

UNIVERSITETI I PRISHTINËS “HASAN PRISHTINA”

FAKULTETI I INXHINIERISË ELEKTRIKE DHE KOMPJUTERIKE

DEPARTAMENTI : INXHINIERI KOMPJUTERIKE



LËNDA : RRJETA KOMPJUTERIKE

DIZAJNIMI KLIENT – SERVER (Projekti 1)

Studentja : Leonita Nika

Ass:Haxhi Lajqi

Prishtinë, 2020

Përmbajtja

- 1. Hyrje e shkurtër rreth projektit**
- 2. Protokollet të transportit**
 - 2.1. TCP protokoli
 - 2.2. UDP protokoli
- 3. Përshkrimi i FIEK-TCP Serverit dhe FIEK-TCP Klientit**
 - 3.1 FIEK-TCP klienti
 - 3.2 FIEK-TCP server
- 4. Përshkrimi i FIEK-UDP Serverit dhe FIEK-UDP Klientit**
 - 4.1 FIEK-UDP klienti
 - 4.2 FIEK-UDP server
- 5. Metodat e përdorura në protokollin FIEK**
 - 5.1 Metoda IPADRESA
 - 5.2 Metoda PORT
 - 5.3 Metoda COUNT
 - 5.4 Metoda REVERSE
 - 5.5 Metoda PALINDROME
 - 5.6 Metoda TIME
 - 5.7 Metoda GAME
 - 5.8 Metoda GCF
 - 5.9 Metoda CONVERT
- 6. Metodat shtesë**
 - 6.1 Metoda LETTERS
 - 6.2 Metoda PËRQINDJA

1.Hyrje e shkurtër rreth projektit

Qëllimi i këtij projekti është që ti kuptojmë bazamentin e funksionimit të arkitektures klient-server. Është një protokoll i thjeshtë komunikimi i cili për bazë ka fjalën kyqe në fillim të kërkesës e cila përcakton metodën të cilën do ta kthej serveri. FIEK është një protokoll i thjeshtë që i lejon klientit dhe serverit ti testojnë lidhjet e tyre. Projekti është realizuar në dy protokole, në atë TCP dhe UDP.

Përmban programet:

- FIEK-TCP Klienti
- FIEK-TCP Serveri
- FIEK-UDP Klienti
- FIEK-UDP Serveri

Vegla e përdorura për zhvillim të projektit:

- PyCharm Community Edition 2019.3.3

2. Protokollet të transportit

Në internet kemi dy protokolle të transportit:

2.1.TCP protokolli

TCP protokoli përfshin shërbimin e orientuar-në-lidhje dhe shërbim të transferit të besueshëm të të dhënave. Kur dy hoste komunikojnë duke përdorur TCP, lidhja midis tyre duhet të vendoset përpara se të shkëmbehen të dhënat. Ndonjëherë përshkak të mbingarkimit të rrjetës, IP paketat mund të humbasin, dyfishohen, ose të pranohen në renditje të gabuar.

Nëse e dhëna nuk dërgohet, burimit i asaj të dhëne lajmërohet për këtë dështim në transimision. E nëse TCP pranuesi pranon të gjithë sekuencën e paketave të transmetuara, ai i dërgon këto të dhëna të aplikacioni marrës. TCP garanton që e dhëna nuk do të ndryshojë në asnjë bitë gjatë dërgimit, pra dërgohet në renditje të saktë.

2.2.UDP protokoll

UDP është një protokoll i thjeshtë, i cili nuk është i orientuar në lidhje dhe siguron mbingarkesë të vogël në transmetimin e të dhënave. Paketat dërgohen vetëm tek marrësi. UDP nuk garanton dërgimin e të dhënave deri në destinacion dhe nëse ndonjë pakete nuk ka arritur tek marrësi ai nuk bënë ritransmetimin e paketave të humbura. Ku kjo humbje në shumicën e rasteve bëhet për shkak të shpejtësisë së madhe që ofron ky protokoll .

