, njegovoj cijeni i efikasnosti. Razvijanje ASIC čipova je skup proces i ti čipovi su ono što najviše kompjuterske snage troši u Bitcoin minerima. Brzina pogađanja Bitcoin ASIC minera se mjeri u gigahashes per second.

Sl. 2.12. Mining farma

# 3. BITCOIN ALGORITAM

U nastavku je dano objašnjenje nekih od struktura i klasa bitcoin algoritma.

## 3.1.BLOCK.CPP i BLOCK.H

Blok klasa predstavlja blok unutar transakcije.

On se sastoji od CBlock I CBlockHeader klase.

```
class CBlockHeader
{
public:
    // header
    int32_t nVersion;
    uint256 hashPrevBlock;
    uint256 hashMerkleRoot;
    uint32_t nTime;
    uint32_t nBits;
    uint32_t nNonce;
```

Sl. 3.1. Atributi klase CblockHeader

CBlockHeader klasa sadrži atribute prikazane na slici 3.1. U nastavku se nalazi kratki opis prikazanih atributa.

- nVersion – označava verziju blockchain-a prikazanu 32-bitnim integer-om
- hashPrevBlock – sadrži 256 bitni hash prethodnog bloka
- hashMerkleRoot – sadrži 256 bitni hash izvornog čvora u Merkle stablu
- nTime – 32 bitni integer koji sadrži vrijeme kreiranja trenutnog bloka
- nBits – broj bitova koji označava veličinu bloka
- nNonce – 32 bitni integer čija vrijednost se postavlja da bi hash bloka sadržavao vodeće nule

Bitne funkcije koje se nalaze unutar CblockHeader:

- GetHash (Sl. 3.2.) – poziva funkciju SerializeHash iz hash.h datoteke koja vraća serijsku hash vrijednost bloka
- GetBlockTIme – vraća vrijeme kreiranja bloka

```
uint256 CBlockHeader::GetHash() const
{
    return SerializeHash(*this);
}
```

Sl. 3.2. GetHash funkcija unutar CblockHeader-a

```
class CBlock : public CBlockHeader
{
public:
    // network and disk
    std::vector<CTransactionRef> vtx;

    // memory only
    mutable bool fChecked;
```

Sl. 3.3. Atributi i deklaracija klase Cblock

CBlock klasa nasljeđuje CBlockHeader klasu i tako poprima sve atribute te klase. Klasa definira listu referenci na transakcije (Sl. 3.3.) u kojima sudjeluje. Ta lista predstavlja Ledger koji se prenosi zajedno s blokom na mrežu kao i u lokalno spremište (HDD, SSD). Sadrži još i fChecked varijablu koja se čuva trenutno u memoriji i označava da li je blok provjeren ili ne. CBlock je jednostavni objekt koji ne implementira funkcionalnost nad blokom nego samo služi kao model bloka.

## 3.2.TRANSACTION.CPP i TRANSACTION.H

Transaction sadrži sljedeće klase koje definiraju transakciju:

- CTransaction
- CMutableTransaction
- CtxOut
- CtxIn
- COutPoint

COutPoint je klasa koja sadrži kombinaciju hash-a izlazne transakcije i izlazni index te transakcije (Sl. 3.4.). Također, definira logičke operatore usporedbe (<, ==, !=) za jednostavnije uspoređivanje objekata te klase.

```
class COutPoint
{
public:
    uint256 hash;
    uint32_t n;
```

Sl. 3.4. Atributi klase COutPoint

CtxIn je klasa koja predstavlja ulaz transakcije. Sadrži lokaciju izlaza prethodne izlazne transakcije i potpis koji se slaže sa izlaznim javnim ključem (Sl. 3.5.).

```
class CTxIn
{
public:
    COutPoint prevout;
    CScript scriptSig;
    uint32_t nSequence;
    CScriptWitness scriptWitness;
```

