
Question 1:

Government of India has decided to give the unique id for the Indian citizen. Therefore, it has formed a high-level committee for technical review and approval. The committee has released a tender for the project. Various companies have sent the applications for the tender with their cost estimation. The applications are received in a queue. Sort the applications based on their cost estimation and given ranking. Every client is able to receiving status about ranking. Write a java program for the above scenario using TCP.

Solution:

SERVER CODE: (Government Side)

```
import socket
port=60008
s=socket.socket()
host=socket.gethostname()
s.bind((host,port))
s.listen(5)
print('Government is listening...')
while(True):
    conn,addr=s.accept()
    print('Got connection from:',conn,addr)

    costE=conn.recv(1024).decode()
    compN=conn.recv(1024).decode()

    #received costs array from client side
    #received company number array from client side

    #sorting according to cost estimation in ascending order

    for i in range(n-1):
        for j in range(0, n-i-1):
            #Swap if the element found is greater than the next element
            if costE[j] > costE[j+1] :
                costE[j], costE[j+1] = costE[j+1], costE[j]

                compN[j], compN[j+1] = compN[j+1], compN[j]

            #swapping company numbers in accordance to cost estimation (ascending order)

    rank_string= pickle.dumps(compN)
    #to send the company number queue to the server
```

```
s.send(rank_string.encode())
print('Sent Ranking...')
conn.close()
```

Shaunak_Sensarma\Desktop\Programming - Geany

File Document Project Build Tools Help

```

1  import socket
2  port=60008
3  s=socket.socket()
4  host=socket.gethostname()
5  s.bind((host,port))
6  s.listen(5)
7  print('Government is listening...')
8  while(True):
9      conn,addr=s.accept()
10     print('Got connection from:',conn,addr)
11
12     costE=conn.recv(1024).decode()    #received costs array from client side
13     compN=conn.recv(1024).decode()    #received company number array from client side
14
15     #sorting according to cost estimation in ascending order
16
17     for i in range(n-1):
18         for j in range(0, n-i-1):
19             # Swap if the element found is greater than the next element
20             if costE[j] > costE[j+1] :
21                 costE[j], costE[j+1] = costE[j+1], costE[j]
22
23                 compN[j], compN[j+1] = compN[j+1], compN[j]    #swapping company numbers in
24                                                                #accordance to cost estimation(ascending order)
25
26     rank_string= pickle.dumps(compN)    #to send the company number queue to the server
27     s.send(rank_string.encode())
28
29     conn.close()
30

```

CLIENT CODE (Companies)

```

import socket
s=socket.socket()
host=socket.gethostname()
port=60008

print("Enter the number of companies taking part of the tender..")
n=int(input())
print("Enter the cost estimation of the companies followed by company number...")
cost = []    #queue to store cost estimation
com= []    #queue to store company number
for i in range(0, n):    #loop to fetch cost and number

```

```

ele = int(input())
cost.append(ele)
num=int(input())
com.append(num)

s.connect((host,port))
cost_string = pickle.dumps(cost)           #to send the cost estimation queue to the server
s.send(cost_string.encode())
com_string= pickle.dumps(com)              #to send the company number queue to the server
s.send(com_string.encode())

rank=s.recv(1024).decode()                 #receiving the queue containing ranks

print("The company rankings are...")
for i in (rank):
    print(i, end = " ")                    #printing company ranks in ascending format
s.close

```

```

client5.py x server5.py x
1  import socket
2  s=socket.socket()
3  host=socket.gethostname()
4  port=60008
5
6  print("Enter the number of companies taking part of the tender..")
7  n=int(input())
8  print("Enter the cost estimation of the companies followed by company number...")
9  cost = []                                #queue to store cost estimation
10 com= []                                  #queue to store company number
11 for i in range(0, n):                    #loop to fetch cost and number
12     ele = int(input())
13     cost.append(ele)
14     num=int(input())
15     com.append(num)
16
17 s.connect((host,port))
18 cost_string = pickle.dumps(cost)          #to send the cost estimation queue to the server
19 s.send(cost_string.encode())
20 com_string= pickle.dumps(com)             #to send the company number queue to the server
21 s.send(com_string.encode())
22
23 rank=s.recv(1024).decode()                #receiving the queue containing ranks
24 print("The company rankings are: ")
25 for i in (rank):
26     print(i, end = " ")                  #printing company ranks in ascending format
27
28 s.close
29