3.Përshkrimi i FIEK-TCP Serverit dhe FIEK-TCP Klientit

3.1.FIEK-TCP klienti

Importimi I librarive të nevojshme:

```
import socket
from _thread import *
from datetime import datetime
import random
from collections import Counter
```

Ne fillim,pas importimit të librarive të nevojshme, përcaktojmë portin dhe ip adresën vlerën e të cilave në mund ti marrim default apo mund t'i nëse dëshirojmë mund ti cakojmë ne .Thamë që TCP protokoll i është i orientuar në lidhje ,prandaj pas deklarimit të portit dhe ip adreses ,ajo që ne duhet të bëjme është krijimi i një lidhje mes serverit dhe klientit.Pas krijimit te lidhjes ne mund t'i parashtrojmë serverit kërkesa të cilat serveri patjetër na dërgon një pergjigjje varësisht së a mund ta plotësoje atë kërkesë apo jo.

Krijimi I soketit te klientit :

- serverSocket = socket.socket(socket.AF_INET, socket.SOCK_STREAM)

Lidhja e klientit me serverin:

- serverSocket.connect((host, serverPort))

Metoda sendall () i dërgon të dhënat në socket kjo metode vazhdon të dërgojë të dhëna nga bytes derisa të gjitha të dhënat janë dërguar ose të ndodhë ndonjë gabim.

- serverSocket.sendall(str.encode(kerkesa))

Pranimi I të dhënave më ane të metodes recv()

- serverAnsëerByte = serverSocket.recv(128)

3.2.FIEK-TCP serveri

Ngjajshëm sikur tek Klienti importojmë librarite e nevojshme dhe caktojmë portin dhe ip adresen pastaj krijojmë sokenin e serverit:

➤ `serverSocket = socket.socket(socket.AF_INET, socket.SOCK_STREAM)`

Metoda `bind()` shërben për lidhjen e sokenit me cilindo klient që e përdore një port të caktuar.

➤ `serverSocket.bind((host, serverPort))`

Përcaktimi i numrit të klientëve të cilët mundën me u lidh me serverin mundësohet me anë të metodës `listen()`:

➤ `serverSocket.listen(4)`

Pranimi i lidhjes me klientin mundësohet me anë të:

➤ `connectionSocket, adresa = serverSocket.accept()`

4.Përshkrimi i FIEK-UDP Serverit dhe FIEK-UDP Klientit

4.1.FIEK-UDP klienti

Pas importimit të librarive të nevojshme, përcaktojmë portin dhe ip adresën vlerën e të cilave ne mund ti marrim default apo mund t'i nëse dëshirojmë mund ti caktojmë ne.

Pasi protokollin UDP përdor një model të thjeshtë të lidhjes, ku nuk vendos lidhje po thjeshtë siguron mbrojtjen e të dhënave dhe portin për të adresuar funksione të ndryshme ndërmjet datagramit të burimit dhe destinacionit nuk kemi nevojë të krijojmë lidhje Server-Klient

Krijimi i sokenit tek protokollin UDP:

➤ `serverSocket = socket.socket(socket.AF_INET, socket.SOCK_DGRAM)`

Më anë të metodës `sendto()` kemi dërguar të dhëna në një UDP socket. Që kjo metode të përdoret, socket nuk duhet të jetë në gjendje të lidhur tashmë.

➤ `serverSocket.sendto(str.encode(kerkesa), (host, serverPort))`

Klienti përdore metodën `recvfrom()` për marrjen e të dhënave nga një server UDP:

➤ `pergjigja,adresa= serverSocket.recvfrom(2048)`

4.2.FIEK-UDP server

Pasi protokollit UDP përdor një model të thjeshtë të lidhjes, ku nuk vendos lidhje po thjeshtë siguron mbrojtjen e të dhënave dhe portin për të adresuar funksione të ndryshme ndërmjet datagramit të burimit dhe destinationit nuk kemi nevojë të krijojmë lidhje Server-Klient