Sl. 3.5. Definicija klase CtxIn i njenih atributa

CtxOut klasa predstavlja izlaz transakcije. Sadrži javni ključ koji sljedeći ulaz mora biti u mogućnosti da potpiše kako bih potvrdio transakciju.

```
class CTxOut
{
public:
    CAmount nValue;
    CScript scriptPubKey;
```

Sl. 3.6. Definicija klase CtxOut i njenih atributa

CTransaction klasa predstavlja transakciju koja se šalje na mrežu i koja se sprema u blokove. Transakcija može da sadrži višestruke ulaze i izlaze.

```
class CTransaction
{
public:

    const std::vector<CTxIn> vin;
    const std::vector<CTxOut> vout;
    const int32_t nVersion;
    const uint32_t nLockTime;

private:
    /** Memory only. */
    const uint256 hash;
    const uint256 m_witness_hash;

    uint256 ComputeHash() const;
    uint256 ComputeWitnessHash() const;
```

Sl. 3.7. Definicija klase CTransaction i njenih javnih i privatnih atributa

Sadrži sljedeće atribute:

- Vin – listu ulaznih transakcija klase CtxIn
- Vout – listu izlaznih transakcija klase CtxOut
- nVersion – verziju trenutnog tipa transakcije
- nLockTime – vrijeme zaključavanja transakcije
- hash – privatna varijabla u koju se sprema hash vrijednost transakcije
- m_witness_hash – privatna varijabla koja predstavlja hash svjedoka prisutnih u mreži

```
uint256 CTransaction::ComputeHash() const
{
    return SerializeHash(*this, SER_GETHASH, SERIALIZE_TRANSACTION_NO_WITNESS);
}

uint256 CTransaction::ComputeWitnessHash() const
{
    if (!HasWitness()) {
        return hash;
    }
    return SerializeHash(*this, SER_GETHASH, 0);
}
```

Sl. 3.8. Funkcije klase Ctransaction

Slika 3.8. definira funkcije koje računaju hash transakcijskog objekta pozivom funkcije SerializeHash.

CMutableTransaction je klasa slična klasi CTransaction. Jedina razlika je u tome što ne sadrži privatne konstantne varijable hash i m_witness_hash.

## 3.3.CHAIN.CPP i CHAIN.H

Chain sadrži sljedeće klase:

- CBlockFileInfo
- CDiskBlockPos
- CBlockIndex
- CDiskBlockIndex
- Cchain

```
class CBlockFileInfo
{
public:
    unsigned int nBlocks;
    unsigned int nSize;
    unsigned int nUndoSize;
    unsigned int nHeightFirst;
    unsigned int nHeightLast;
    uint64_t nTimeFirst;
    uint64_t nTimeLast;
```

Sl. 3.9. Definicija klase CBlockFileInfo i njeni atributi

CBlockFileInfo je klasa koja sadrži osnovne informacije o block datoteci spremljenoj na disku.

Klasa sadrži sljedeće atribute:

- nBlocks – broj blokova u datoteci
- nSize – veličina datoteke
- nUndoSize – broj bajtova u undo datoteci
- nHeightFirst – najmanja visina bloka u datoteci
- nHeightLast – najveća visina bloka u datoteci
- nTimeFirst – najranije vrijeme bloka
- nTimeLast – zadnje vrijeme bloka

```
void AddBlock(unsigned int nHeightIn, uint64_t nTimeIn) {
    if (nBlocks==0 || nHeightFirst > nHeightIn)
        nHeightFirst = nHeightIn;
    if (nBlocks==0 || nTimeFirst > nTimeIn)
        nTimeFirst = nTimeIn;
    nBlocks++;
    if (nHeightIn > nHeightLast)
        nHeightLast = nHeightIn;
    if (nTimeIn > nTimeLast)
        nTimeLast = nTimeIn;
}
```

Sl. 3.10. Funkcija unutar klase CBlockFileInfo

Funkcija na slici 3.10. prikazuje funkciju za dodavanje bloka u trenutnu datoteku.

