**Vantage Circle | Assignment for backend developer**

**Part 1: Communication between the client and the server over request-response**

**Solution:**

The client-server architecture represents the relationship between the client/s and the server which generally exists in a common network. Basically the server-host runs server program/s which enables it to share its resources or services which the clients requests for.

Request-response is one of the basic methods by which computers/nodes are able to communicate with other computers/nodes. Here the initial computers/nodes generate a request and sends it to the target computer/nodes, then the other responds to the request. It is similar to a telephone call process where the caller waits for the other person to respond to the call.

The communication in client-server over request-response is generally purely synchronous but could be asynchronous too where the response to the request is returned later at an unknown time.

The server shares its resources or services with its clients who produced the requests, but the client does not share its resources except the data that is sent along with the request-response event. Generally there is a single server connected to multiple clients.

The first communication between the client and server (or between any other nodes) occurs with a handshake where both parties go through multiple steps of validation before actual communication takes place.

**Part 2: Finding two numbers from an array that sums up to a value ‘n’**

**Solution:**

**Algorithm:**

* 1. Initialize the array (arr) with predefined or user-defined array length (len)
  2. Input the values for each array element.
  3. Input sum value (val) for finding the pairs that add up to sum value
  4. Initialize empty HashSet (set) [in Java]
  5. Repeat following steps for all elements in arr[]:
     1. temp = val – arr[i]
     2. IF set contains temp

THEN print temp and arr[i]

ELSE add arr[i] to set

**Big O Notation:**

* **Time complexity: O(n)**
  + O(n) since the array is traversed only once.
  + From the algorithm:
    - Step 1: O(1+1)
    - Step 2: O(n)
    - Step 3: O(1)
    - Step 4: O(1)
    - Step 5: O(n)
  + Adding up the Big O for each operations [O(1+1+n+1+1+n)] and finding the highest order term we get time complexity **O(n).**
  + Time complexity of an algorithm or any operation is basically the time required to complete the execution of the algorithm or the operation according to the number of steps in it. It is also known as worst-case time complexity.
  + Big O notation is basically the metric for calculating complexities. In our case O(n) signifies linear time complexity which means that the number of operations increases linearly with the size of the input.
* **Space Complexity: O(n)**
  + O(n) since HashSet is utilized for storing array elements.
  + From the algorithm:
    - Step 1: O(1)
    - Step 2: O(n)
    - Step 3: O(1)
    - Step 4: O(1)
    - Step 5: O(n)
  + Space complexity = Auxiliary Space + Input Space
  + Space complexity is the total amount of memory used by an algorithm to execute and produce the results.
  + In this case it is O(n) because the algorithm shows linear space complexity.

The source code for the above problem is provided in the following GitHub link/path:

GitHub Link: <https://github.com/GYANANGKUSH/VantageCircleAssignment.git>

Path: [VantageCircleAssignment](https://github.com/GYANANGKUSH/VantageCircleAssignment)/[VantageCircleAssignment](https://github.com/GYANANGKUSH/VantageCircleAssignment/tree/master/VantageCircleAssignment)/[src](https://github.com/GYANANGKUSH/VantageCircleAssignment/tree/master/VantageCircleAssignment/src)/**part2**/

**Part 3: Program to find the cheapest hotel for a customer.**

**Solution:**

**The programs are made as per the provided additional guidelines.**

The program for the above solution has been provided in the GitHub link provided below:

[**https://github.com/GYANANGKUSH/VantageCircleAssignment.git**](https://github.com/GYANANGKUSH/VantageCircleAssignment.git)

The java files are present in the following path:

[VantageCircleAssignment](https://github.com/GYANANGKUSH/VantageCircleAssignment)/[VantageCircleAssignment](https://github.com/GYANANGKUSH/VantageCircleAssignment/tree/master/VantageCircleAssignment)/[src](https://github.com/GYANANGKUSH/VantageCircleAssignment/tree/master/VantageCircleAssignment/src)/**HotelBook**/

**THANK YOU**