Importimin e librave ,caktimin e portit dhe ipadreses ,krijimin e socket-it i përcaktojmë ngjajshëm sikur tek UDP klienti .Metodat sendto() dhe recvfrom() përdoren po ashtu edhe tek server mirëpo këtu shërbejnë për dërgimin/marrjen e të dhënave nga klienti.

```
import socket
from datetime import datetime
import random
from collections import Counter

host = 'localhost'
serverPort = 13000
serverSocket = socket.socket(socket.AF_INET, socket.SOCK_DGRAM)
try:
    serverSocket.bind((host, serverPort))
except socket.error as e:
    print("Nuk mund te lidheni")
print('Serveri eshte i gatshem te pranoj kerkesa')
```

5. Metodat e përdorura në protokollin FIEK

5.1. Metoda IPADDRESS

Adresa IP është adresa e sistemit në Rrjet ,për ta lehtësuar punën me IP adresa,zakonisht ato I shkruajme si katër numra decimale të ndarë më një pikë, psh 127.0.0.1.

➤ Realizimi përmes kodit:

```
if klienti[0].upper() == 'IPADDRESS':
    pergjigjja = " IP adresa e klientit është : " + adresa[0]
    if klienti[1:]:
        pergjigjja = "Ju lutem shkruani kerkesen ne cregull!"
```

Për aktivizimin e kësaj metode që kthen si përgjigje IP adresën e klientit në formë dhjetore, mjafton që klienti ta shkruaj kërkesën IPADDRESS , atëherë IP gjendet më antarin e parë të adresës (IP + Port përcakton adresën e shërbimit të veçantë në cdo sistem) dhe kthen si përgjigje IP adresën , mirëpo në qoftë se klienti shkruan më shumë pas metodës IPADDRESS kam kushtezuar kërkesën duke mos kthyer IP adresën por një mesazh për informim.

➤ Rezultati pas ekzekutimit të kësaj kërke :

```

Ju lutem shkruani njerën nga kërkesat :IPADDRESS
IP adresa e klientit është :127.0.0.1
Vazhdoni me kerkese tjeter ose shtyp 0 per dalje.

Ju lutem shkruani njerën nga kërkesat :IPADDRESS 1
Ju lutem shkruani vetem kerkesen!

```

5.2.Metoda PORT

Porti është adresa e shërbimit brenda sistemit.

- Realizimi përmes kodit:

```

elif klienti[0].upper() == 'PORT':
    pergjigjja = "Klienti është duke përdorur portin " + str(adresa[1])
    if klienti[1:]:
        pergjigjja = "Ju lutem shkruani vetem kerkesen!"

```

Për aktivizimin e kësaj metode që kthen si përgjigje portin e klientit në formën e një numri të plotë , mjafton që klienti ta shkruaj kërkesën PORT , atëherë porti gjëndet me antarin e dytë të adresës (IP + Port përcakton adresën e shërbimit të veçantë në çdo sistem)dhe kthen si përgjigje Portin,mirëpo në qoftë se klienti shkruan më shumë pas metodës PORT kam kushtëzuar kërkesën duke mos kthyer Portin por një mesazh për informim

- Rezultati pas ekzekutimit të kësaj kërkesë :

```

Ju lutem shkruani njerën nga kërkesat :PORT
Klienti është duke përdorur portin 59169
Vazhdoni me kerkese tjeter ose shtyp 0 per dalje.

Ju lutem shkruani njerën nga kërkesat :PORT 1
Ju lutem shkruani vetem kerkesen!
Vazhdoni me kerkese tjeter ose shtyp 0 per dalje.

```

5.3.Metoda COUNT

Metoda COUNT është një metode e cila mundëson numerimin e zanoreve dhe bashkëtingëlloreve në një fjali .Për realizimin e kodit së pari kam inicializuar më 0 numrin e zanorëve dhe bashkëtingëlloreve ,pastaj me anë të vargut mundësohet që numërimi të vazhdojë deri në gjatësinë e asaj fjalie dhe në fund më

ane të statementit if else mundësohet përcaktimi se cilat shkronja duhet të numërohen tek zanorët cilat tek bashkëtingëlloret.

