Network Programming

Assignment 1

Submission Instruction:

==================

- Due date: March 31, 2022
- File format: Compress the following files in a zip format (1) Python codes (file format: py) including comments, (2) Execution results (file format: pdf).
- Python file naming: {student ID}_{assignment #}_{question #}.py (e.g., 123456789_1_2.py for assignment #1 question #2).
- There will be a deduction of points (up to 50%) if you do not follow the above instruction.

Evaluation Criteria:

==================

- Your codes will be cross-checked with others to detect copying, and you will get zero points under such circumstances.
- Your codes will be evaluated based on programming style (organization, comments, readability), program function (completeness), documentation (demonstration, exception handling), etc.

Inquiries:

==================

- Send inquiries to the TA: Ankit Kumar Singh (ankit@soongsil.ac.kr).

Questions:

==================

[Total: 10 points]

1. [3 points] Write a Python script for demonstrating "Raw Network Conversation." Use socket library to request Open Notify API to GET an estimate for when the ISS will fly over a specified point.

parameters = {'lat':'45', 'lon':'180'}
base_address = http://api.open-notify.org/iss-pass.json

Hint: Take a reference from Chapter 1 Listing 1-4 (search.py)
Note: Don't use any other library(such as requests, http.client, etc.) for an API request.

2. [3 points] Write a python script to create a UDP Server and Client using python socket standard library.

The server can check the received data size:

```
 if the received data size is even in number,
                reply with received data size,
 else
         reply with "Error 403 Forbidden."
```

Example:
Suppose the client sent "Soongsil University" as data to the server. then the size of "Soongsil University" is 19 bytes long, so 19 is an odd number; therefore, it received a reply from the server: "Error 403 Forbidden."

3. [4 points] Write a Python script to create a TCP Server and Client using python socket standard library to develop a game "Guess the number."

The game will follow as :
1) The server will choose a random number between 1 to 10.
2) Inform the client and ask to guess a number between 1 to 10.
3) Client will send a guessed number as a request to the server.

Conditions:
1) Client will request to start the game by sending the first request as "start."
2) Client will have only 5 attempts to guess the correct number
3) The client will only win if he guesses the correct number within 5 attempts and loses the game.

The exchange of messages between server and client during the game will follow the following conditions and message text based on the difference of actual number with server and guessed number by client:

x = random chosen number by server
guess = number guessed by a client

conditions and messages:

(x = guess) -> "Congratulations you did it."

(x > guess) -> "You guessed too small!"

(x < guess) -> "You Guessed too high!"