CDiskBlock struktura opisuje lokaciju block datoteke na disku.

```
struct CDiskBlockPos
{
    int nFile;
    unsigned int nPos;
```

Sl. 3.11. Definicija CDiskBlockPos strukture i njeni atributi

CBlockIndex je klasa koja predstavlja index trenutnog bloka unutar blockchain-a.

CDiskBlockIndex je klasa koja predstavlja isto što i CBlockIndex klasa, samo unutar datoteke na disku.

CChain je klasa koja predstavlja indeksirani lanac blokova unutar memorije.

```
class CChain {
private:
    std::vector<CBlockIndex*> vChain;
```

Sl. 3.12. Definicija klase CChain sa atributima

Slika 3.12. prikazuje jedinu privatnu podatkovnu varijablu *vChain* koja je sadržana unutar klase CChain.

vChain je vektor (dinamički niz elemenata) koji sadrži CBlockIndex pokazivače.

CChain klasa definira i neke bitne funkcije za prolazak kroz lanac blokova.

```
CBlockIndex *Genesis() const {
    return vChain.size() > 0 ? vChain[0] : nullptr;
}
```

Sl. 3.13. Genesis funkcija

Funkcija na slici 3.13. vraća pokazivač tipa CBlockIndex početnog bloka u lancu ili null pokazivač ako ne postoji tj. lanac je prazan.

```
CBlockIndex *Tip() const {
    return vChain.size() > 0 ? vChain[vChain.size() - 1] : nullptr;
}
```

Sl. 3.14. Tip funkcija

Funkcija na slici 3.3.6. vraća pokazivač tipa CBlockIndex zadnjeg bloka u lancu.

```
CBlockIndex *operator[](int nHeight) const {
    if (nHeight < 0 || nHeight >= (int)vChain.size())
        return nullptr;
    return vChain[nHeight];
}
```

Sl. 3.15. Override operatora []

Slika 3.15. prikazuje override operatora koji služi za vraćanje vrijednosti bloka iz liste na traženoj visini lanca.

```
CBlockIndex *Next(const CBlockIndex *pindex) const {
    if (Contains(pindex))
        return (*this)[pindex->nHeight + 1];
    else
        return nullptr;
}
```

Sl. 3.16. Next funkcija

Funkcija na slici 3.16. služi za dohvaćanje sljedećeg bloka u lancu koji se nalazi iza proslijeđenog pokazivača bloka.

```
CBlockIndex* CChain::FindEarliestAtLeast(int64_t nTime) const
{
    std::vector<CBlockIndex*>::const_iterator lower = std::lower_bound(vChain.begin(), vChain.end(), nTime,
        [](CBlockIndex* pBlock, const int64_t& time) -> bool { return pBlock->GetBlockTimeMax() < time; });
    return (lower == vChain.end() ? nullptr : *lower);
}
```

Sl. 3.17. FindEarliestAtLeast funkcija

Funkcija na slici 3.17. vraća pokazivač na blok koji se nalazi najbliže proslijeđenom vremenu u parametru nTime.

```
const CBlockIndex *CChain::FindFork(const CBlockIndex *pindex) const {
    if (pindex == nullptr) {
        return nullptr;
    }
    if (pindex->nHeight > Height())
        pindex = pindex->GetAncestor(Height());
    while (pindex && !Contains(pindex))
        pindex = pindex->pprev;
    return pindex;
}
```

Sl. 3.18. FindFork funkcija

Funkcija na slici 3.18. pronalazi index prethodnog bloka u kojem nije odlučeno koji blok je riješio ključ za otključavanje prethodne transakcije.

## 3.4.HASH.CPP i HASH.H

Datoteka koji sadrži funkcije za digitalno potpisivanje transakcije koristeći SHA256 algoritam.

```
/** A hasher class for Bitcoin's 256-bit hash (double SHA-256). */
class CHash256 {
private:
    CSHA256 sha;
public:
    static const size_t OUTPUT_SIZE = CSHA256::OUTPUT_SIZE;
    ...
    CHash256& Write(const unsigned char *data, size_t len) {
        sha.Write(data, len);
        return *this;
    }
    ...
};
```

Sl. 3.19. Klasa CHash256

Klasa na slici 3.19. se koristi za digitalno potpisivanje transakcije.