➤ Realizimi përmes kodit:

```
elif klienti[0].upper() == 'COUNT':
    a = ""
    a = a.join(klienti[1:])
    zanore = 0
    bashkëtingëllore = 0
    if len(a) != 0:
        for i in range(0, len(a)):
            if a[i] == "A" or a[i] == "E" or a[i] == "I" or a[i] == "O" or a[i] == "U" or a[i] == "Y":
                zanore = zanore + 1
            else:
                if a[i] >= 'A' and a[i] <= 'Z':
                    bashkëtingëllore = bashkëtingëllore + 1
        pergjigjja = "Teksti i pranuar përmban " + str(zanore) + " zanore dhe \"\n\" + str(bashkëtingëllore) + " bashkëtingëllore "
    else:
        pergjigjja = "JU lutem shenoni nje fjali pas kërkesës !"
```

Klienti pas paraqitjes së kësaj kërkesë mund të shkruajë një tekst të cilit dëshiron që t'ia numëroj zanorët dhe bashkëtingëlloret, nuk kam kufizuar mundësinë e shkrimit të klientit, ai mund të shkruajë deri në 128 karaktere në atë kërkesë, përndryshe në qoftë se nuk jep ndonjë tekst I kthehet përgjigjja që duhet të shkruajë një fjali pas kërkesës.

➤ Rezultati pas ekzekutimit të kësaj kërkesë :

```
Ju lutem shkruani njerën nga kërkesat :COUNT Projekti i pare nga rrjeta kompjuterike
Teksti i pranuar përmban 14 zanore dhe 20 bashkëtingëllore
Vazhdoni me kerkese tjeter ose shtyp 0 per dalje.

Ju lutem shkruani njerën nga kërkesat :COUNT
JU lutem shenoni nje fjali pas kërkesës !
Vazhdoni me kerkese tjeter ose shtyp 0 per dalje.
```

5.4. Metoda REVERSE

Me anë të metodës REVERSE mundësohet që teksti të cilin e jep klienti të kthehet në formën reverse.

Kjo arrihet nëse e marrim tekstin ashtu si e ka dhënë klienti dhe me anë të funksionit rev:

reverse=teksti[::-1] mundësohet që teksti I dhënë të kthehet në formën reverse.

➤ Realizimi përmes kodit:


```

elif klienti[0] == "REVERSE":
    try:
        if klienti[1]:
            pergjigjja = "Teksti i kthyer ne formen reverse është : " + str(rev(klienti[1:]))
        except Exception:
            pergjigjja = "Ju lutem shenoni nje fjali pas fjalës " + str(klienti[0])

def rev(x):
    teksti = str.join(" ", x)
    reverse = teksti[::-1]
    return reverse

```

Klienti pas thirrjes së kësaj metode duhet ta shkruaj fjalinë që dëshiron ta kthej në formën REVERSE. Nëse klienti nuk jep tekst atëherë I kthehet pergjigjja që duhet te shkruaj një fjali pas kërkesës.

➤ Rezultati pas ekzekutimit të kësaj kërkesë :

```

Ju lutem shkruani njerën nga kërkesat :REVERSE Projekti i pare nga rrjeta kompjuterike
Teksti i kthyer ne formen reverse është : EKIRETUJPMOK ATEJRR AGN ERAP I ITEKJORP
Vazhdoni me kerkese tjeter ose shtyp 0 per dalje.

Ju lutem shkruani njerën nga kërkesat :REVERSE
Ju lutem shenoni nje fjali pas fjalës REVERSE
Vazhdoni me kerkese tjeter ose shtyp 0 per dalje.

```

5.5. Metoda PALINDROME

Një palindrome është një fjalë, numër, frazë ose sekuencë tjetër e karaktereve që lexohet njëjtë në të dy anët.

Për ta kontrolluar nëse teksti i dhënë është palindrome e marrim tekstin që e shtyp klienti dhe nëse teksti i dhënë është i barabart me formën reverse të tij atëherë ai tekst është palindrom .