```

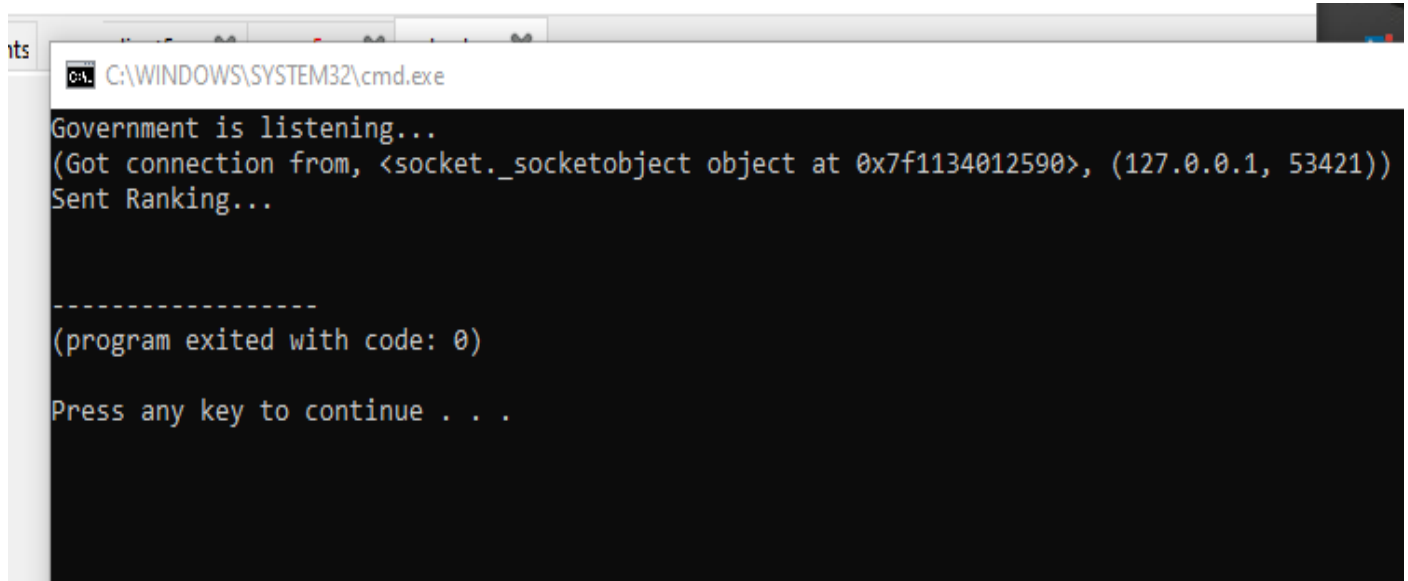
file "untitled" opened.

:\Users\Shaunak_Sensarma\Desktop\Programming\practice4.py closed.

:\Users\Shaunak_Sensarma\Desktop\Programming\client5.py saved.

Output:

Server/ Government Side:

