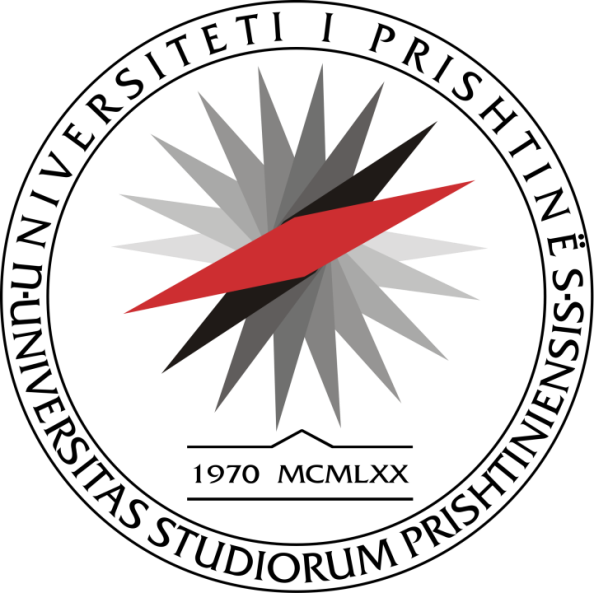
**Universiteti i Prishtinës “Hasan Prishtina”  
Fakulteti i Inxhinierisë Elektrike dhe Kompjuterike**

**Departamenti i Kompjuterikës**



RAPORT: Dizajnimi **KLIENT-SERVER**

LËNDA: Rrjeta Kompjuterike

Profesori i lëndës: Prof. Blerim Rexha Studentja: Teuta Ukshini

Asistenti i lëndës: MSc. Haxhi Lajqi Nr.ID-së: 170714100084

Prishtinë, 2019

**Abstrakt**

Për të lidhur nje aplikacion në rrjetë lokale ose në internet , ai aplikacion e përdor komponentën softuerike të quajtur Socket. Socket i hap lidhjet ne rrjetë për t’i lejuar aplikacionët që të mund të shkruajnë dhe të lexojnë nëpermjet rrjetës. Socket janë një pikë fundore ose Endpoint e komunikimit në mes të dy programeve në rrjetë. Një Socket perbehet prej numrit te portit (ang. port number) dhe prej një IP adresë dhe këto dy komponente e identifikojnë në mënyrë unike Socket-in. Për shkruarjën e programeve që komunikojnë në rrjetë, programerët duhet të vendosin se me cilin protokol të komunikimit do të komunikojnë ***aplikacionet*.** Kemi dy protokolle të cilat mund ti përdorim, protokollin **TCP** dhe protokollin **UDP**.

**Hyrje**

Për realizimin e klientit dhe serverit është përdorur gjuha programuese **PYTHON** dhe editori **Visual Studio 2017** ndërsa për krijimin e lidhjes mes klientit dhe serverit janë përdorur soketat(sockets) si pika fundore në secilën anë të cilat mundësojnë qarkullimin e të dhënave mes klientit dhe serverit.

**TCP** protokolli është protokoll që siguron një kanal të sigurtë të komunikimit në mes klientit dhe serverit ndërsa protokolli **UDP** nuk siguron nje kanal të tillë mirpo këtu rrjedha e informatave bëhet pa garancionin se ato do të arrijnë në destinacion. Në gjatë këtij ushtrimi do të përdorim protokollin **TCP**.

Kemi përdorur protokollin e lidhjes **sock.STREAM** për lidhjen **TCP** dhe **sock.DGRAM** për lidhjen **UDP**.

**Përshkrimi i socket klient dhe socket server**

Janë përdorur libraritë e gatshme: socket me metodat e saj, random, thread, sys.

Nga libraritë e lartë përmendura kemi përdorur edhe metodat: bind, listen, encode, recv,recvfrom,rsplit,clientthread për **TCP**, ndërsa për **UDP** , toupper, gethostname, gethostbyname etj.