➤ Realizimi përmes kodit:

```
elif klienti[0] == "PALINDROME":
    try:
        if klienti[1]:
            teksti=palindrom(klienti[1:])
            reverse=rev(klienti[1:])
            if teksti == reverse:
                pergjigjja = "Teksti i dhënë është palindrome"
            else:
                pergjigjja = "Teksti i dhënë nuk është palindrome"
        except Exception:
            pergjigjja = "Ju lutem shenoni nje tekst pas kërkesës " + klienti[0]

def palindrom(x):
    teksti = str.join(" ", x)
    return teksti

def rev(x):
    teksti = str.join(" ", x)
    reverse = teksti[::-1]
    return reverse
```

Klienti pas thirrjes së kësaj metode duhet ta shkruaj tekstin që dëshiron ta kthej në formën PALINDROME. Nëse klienti nuk jep tekst atëherë i kthehet pergjigjja që duhet të shkruaj një fjali pas kërkesës.

➤ Rezultati pas ekzekutimit të kësaj kërkesë :

```
Ju lutem shkruani njerën nga kërkesat :PALINDROME takova duket
Teksti i dhënë është palindrome
Vazhdoni me kerkese tjeter ose shtyp 0 per dalje.

Ju lutem shkruani njerën nga kërkesat :PALINDROME Projekti i pare nga rrjeta kompjuterike
Teksti i dhënë nuk është palindrome
Vazhdoni me kerkese tjeter ose shtyp 0 per dalje.

Ju lutem shkruani njerën nga kërkesat :PALINDROME
Ju lutem shenoni nje fjalë pas kërkesës PALINDROME
Vazhdoni me kerkese tjeter ose shtyp 0 per dalje.
```

5.6. Metoda TIME

Kjo metode përcakton kohën aktuale në server dhe e dërgon atë tek klienti si format të lexueshme për njerëzit.

Përmes librarisë strfTime merret koha aktuale e sistemit dhe kthehet në string në formatin “%d.%m.%Y %I:%M:%S:%p”

- Realizimi përmes kodit:

```
elif klienti[0] == 'TIME':  
    if klienti[1:]:  
        pergjigjja = "Nuk keni nevoj te shkruani asgje pas kërkesës TIME! "  
    else:  
        koha = datetime.now().strftime('%d.%m.%Y %I:%M:%S:%p')  
        pergjigjja = koha
```

- Rezultati pas ekzekutimit të kësaj kërkesë :

```
Ju lutem shkruani njerën nga kërkesat :TIME  
10.04.2020 04:43:08:PM  
Vazhdoni me kerkese tjeter ose shtyp 0 per dalje.
```

```
Ju lutem shkruani njerën nga kërkesat :TIME 1  
Nuk keni nevoj te shkruani asgje pas kërkesës TIME!  
Vazhdoni me kerkese tjeter ose shtyp 0 per dalje.
```

5.7. Metoda GAME

Më anë të metodes GAME gjënerohën automatikisht 5 numra të rastesishëm dhe dërgohën tek klienti.

- Realizimi përmes kodit:

```
elif klienti[0].upper() == 'GAME':  
    if klienti[1:]:  
        pergjigjja = "Game i merr rastesisht numrat nuk keni nevoj te shkruani asgje! "  
    else:  
        game = [random.randint(1, 35) for i in range(5)]  
        game.sort()  
        pergjigjja = "5 numra te rastesishem nga rangu 1-35: " + str(game)
```

Se pari është bërë importimi I librarisë random dhe është bërë krijimin e nje vargu me 5 numra me vlera prej 1-35 pastaj me ane të metodes sort() është bërë sortimi I atij vargu.

- Rezultati pas ekzekutimit të kësaj kërkesë :

```
Ju lutem shkruani njerën nga kërkesat :GAME  
5 numra te rastesishem nga rangu 1-35: [2, 3, 10, 22, 29]  
Vazhdoni me kerkese tjeter ose shtyp 0 per dalje.  
  
Ju lutem shkruani njerën nga kërkesat :GAME 1  
Game i merr rastesisht numrat nuk keni nevoj te shkruani asgje!  
Vazhdoni me kerkese tjeter ose shtyp 0 per dalje.
```

5.8. Metoda GCF

Më anë të metodes GCF gjejmë faktorin më të madh të përbashkët në mes dy numra.

