**Computer Networks – Advanced Course**

Exercise 2 - Socket programming SMTP client & server

**Implementing SMTP client and server:**

Develop a client server system which reads user input, sends to the server, receives server response and transmits back to the client. Everything should be done according to SMTP protocol as recorded in Wireshark record.

Special note – use only socket and base64 libraries. Using SMTP libraries is forbidden.

Before beginning, read:

* Chap. 2 in the book "רשתות מחשבים". (in Hebrew) or
* Chap. 2.7 in the book “Computer networking, a Top-Down approach” (in English) (available also in Moodle).

**The client side:**

1. The client should establish connection with the server using the SMTP messages (“EHLO” message etc...).
2. Make sure to base64 encode those messages which are required in the protocol.
3. Make sure to write an email with the required fields and the string that indicates email end.

**The server side:**

1. The server should respond to the client connection establishment with the proper SMTP responses.
2. The server shall include known combination of username and password, which the client can use to connect.
3. It the client does not comply with SMTP protocol, or the username/password is not correct, the server shall use SMTP error messages. **Use RFC 2821 for details**.
4. The server shall print all incoming / outgoing messages, to make it easier for the course staff to check.
5. The server side is not required to work with multiple clients at the same time.

The protocol

1. All protocol constants should be defined. Error codes and port.

**Good programming:**

1. Consider all of the extreme cases. The server should keep working no matter what the client sends.
2. The server side has functions that check if the command is good.
3. The server’s functions are checked using asserts.
4. Proper documentation and docstrings.
5. Program using PEP8 guidelines.

**Skeleton files:**

Use the provided skeleton files, they will help you make sure your work complies with the requirements.

**Examples of base64 encoding / decoding:**

import base64

my\_string = "Hello, World!"

encoded\_string = base64.b64encode(my\_string.encode())

print(encoded\_string)

b'SGVsbG8sIFdvcmxkIQ=='

decoded\_string =base64.b64decode(encoded\_string.decode())  
print(decoded\_string)

Good luck!