**Metodat**

**METODA IPADRESA**

Me anën e metodës IPADRESA klienti është në gjendje të gjej IP e pajisjes së tij. Është përdorur gethostbyname e cila kthen emrin e kompjuterit të klientit për arsye që metoda të marr IP adresës.

**METODA NUMRIPORTIT**

Metoda NUMRIPORTIT kthen numrin e portit të cilin klienti është duke e përdorur .

**METODA BASHTINGËLLORE**

Metoda BASHTINGELLORE është metodë që na mundëson të gjejmë numrin e bashtingëlloreve në një fjali. Është deklaruar dhë një variabël tjetër me “z” dhe ka marrë vlerën 0. Vlera ë kësaj variable shtohet për një sa herë që programi hasë në një bashtingëllore gjatë krahasimit.

Në lidhjen TCP metoda është dhënë në atë mënyrë që fillimisht ceket metoda pastaj programi kërkon fjalinë nga klienti ndërsa në lidhjen UDP për arsye se serveri nuk krijon lidhje me serverin kjo nuk ka mundur të arrihet dhe metoda është dhënë në formatin BASHTINGËLLORE {hapsirë} fjalia e klientit.

**METODA PRINTIMI**

Më anë të metodës PRINTIMI klientit i ofrohet mundësia të shtypë një fjali dhe pas kësaj i shfaqet në ekran fjalia të cilën ai e ka shtypur. Me anë të komandës print() printohet fjalia e klientit.

**METODA EMRIKOMPJUTERIT**

Metoda EMRIKOMPJUTERIT është bërë me qëllimin që klienti të këtë mundësi të dijë emrin e pajisjes së tijËshtë përdorur funksioni i gatshëm gethostname() e cila e kthen emrin e pajisjes.

**METODA KOHA**

Metoda KOHA kthen kohën. Kemi përdorur funksionin e gatshëm të kësaj librarie datetime.noë() e cila kthen kohën dhe datën aktuale por është formatuar në mënyrë që të kthej vetëm kohën e serverit.

**METODA LOJA**

Qëllimi i metodës LOJA është që të gjejë numra rëndom në mes numrit 1 dhe 49. Është importuar libraria e gatshme random pastaj është përdorur funksioni i gatshëm random.sample(range(,),).

**METODA FIBONACCI**

Metoda FIBONACCI ka për funksion të gjejë fibonaccin e numrit të shtypur nga klienti. Në lidhjen TCP kjo metodë është dhënë në atë mënyrë që së pari i tregohet serverit se dëshirojë të përdorë metodën Fibonacci pastaj i kërkohet klientit të shtyp numri, kursë në lidhjën UDP për arsye që lidhja më klientin dhe serverin nuk krijohet në fillim nuk është e mundur të dërgohet më shumë se një e dhënë kjo metodë është dhënë në formatin FIBONACCI {hapsire} numri.

**METODA KONVERTIMI**

Kjo metodë i ofron klientit mundësinë të konvertojë madhësi të ndryshme. Kjo metodë lejon konvertimin e njësive Kiloëatt në Horsepoëer, Horsepoëer në Kiloëatt, Degrees në Radians, Radians në Degrees, Gallons në Liters dhe Liters në Gallons.

Në lidhjen TCP kjo metodë është dhënë në atë mënyrë që klienti të tregojë që dëshiron të përdorë metodën konvertimi pastaj të jipet opsioni dhe më pas sasia e madhësisë kurse tek lidhja UDP është dhënë në formatin KONVERTIMI {hapsirë} opsioni {hapsirë} sasia e madhësisë.

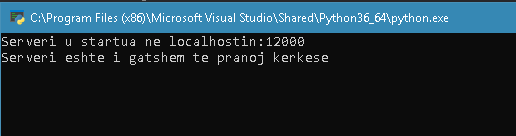
**METODA FAKTORIEL**

Me ane te metodes FAKTORIEL, ne rastin kur klienti jep numrin qe deshironë ta faktorizojë i kthen numrin që klienti dëshiron të marr.

