Name:_____ **Class:**_____

JURONG PIONEER JUNIOR COLLEGE

JC2 Test 3 2025

COMPUTING **9569**
Higher 2 **July 2025**

Paper 2 (Lab-based)

1 hour 15 minutes

READ THESE INSTRUCTIONS FIRST

Answer **all** the questions.

All tasks must be done in the computer laboratory. You are not allowed to bring in or take out any pieces of work or materials on paper or electronic media or in any other form.

Approved calculators are allowed.

Save each task as it is completed.

The use of built-in functions, where appropriate, is allowed for this paper unless stated otherwise.

Note that **3** marks out of 40 may be awarded for the use of common coding standards for programming style.

The number of marks is given in brackets [ ] at the end of each task.

The total number of marks for this paper is **40**.

This document consists of **3** printed pages.

[Turn over

Instructions to candidates:

For each of the sub-tasks, add a comment statement at the beginning of the code using the hash symbol '#' to indicate the sub-task the program code belongs to.

1 You are developing a *guess-the-word* game with **socket programming**. The server program randomly selects a secret word from a predefined list `["apple", "banana", "orange", "mango", "durian", "grape"]` and the player (i.e.: client) tries to guess it.

- The player has a maximum of **three** tries to guess the correct word.
- After each incorrect guess, the server program sends the client program a hint: whether the guessed word is alphabetically before or after the correct word.
- The player views the hint.
- When the player guesses the correct word or uses all three tries, the server program sends the client program an appropriate message.
- The player views the message (i.e.: correct or out of tries).
- The game terminates.

The following is a sample run of the game on the client interface:

```
Welcome to Guess-The-Word! You have 3 tries.

Enter your guess: mango
Server: Incorrect. Hint: Your guess is alphabetically before the word.
Enter your guess: orange
Server: Correct! You guessed the word.
```

Task 1.1

An incomplete server code is given in `TASK1_SERVER.txt`. Use the code in the text file as a basis to write the **server** program. **[9]**

Name the file as `TASK1_SERVER_<your name>_<class>.py`

Task 1.2

Write the corresponding **client** program.

Name the file as `TASK1_CLIENT_<your name>_<class>.py` **[6]**

Task 1.3

Play the game where the client guesses correctly in the third try.

Save the screenshot of the client interface as `TASK1_<your name>_<class>.jpg` **[1]**

2 Design a simple web-based "Guess-the-Word" game where the user guesses the word `"CODERS"` using the given hint. The hint is `"Another word for PROGRAMMERS."` The application should consist of two web pages: a home page and a result page.

Home Page

- This page `index.html` should display:
  - The hint.
  - A form with a text input box for the user to enter his/her guess.
  - A submit button that sends the guess to the server using a POST request.

- After each guess, the server should:
  - Compare the input with the word `"CODERS"` (case-insensitive).
  - Track the number of attempts made by the user (up to **three** tries).
  - Redirect the user back to the home page unless the game has ended.

Result Page

- This page `result.html` should display the outcome of the game:
  - If the user guesses `"CODERS"` correctly, display a congratulatory message.
  - If the user exhausts all three attempts without guessing correctly, display a game over message along with the correct word.

Task 2.1

Write a Python program and the necessary files to create a web application for this game.

Save your Python program as `TASK2_<your name>_<class>.py`
with any additional files/ subfolders. **[20]**

Task 2.2

Run the application where the user exhausts all three attempts.

Save the web page output as `TASK2_<your name>_<class>.html` **[1]**

END OF PAPER