- Realizimi përmes kodit

```
def gcf(x, y):  
    while y:  
        x, y = y, x % y  
    return x  
  
elif klienti[0].upper() == 'GCF':  
    try:  
        num1 = int(klienti[1])  
        num2 = int(klienti[2])  
        if klienti[3]:  
            pergjigjia = "Nuk mund të shkruani me shumë se dy numra pas kërkesës"  
        else:  
            num = gcf(num1, num2)  
            pergjigjia = "GCF: " + str(num)  
    except:  
        pergjigjia = "Ju lutem shënoni dy numra të plote pas kërkesës GCF"
```

Krijimi i kësaj metode është realizuar në bazë të funksionit gcf i cili me anë të llogaritjeve matematikore përcakton faktorin më të madh, pastaj me thirrjen e kësaj metode gjejmë GCF-në e dy numrave të plote të cilët duhet të shenohën pas kërkesës në të kundërtën shfaqet teksti për informim.

- Rezultati pas ekzekutimit të kësaj kërkesë:

```
Ju lutem shkruani njerën nga kërkesat :GCF 10 20  
GCF: 10  
Vazhdoni me kërkesë tjetër ose shtyp 0 për dalje.  
  
Ju lutem shkruani njerën nga kërkesat :GCF  
Ju lutem shënoni dy numra të plote pas kërkesës GCF  
Vazhdoni me kërkesë tjetër ose shtyp 0 për dalje.  
  
Ju lutem shkruani njerën nga kërkesat :GCF 10 20 30  
Nuk mund të shkruani me shumë se dy numra pas kërkesës  
Vazhdoni me kërkesë tjetër ose shtyp 0 për dalje.
```

5.9. Metoda CONVERT

Metode më anë të së cilës mundësohet konvertimi i një njesie në njësi tjetër. Lista e parametrave njësive që shëndrrojnë numrat në njësi tjetër janë:

- cmToFeet
- FeetToCm
- kmToMiles
- MileToKm

Këto shëndrime janë mundësuar më anë të një funksionit konvertimi(s,num) i cili i paraqet llogaritjet matematikore ku s- tregon së çka në çka duhet të shëndrrohet, kurse num paraqet numrin i cili do të konvertohet.

- Realizimi përmes kodit

```

elif klienti[0].upper() == 'CONVERT':
    try:
        opsi = klienti[1]
        num1 = float(klienti[2].upper())
        ops = klienti[3:]
        if ops:
            pergjigjja = "Duhet ta shenoni vetem qka deshironi te konvertoni dhe numrin pas kerkeses! \n "
        else:
            num = "%.2f" % (konvertimi(opsi, num1))
            pergjigjja = num
    except:
        pergjigjja = "Ju lutem shenoni kerkesen ne rregull \n"

def konvertimi(s, num):
    if s == "CMTOFEET":
        return num / 30.48
    elif s == "FEETTOCM":
        return num * 30.48
    elif s == "KMTOMILES":
        return num / 1.609344
    elif s == "MILESTOCH":
        return num * 1.609344

```

Klienti pasi ta thirr këtë metode permës fjalës {CONVERT} ,duhet ta shkruaj opsionin për shëndrrim dhe me pas vlerën që dëshiron ta shëndrroj.Nese klienti nuk shkruan kërkesën në rregull atëhëre server e informon për këtë

➤ Rezultati pas ekzekutimit te kesaj kerkese:

```

Ju lutem shkruani njerën nga kërkesat :CONVERT CMTOFEET 500
16.40
Vazhdoni me kerkese tjeter ose shtyp 0 per dalje.

Ju lutem shkruani njerën nga kërkesat :convert FEETTOCM 16.40
499.87
Vazhdoni me kerkese tjeter ose shtyp 0 per dalje.

Ju lutem shkruani njerën nga kërkesat :convert KMTOMILES 100
62.14
Vazhdoni me kerkese tjeter ose shtyp 0 per dalje.

Ju lutem shkruani njerën nga kërkesat :Convert MILESTOCH 100
160.93

```

6.METODAT SHITES

6.1.Metoda LETTERS

Metoda LETTERS është një metode që shërben për gjetjen e numrit të shkronjave të njëjta dhe të ndryshme të përdorura në dy fjalë.

