

ФЕДЕРАЛЬНОЕ ГОСУДАРСТВЕННОЕ БЮДЖЕТНОЕ
ОБРАЗОВАТЕЛЬНОЕ УЧРЕЖДЕНИЕ ВЫСШЕГО ОБРАЗОВАНИЯ
"МОСКОВСКИЙ АВИАЦИОННЫЙ ИНСТИТУТ
(НАЦИОНАЛЬНЫЙ ИССЛЕДОВАТЕЛЬСКИЙ УНИВЕРСИТЕТ)"

Факультет: «Информационные технологии и прикладная математика»

Кафедра: 806 «Вычислительная математика и программирование»

Курсовой проект
по курсу «Операционные системы»

Группа: М8о-307Б-18

Студент: Тояков Артем Олегович

Преподаватель: Миронов Евгений Сергеевич

Оценка:

Дата:

Москва, 2019

Оглавление

1.Постановка задачи	2
2.Структура программы.....	3
3.Описание программы.....	3
4.Листинг программы	3
5.Результат работы	5
6.Вывод.....	6

1.Постановка задачи

«Необходимо написать 3-и программы. Далее будем обозначать эти программы А, В, С.

Программа А принимает из стандартного потока ввода строки, а далее их отправляет программе С.

Отправка строк должна производиться построчно.

Программа С печатает в стандартный вывод, полученную строку от программы А. После получения программа С отправляет программе А сообщение о том, что строка получена.

До тех пор пока программа А не примет «сообщение о получение строки» от программы С, она не может отправлять следующую строку программе С.

Программа В пишет в стандартный вывод количество отправленных символов программой А и количество принятых символов программой С.
Данную информацию программа В получает от программ А и С соответственно.

2. Структура программы

В данной работе моя программа состоит из трех файлов:

1. А.py
2. В.py
3. С.py

3. Описание программы

В данной работе связь между сервером и клиентом осуществляется через сокеты, также хотелось бы отметить, что используется библиотека через которую принимаются и отправляются данные. Программа С по сути является сервером, куда посылает сообщения клиент (программа А). Когда программа С печатает сообщение, она отправляет сигнал клиенту о том, что она получила сообщение и тогда можно посылать следующую строку. Программа В также является клиентом, однако всё спроектировано так, что она пишет только количество символов принятых сервером С и отправленных клиентом А.

4. Листинг программы

//А.py

```
import socket, threading, time

key = 8194

shutdown = False
join = False

def receiving (name, sock):
    while not shutdown:
        try:
            while True:
                data, addr = sock.recvfrom(1024)

                print(data.decode("utf-8"))

                time.sleep(0.2)
```

```

        except:
            pass
host = socket.gethostname(socket.gethostname())
port = 0

server = ("127.0.0.1",9090)

s = socket.socket(socket.AF_INET,socket.SOCK_DGRAM)
s.bind((host,port))
s.setblocking(0)

rT = threading.Thread(target = receving, args = ("RecvThread",s))
rT.start()

while shutdown == False:
    try:
        message = input()

        if message != "":
            s.sendto((message).encode("utf-8"),server)
            s.sendto(str(len((message))).encode("utf-8"),server)

        time.sleep(0.2)
    except:
        shutdown = True

rT.join()
s.close()

```

//B.py

```

import socket, threading, time

key = 8194

shutdown = False
join = False

def receving (name, sock):
    while not shutdown:
        try:
            while True:
                data, addr = sock.recvfrom(1024)

                print(data.decode("utf-8"))

                time.sleep(0.2)
            except:
                pass
host = socket.gethostname(socket.gethostname())
port = 0

server = ("127.0.0.1",9090)

s = socket.socket(socket.AF_INET,socket.SOCK_DGRAM)
s.bind((host,port))
s.setblocking(0)

rT = threading.Thread(target = receving, args = ("RecvThread",s))
rT.start()

while shutdown == False:
    try:
        message = input()

        if message != "":
            s.sendto((message).encode("utf-8"),server)

        time.sleep(0.2)
    except:
        shutdown = True

rT.join()
s.close()

```

//C.py

```

import socket, time

host = "127.0.0.1"

```

```

port = 9090

clients = []

s = socket.socket(socket.AF_INET, socket.SOCK_DGRAM)
s.bind((host, port)) #запустил сервер по ip и порту

quit = False
print("[ Server Started ]")
i = -1
j = 0
while not quit:
    try:
        data, addr = s.recvfrom(1024) #получил msg и unique num

        if addr not in clients: #если это новый клиент то добавляю адрес этого клиента
            clients.append(addr)
        if(j!= 0):
            if(i == 0):
                print(data.decode("utf-8"))
            j = j + 1
        for client in clients:
            if addr != client:
                if(i == 1):
                    res = str(recv) + " " + str(data.decode("utf-8"))
                    s.sendto(res.encode("utf-8"), client)
                elif (i == 0):
                    s.sendto(("I got the string").encode("utf-8"), client)
                    recv = len(data.decode("utf-8"))
                    i = i + 1
            else :
                i = 0

    except:
        print("\n[ Server Stopped ]")
        quit = True

s.close()

```

5.Результат работы

artoy@artoy:~/Desktop/Labs/3 sem/OS/Kp\$ python3 [A.py](#)

Hello world!

I got the string

Life is strange...

I got the string

artoy@artoy:~/Desktop/Labs/3 sem/OS/Kp\$ python3 [C.py](#)

[Server Started]

Hello world!

Life is strange...

artoy@artoy:~/Desktop/Labs/3 sem/OS/Kp\$ python3 [B.py](#)

ready

12 12

18 18

6.Вывод

При выполнении курсового проекта были изучены основы языка программирования Python. В ходе работы я также познакомился с такой технологией как сокеты. Мне было интересно изучить топологию клиент-сервер на питоне и понять её реализацию. Работа была простая и увлекательная, но к сожалению не претендует на оценку выше тройки.