# 4. ZAKLJUČAK

U ovom diplomskom radu na obrađena je Bitcoin mreža i bitcoin algoritam. Bitcoin mreža je digitalna, decentralizirana, anonimna platna mreža koja koristi svoju kripto valutu, bitcoin. Osnovni principi Bitcoin algoritma je da je Ledger javan tako da ga svi mogu vidjeti i odrediti da li je transakcija legitimna ili ne, te da se Ledger nalazi u više node-ova unutar mreže kako bi se otklonila potreba za posrednikom. Također je obrađen koncept minera čija je uloga nelegitimne transakcije učiniti legitimnim i upisati ih u Ledger. S obzirom na razvoj i popularnost bitcoina, gotovo sa sto posto sigurnosti se može reći da je bitcoin budućnost ekonomije i marketinga zbog svoje sigurnosti, efikasnosti i brzine prilikom obavljanja transakcija. Ukoliko potražnja i upotreba kripto valuta nastavi rasti kripto valute bi mogle zamijeniti stvarni novac, te ne bismo imali više potrebe za njime.

# SAŽETAK

**Naslov:** Bitcoin algoritam

Bitcoin (BTC) je digitalna, decentralizirana, anonimna platna mreža, a ujedno i kripto valuta koja je korištena u toj mreži. BTC se koristi za elektronsko plaćanje usluga. U pojedinim mjestima, također, je moguće platiti noćenje u lancu hotela, kupiti pizzu, kupiti VPB servis...

Diplomski rad podijeljen je na dva glavna dijela.

U prvom dijelu objašnjeno je sve što potrebno za razumjeti što je to Bitcoin.

U drugom dijelu opisani su bitniji dijelovi koda Bitcoin algoritam.

**Ključne riječi:** Bitcoin, Blockchain, transakcija, rudarenje, hash rate

# ABSTRACT

**Title:** Bitcoin algorithm

Bitcoin (BTC) is digital, decentralized, anonymous payment network and at the same time crypto value that is used in this network. BTC is used for electronic payment service. In some places, it is possible to pay night in hotel chain, buy pizza, buy VPN service etc.

Graduate paper is divided into two main parts:

The first part explains what you need to know about Bitcoin.

The second part explains some of the Bitcoin algorithm code.

**Key words:** Bitcoin, Blockchain, transaction, mining, hash rate

# ŽIVOTOPIS

Stefan Radošević je rođen u Vukovaru 26. ožujka 1995. godine u kojem živi od tada. 2001. godine kreće u Osnovnu školu Siniše Glavaševića u Vukovaru. 2009. godine kreće u Tehničku školu Nikole Tesle u Vukovaru, koju završava 2013. godine. Iste godine upisuje Elektrotehnički fakultet u Osijeku, preddiplomski studij, smjer Računarstvo. 2016. godine završava preddiplomski studij, te upisuje diplomski studij, smjer Računalno inženjerstvo koji završava 2018. godine.

# LITERATURA

[1] Bitcoin, https://bitcoin.org/bitcoin.pdf , srpanj 2018.
[2] Blockchain, https://blockgeeks.com/guides/what-is-blockchain-technology/ , srpanj 2018.
[3] Blockchain, https://www.bug.hr/tehnologije/sto-je-u-stvari-blockchain-i-kako-radi-3011 , srpanj 2018.
[4] Digitalni potpis, CARNet,2007.
[5] Bitcoin algoritam, https://github.com/bitcoin/bitcoin , srpanj 2018.
[6] Andreas M. Antonopoulos, Mastering Bitcoin, O'Reilly Media Inc.,2015.
[7] Satoshi Nakamoto, Bitcoin: A Peer-to-Peer Electronic Cash System, 2008.
[8] M. Nielsen, How the Bitcoin protocol actually works, 2013.

# PRILOG

Prilog na CD-u.