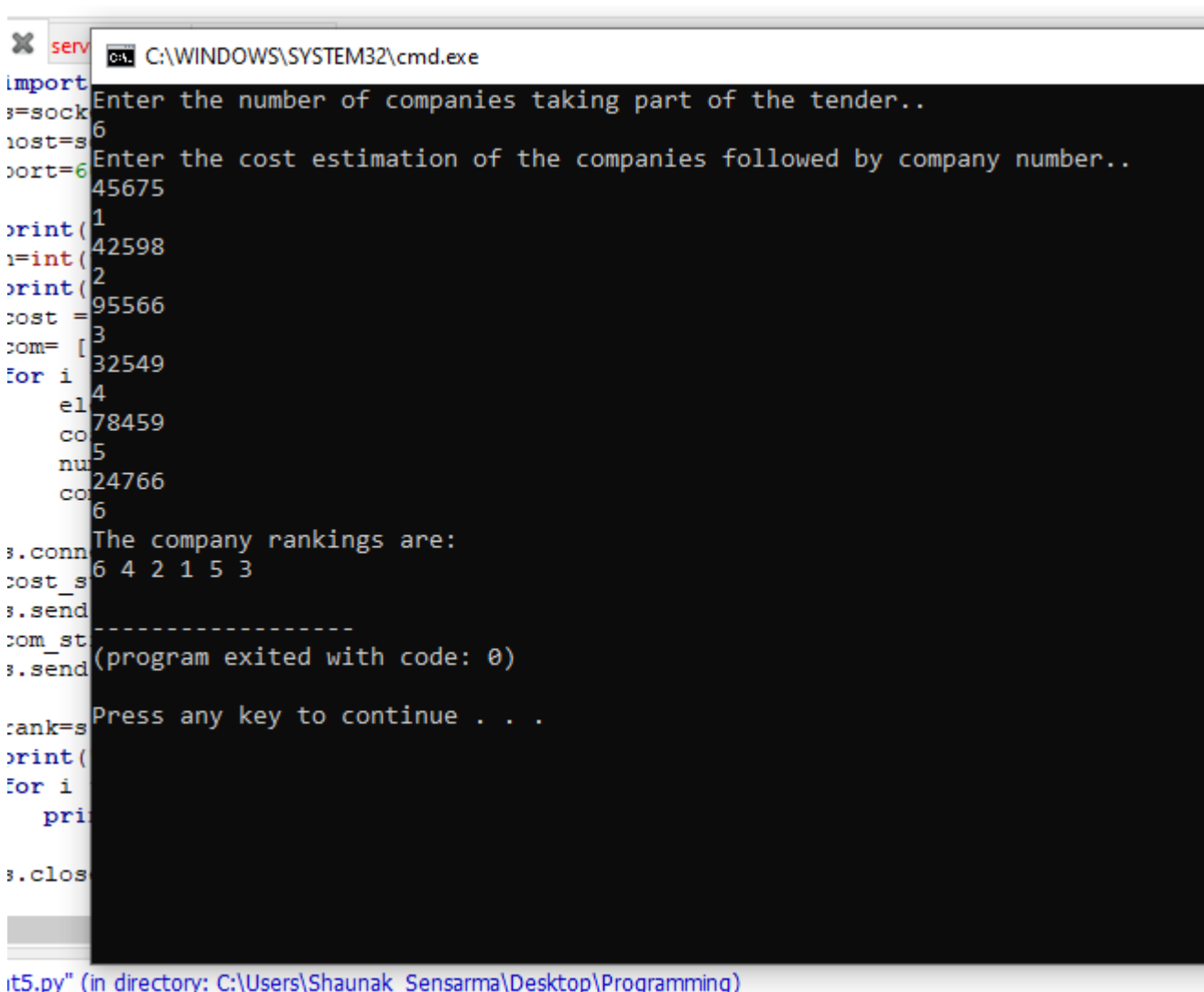


```
C:\WINDOWS\SYSTEM32\cmd.exe
Government is listening...
(Got connection from, <socket._socketobject object at 0x7f1134012590>, (127.0.0.1, 53421))
Sent Ranking...

-----
(program exited with code: 0)

Press any key to continue . . .
```

Client Side:



```
C:\WINDOWS\SYSTEM32\cmd.exe
import socket
s=socket.socket()
s.bind(('localhost', 53421))
s.listen(6)
print('Server is listening...')
while True:
    (conn, addr) = s.accept()
    data = conn.recv(1024)
    if not data:
        break
    data = data.decode('utf-8')
    data = data.split(',')
    num = int(data[0])
    cost = []
    for i in range(1, num+1):
        cost.append(int(data[i]))
    print('The company rankings are:')
    print(' '.join(str(i) for i in cost))
    conn.close()
    print('Press any key to continue . . .')
    input()
    print('Program ended successfully')
    print('it5.py" (in directory: C:\Users\Shaunak_Sensarma\Desktop\Programming)
```

QUESTION- 2:

Write a TCP/IP based client-server program to detect the error by using parity check in the codeword received at the server program. Create the codeword at the client program from the dataword by using the X-OR operation.

SERVER Code:

```
import socket
s=socket.socket()
host=socket.gethostname()
port=60005
s.bind((host,port))
s.listen(20)
print('Server is listening...')
while(True):
    c,addr=s.accept()
    print("Got connection from : ",addr)
    code=c.recv(1024).decode()
    n6=code[0]^code[2]^code[4]
    n5=code[0]^code[1]^code[4]
    n3=code[0]^code[1]^code[2]
    c=4*n3+2*n5+n6
    if(c==0):
        printf("\nNo error while transmission of data\n")
    else:
        printf("\nError on position %d",c);
    print('Connection over...')
    conn.close()
```