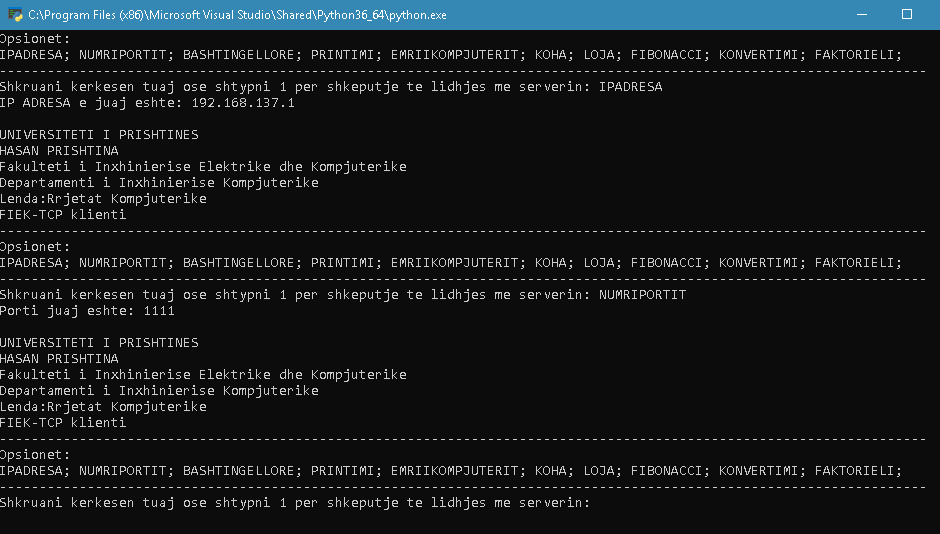
**Testimi**

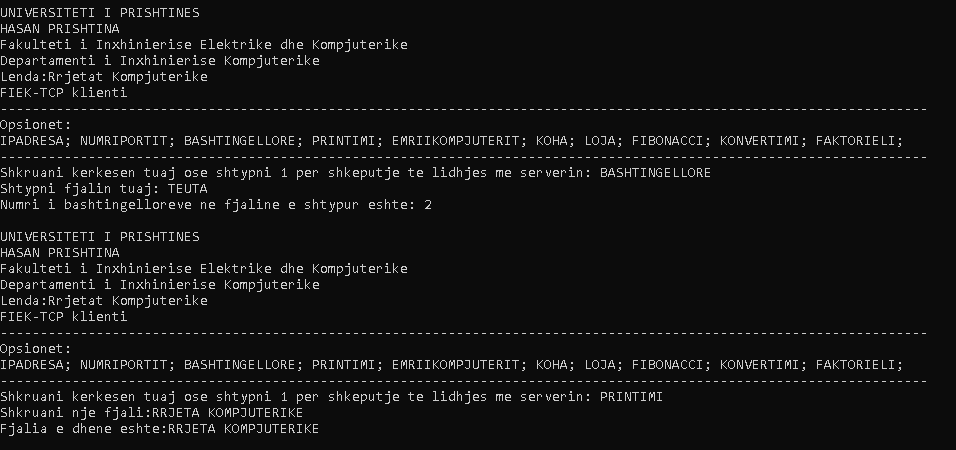
**TCP**

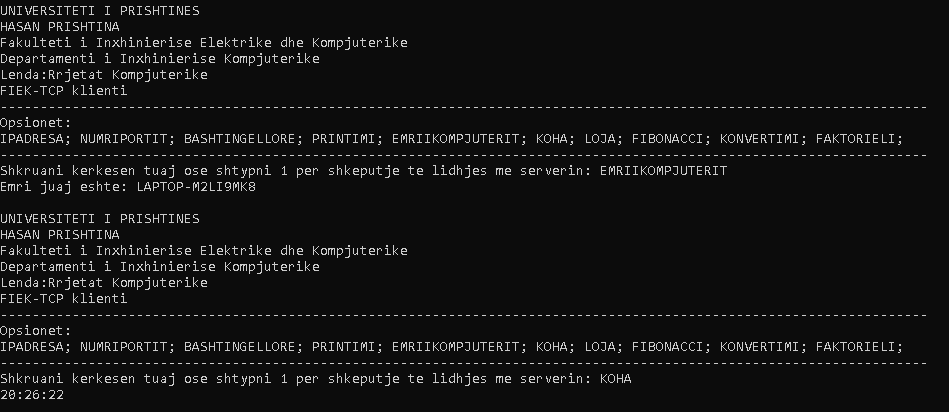
***Serveri:***

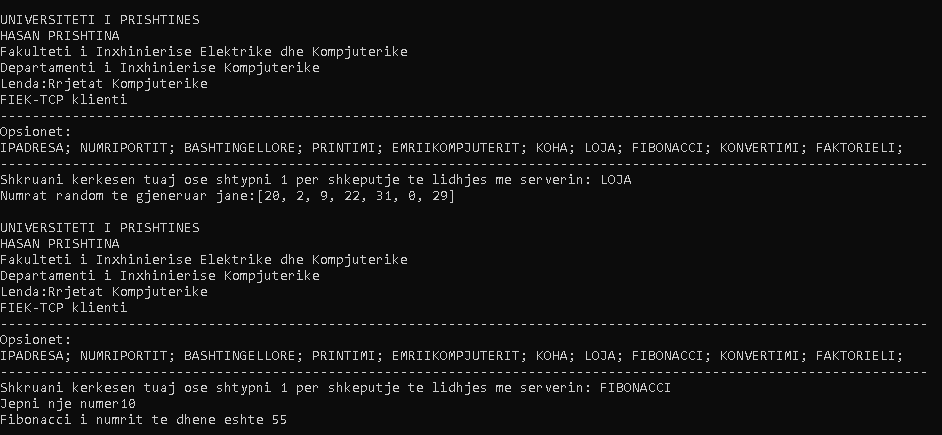


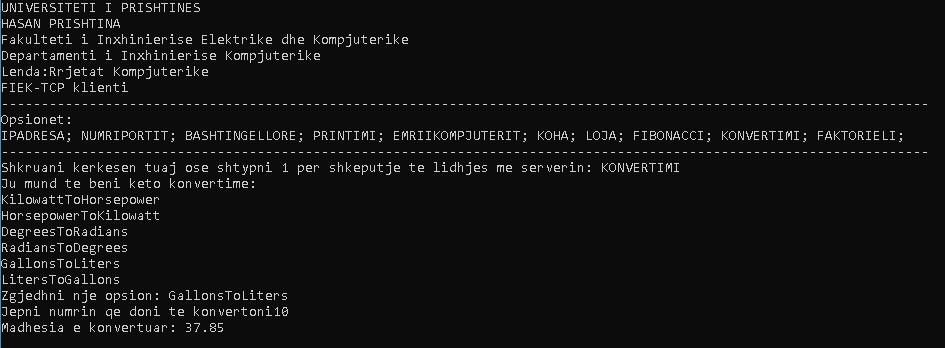
