

UNIVERSITY EXAMINATIONS

1ST SEMESTER 2022/2023 ACADEMIC YEAR

THIRD YEAR EXAMINATION FOR THE DEGREE OF
BACHELOR OF SCIENCE (ICT)

BICT 312: NETWORK PROGRAMMING

STREAM: R

TIME: 2 HRS

DAY: WED [8.30AM – 10.30AM]

DATE: 21/12/2022

THIS QUESTION PAPER CONSISTS OF FIVE (5) PAGES

PLEASE DO NOT OPEN UNTIL THE INVIGILATOR SAYS SO.

SECTION A: (COMPULSORY) TOTAL MARKS FOR THIS SECTION IS 30.**QUESTION ONE**

(a) Discuss the primary socket API functions and methods in this module are:

(10 Marks)

- (i) socket()
- (ii) bind()
- (iii) listen()
- (iv) accept()
- (v) connect()
- (vi) connect_ex()
- (vii) send()
- (viii) recv()
- (ix) close()

(b) Discuss the following multi-connection server, python listening code as the multiconn-server.py. sets up the listening socket:

(5 marks)

```
import selectors
sel = selectors.DefaultSelector()
# ...
lsock = socket.socket(socket.AF_INET, socket.SOCK_STREAM)
lsock.bind((host, port))
lsock.listen()
print('listening on', (host, port))
lsock.setblocking(False)
sel.register(lsock, selectors.EVENT_READ, data=None)
```

(c) Below is the server and the client output when listening on the loopback interface on port 65432: Highlight the interconnection of the two as they create an intercommunication process between the server and the client.

(10 Marks)

```
$ ./multiconn-server.py 127.0.0.1 65432
listening on ('127.0.0.1', 65432)
accepted connection from ('127.0.0.1', 61354)
accepted connection from ('127.0.0.1', 61355)
echoing b'Message 1 from client.Message 2 from client.' to ('127.0.0.1', 61354)
echoing b'Message 1 from client.Message 2 from client.' to ('127.0.0.1', 61355)
closing connection to ('127.0.0.1', 61354)
closing connection to ('127.0.0.1', 61355)
```

Below is the client output when it creates two connections to the server above:

```
$ ./multiconn-client.py 127.0.0.1 65432 2
starting connection 1 to ('127.0.0.1', 65432)
starting connection 2 to ('127.0.0.1', 65432)
sending b'Message 1 from client.' to connection 1
sending b'Message 2 from client.' to connection 1
sending b'Message 1 from client.' to connection 2
sending b'Message 2 from client.' to connection 2
received b'Message 1 from client.Message 2 from client.' from connection 1
closing connection 1
received b'Message 1 from client.Message 2 from client.' from connection 2
closing connection 2
```

- (d) What is the socket interface all about? What are the advantages of using this interface?
(5marks)

QUESTION TWO

- (a) Briefly explain the following socket primitives commands as used in Network client server socket Programming
(10 marks)

- (i) `int listen(int sockfd, int backlog);`
- (ii) `int accept(int sockfd, void *addr, int *addrlen);`
- (iii) `int connect(int sockfd, struct sockaddr *serv_addr, int addrlen);` //used by TCP client
- (iv) `int send(int sockfd, const void *msg, int len, int flags);`
- (v) `int recv(int sockfd, void *buf, int len, unsigned int flags);`
- (vi) (DGRAM-style): `int sendto(int sockfd, const void *msg, int len, int flags, const struct sockaddr *to, int tolen);`
- (vii) (DGRAM-style): `int recvfrom(int sockfd, void *buf, int len, unsigned int flags, struct sockaddr *from, int *fromlen);`

- (b) Briefly explain the Features of TCP that are relevant to the network programmer
(6 marks)

- (c) What is the socket interface all about? What are the advantages of using this interface?
(4 Marks)

QUESTION THREE

(a) What is a socket? (2 mark)

(b) File I/O and socket I/O are exactly similar from the programmer perspective
(4 marks)

(c) Write a PYTHON PROGRAM USING TCP SOCKETS CLIENT SERVER CONNECTION

AIM: To implement date and time display from client to server using TCP Sockets

DESCRIPTION: TCP Server gets the system date and time and opens the server socket to read the client details. Client send its address to the server. Then client receives the date and time from server to display. TCP socket server client connection is opened for communication. After the date time is displayed the server client connection is closed with its respective streams to be closed. (10 Marks)

(d) In a nutshell explain the following TCP State transition (6 marks)

- i. TIMED-WAIT
- ii. QUIET TIME
- iii. FIN_WAIT2
- iv. RESET SEGMENTS
- v. SIMULTANEOUS OPEN/CLOSE
- vi. MAXIMUM SEGMENT SIZE

QUESTION FOUR

(a) Which application at webserver must process this packet? In TCP/IP, each well-known application is identified using ports. The port of DNS is **53**; HTTP is **80**; SMTP is **25**. In our considered example, HTTP server application (port 80) would process the packet. Packet so far: Source Port | Destination Port (6 Marks)

(b) Discuss the following characteristics as used in TCP (6 marks)

- Connection-oriented
- No predefined blocksize
- Processes identified by port numbers
- Services live at specific ports

– TCP/IP provides end-to-end-connectivity specifying how data should be

- (i) Formatted,
- (ii) Addressed,
- (iii) Transmitted,
- (iv) Routed, and
- (v) Received at the destination

(5 marks)

(d) What is a connectionless protocol? Give an example.

(3 marks)

QUESTION FIVE

(a) Briefly Explain the Components of the Socket API below

(8 marks)

- (i) Thread
- (ii) Inter-Process Communication
- (iii) Inter-Host Communication
- (iv) Communication Protocol

(b) User Datagram Protocol (UDP) is not connection-based (connectionless service with datagram sockets Explain the Term Connectionless

(2 marks)

(c) Compare the headers of IPv4 and IPv6. Which are the main differences? How are extensions implemented in IPv6?

(4 marks)

(d) Multicast applications and intended uses in the Internet. How does the communication based on multicast differ from the typical communication model of the point-to-point communication?

(3 marks)

(e) Identify the disadvantages of fragmenting data. Discuss two of these disadvantages

(3 marks)