- Realizimi përmes kodit

```
elif klienti[0].upper() == "LETTERS":
    try:
        if klienti[3:]:
            pergjigjja = "Ju lutem shkruani kërkesën në rregull duhet dhënë vetëm dy fjale pas kërkesës."
        else:
            teksti1 = klienti[1].lower()
            teksti2 = klienti[2].lower()
            shkronjatenjejta = Counter(teksti1) & Counter(teksti2)
            shkronjatendryshme = Counter(teksti1) - Counter(teksti2)
            a = sum(shkronjatenjejta.values())
            b = sum(shkronjatendryshme.values())
            pergjigjja = "Numri i shkronjave të njëjta është [" + str(a) + "] dhe caktimi i tyre me " + \
                str(shkronjatenjejta) + "\nNumri i shkronjave të ndryshme është [" + str(b) + "] dhe caktimi i tyre me " + str(shkronjatendryshme)
    except:
        pergjigjja = "Ju lutem shkruani kërkesën në rregull duhet dhënë dy fjale pas kërkesës"
```

Pasi klienti ta këtë shënuar kërkesën duhet të shkruaj dy fjalë pas saj ,ku në këtë kod më ane të përdorimit të librarise Counter mundesohet përdorimi i funksionit Counter i cili bënë krahasimin e dy fjalëve të parashtruara si kërkesë nga klienti. Si bazë krahasuese marrët fjala e parë (pra fjala që gjendet në antarin e dytë të kërkesës,pra kërkesa klienti[1]) dhe pas ekzekutimit të kodit si rezultat fitojmë numrin e shkronjave të njëjta që gjenden në të dy fjalët dhe numrin e shkronjave të ndryshme që gjenden të fjalë e parë por jo tek e dyta.Po ashtu në rezultat kam paraqit se sa hërë është përdorur shkronja përkatëse(e njëjte apo e ndryshme)

- Rezultati pas ekzekutimit të kësaj kërkesë:

```
Ju lutem shkruani njerën nga kërkesat :LETTERS Inshirleri kompjuterike
Numri i shkronjave të njëjta është [3] dhe caktimi i tyre me Counter({'i': 1, 'e': 1, 'r': 1})
Numri i shkronjave të ndryshme është [7] dhe caktimi i tyre me Counter({'i': 3, 'n': 2, 'x': 1, 'h': 1})
Vazhdoni me kërkesë tjetër ose shtyp 0 për dalje.

Ju lutem shkruani njerën nga kërkesat :LETTERS rrjeta
Ju lutem shkruani kërkesën në rregull duhet dhënë dy fjale pas kërkesës
Vazhdoni me kërkesë tjetër ose shtyp 0 për dalje.

Ju lutem shkruani njerën nga kërkesat :LETTERS
Ju lutem shkruani kërkesën në rregull duhet dhënë dy fjale pas kërkesës
```

6.2. Metoda PËRQINDJA

Metode e cila bazohet në llogaritje matematikore për të caktuar ndryshimin e një vlere fillestare pasi ajo të zbritet më një vlere të caktuar të përqindjes.

- Realizimi përmes kodit

```
elif klienti[0] == 'PËRQINDJA':
    try:
        num1 = float(klienti[1])
        num2 = float(klienti[2])
        if klienti[3]:
            pergjigjja = "Nuk mund të shkruani me shumë se dy numra pas kërkesës"
        else:
            num=num1*num2
            num3=num/100
            num4="%.2f" %(num1-num3)
            pergjigjja="Qmimi ka qene "+str(num1) + " dhe pas zbritjes "+str(num2)+"% qmimi eshte bere "+str(num4)
    except:
        pergjigjja = "Ju lutem shënoni dy numra pas kërkesës "
```

Parametri i parë i kërkesës është PËRQINDJA, parametri i dytë paraqet vlerën fillestare (që në këtë rast është cmimi), kurse i treti paraqet përqindjen (për sa dëshirojmë më ndryshu vlerën fillestare). Klienti mund t'i paraqet kërkesat si vlera float varësisht prej llogaritjeve që atij i nevojiten. Vlerën përfundimtare më ane të “%.2f” serveri e paraqet si vlerë me dy numra pas pikës dhjetore.