#XOR operation

rs\Shaunak_Sensarma\Desktop\Programming - Geany

File Document Project Build Tools Help



client5.py x server5.py x client6.py x server6.py x

```
1 import socket
2 s=socket.socket()
3 host=socket.gethostname()
4 port=60005
5 s.bind((host,port))
6 s.listen(20)
7 print('Server is listening...')
8 while(True):
9     c,addr=s.accept()
10    print("Got connection from : ",addr)
11    code=c.recv(1024).decode()
12    n6=code[0]^code[2]^code[4] #XOR operation
13    n5=code[0]^code[1]^code[4]
14    n3=code[0]^code[1]^code[2]
15    c=4*n3+2*n5+n6
16    if(c==0):
17        printf("\nNo error while transmission of data\n")
18    else:
19        printf("\nError on position %d",c);
20    print('Connection over...')
21    conn.close()
22
23
```

CLIENT Code:

```
import socket
```

```
s=socket.socket()
```

```
host=socket.gethostname()
```

```
port=60005
```

```
data= []
```

```
print('\nEnter the 4 bits of number..')
```

```
n0=int(input())
```

```
data.insert(0, n0)
```

```
n1=int(input())
```

```
data.insert(1, n1)
```

```
n2=int(input())
```

```
data.insert(2, n2)
```

```
n4=int(input())
```

```
data.insert(4, n4)
```

```
print('Dataword is....')
```

```
for i in (data):
```

```
    print (i,end = "")
```

```
n6=data[0]^data[2]^data[4]
```

#XOR operation

```
n5=data[0]^data[1]^data[4]
```

```
n3=data[0]^data[1]^data[2]
```

```
data.insert(6, n6)
```

```
data.insert(5, n5)
```

```
data.insert(3, n3)
```

```
print("\nCodeword is....')
```

```
for i in (data):
```

```
    print (i,end = "")
```

```
s.connect((host,port))
```

```
data_string = pickle.dumps(data)
```

#to send the codeword to the server

```
s.send(data_string.encode())
```

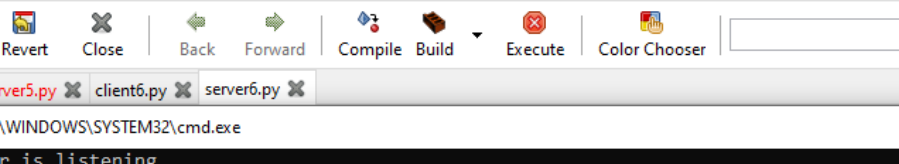
```
print("\nCodeword sent')
```

```
s.close
```

client5.py x server5.py x client6.py x

```
1 import socket
2 s=socket.socket()
3 host=socket.gethostname()
4 port=60005
5
6 data= []
7 print('\nEnter the 4 bits of number..')
8 n0=int(input())
9 data.insert(0, n0)
10 n1=int(input())
11 data.insert(1, n1)
12 n2=int(input())
13 data.insert(2, n2)
14 n4=int(input())
15 data.insert(4, n4)
16
17 print('Dataword is....')
18 for i in (data):
19     print (i,end = "")
20 n6=data[0]^data[2]^data[4]      #XOR operation
21 n5=data[0]^data[1]^data[4]
22 n3=data[0]^data[1]^data[2]
23 data.insert(6, n6)
24 data.insert(5, n5)
25 data.insert(3, n3)
26 print('\nCodeword is....')
27 for i in (data):
28     print (i,end = "")
29
30 s.connect((host,port))
31 data_string = pickle.dumps(data)      #to send the codeword to the server
32 s.send(data_string.encode())
33
34 print('\nCodeword sent')
35 s.close
```


Server Side:



k_Sensarma\Desktop\Programming - Geany

Document Project Build Tools Help

Save All Revert Close Back Forward Compile Build Execute Color Chooser Find

client5.py server5.py client6.py server6.py

```
1 C:\WINDOWS\SYSTEM32\cmd.exe
2
3 Server is listening...
4 (Got connection from,<socket._socketobject object at 0x6f1127023456>, (127.0.0.1,53241))
5 Error on position 3
6 Connection over...
7
8 -----
9 (program exited with code: 0)
10
11 Press any key to continue . . .
```

Client Side:

```
.insert(1, nl)
nt(C:\WINDOWS\SYSTEM32\cmd.exe
.in
nt(
.in Enter the 4 bits of number..
.in 1
.in 1
t(1
i 0
i 1
rin 1
Dataword is....
.in 1101
Codeword is....
.in 1100101
ata 1100101
ata Codeword sent

.pc
-----
ata (program exited with code: 0)
.in
.in Press any key to continue . . .
.in

t('
i i
rin

onn
a_s
end

t('
los
```