***Klienti:***

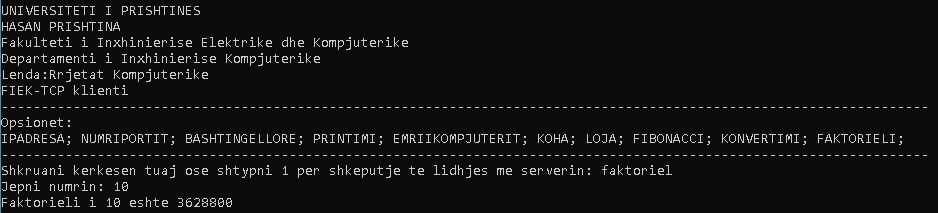






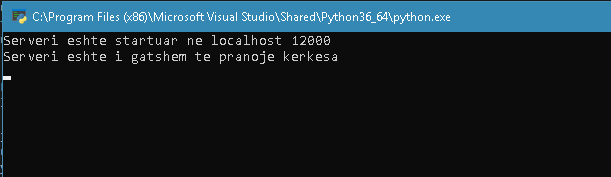




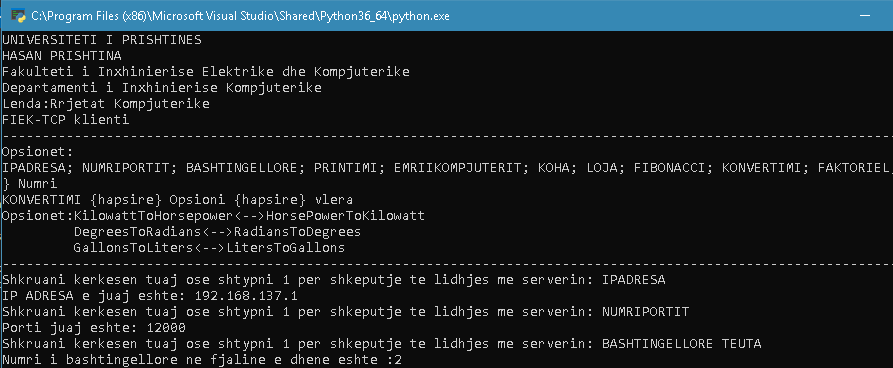


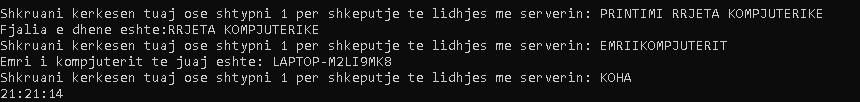
**UDP**

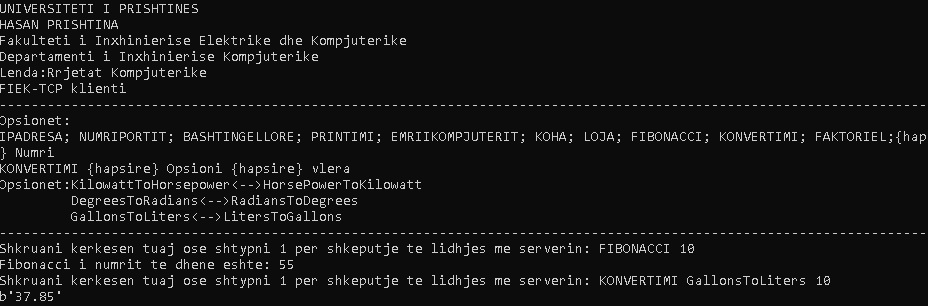
***Serveri:***

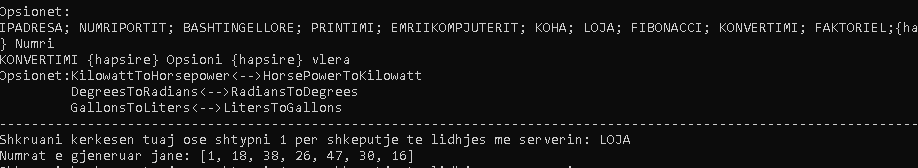


***Klienti:***











**Referencat**

<https://www.pythonforbeginners.com/>  
<https://www.tutorialspoint.com/python/>

<https://www.youtube.com/watch?v=Emuw71lozdA>

<https://www.geeksforgeeks.org/udp-server-client-implementation-c/>