- Rezultati pas ekzekutimit të kësaj kërkesë:

```
Ju lutem shkruani njerën nga kërkesat :perqindja 199.5 20
Qmimi ka qene 199.5 dhe pas zbritjes 20.0% qmimi eshte bere 159.60
Vazhdoni me kerkese tjeter ose shtyp 0 per dalje.

Ju lutem shkruani njerën nga kërkesat :perqindja 100
Ju lutem shënoni dy numra pas kërkesës
Vazhdoni me kerkese tjeter ose shtyp 0 per dalje.
```

Përmbledhje e rezultatit të testimit

Me ane të paraqitjes së ekzekutimit të FIEK Protokollit mund të themi se janë përmbushur të gjitha kërkesat. Dhe server është në gjendje të pranojë një sekuenca të kërkesave nga i njëjti klient apo klient të ndryshëm.

Testimi I Serverit më më shumë së një Klient

```
S | PERQINDJA *
*****
Du lutem shkruani njerën nga kërkesat :ipaddress
IP adresa e klientit është :127.0.0.1
Vazhdoni me kerkese tjeter ose shtyp 0 per dalje.

Du lutem shkruani njerën nga kërkesat :port
Klienti është duke përdorur portin 60709
Vazhdoni me kerkese tjeter ose shtyp 0 per dalje.

Du lutem shkruani njerën nga kërkesat :game
5 numra te rastesishem nga rangu 1-35: [1, 11, 15, 20, 35]
Vazhdoni me kerkese tjeter ose shtyp 0 per dalje.

Du lutem shkruani njerën nga kërkesat :perqindja 100 20
Qmimi i kompjuteri ka qene 100.0 pas zbritjes 20.0% qmimi eshte bere 80.00
Vazhdoni me kerkese tjeter ose shtyp 0 per dalje.

Du lutem shkruani njerën nga kërkesat :ipaddress
IP adresa e klientit është :127.0.0.1
Vazhdoni me kerkese tjeter ose shtyp 0 per dalje.

Du lutem shkruani njerën nga kërkesat :port
Klienti është duke përdorur portin 60709
Vazhdoni me kerkese tjeter ose shtyp 0 per dalje.

Du lutem shkruani njerën nga kërkesat :game
5 numra te rastesishem nga rangu 1-35: [14, 16, 23, 27, 34]
Vazhdoni me kerkese tjeter ose shtyp 0 per dalje.

Du lutem shkruani njerën nga kërkesat :time
10.04.2020 05:00 PM
Vazhdoni me kerkese tjeter ose shtyp 0 per dalje.

Vazhdoni me kerkese tjeter ose shtyp 0 per dalje.

Du lutem shkruani njerën nga kërkesat :ipaddress
IP adresa e klientit është :127.0.0.1
Vazhdoni me kerkese tjeter ose shtyp 0 per dalje.

Du lutem shkruani njerën nga kërkesat :game
5 numra te rastesishem nga rangu 1-35: [3, 5, 7, 26, 31]
Vazhdoni me kerkese tjeter ose shtyp 0 per dalje.

Du lutem shkruani njerën nga kërkesat :reverse rrjeta kompjuterike
Teksti i kthyer ne formen reverse është : EKIRETUJPMOK ATEJRR
Vazhdoni me kerkese tjeter ose shtyp 0 per dalje.

Du lutem shkruani njerën nga kërkesat :letters rrjeta kompjuterike
Numri i shkronjave te njejta eshte [4] dhe caktimi i tyre me Counter({'r':
1, 'j': 1, 'e': 1, 't': 1})
Numri i shkronjave te ndryshme eshte [2] dhe caktimi i tyre me Counter({'r':
1, 'a': 1})
Vazhdoni me kerkese tjeter ose shtyp 0 per dalje.

Du lutem shkruani njerën nga kërkesat :_

Serveri eshte i gatshem te pranoj kerkese
Klienti është lidhur me 127.0.0.1 ne portin 60709
Klienti është lidhur me 127.0.0.1 ne portin 60713
```

Activate Windows

Go to Settings to activate Windows.