**Task: 5**

**Title: Time Complexity**

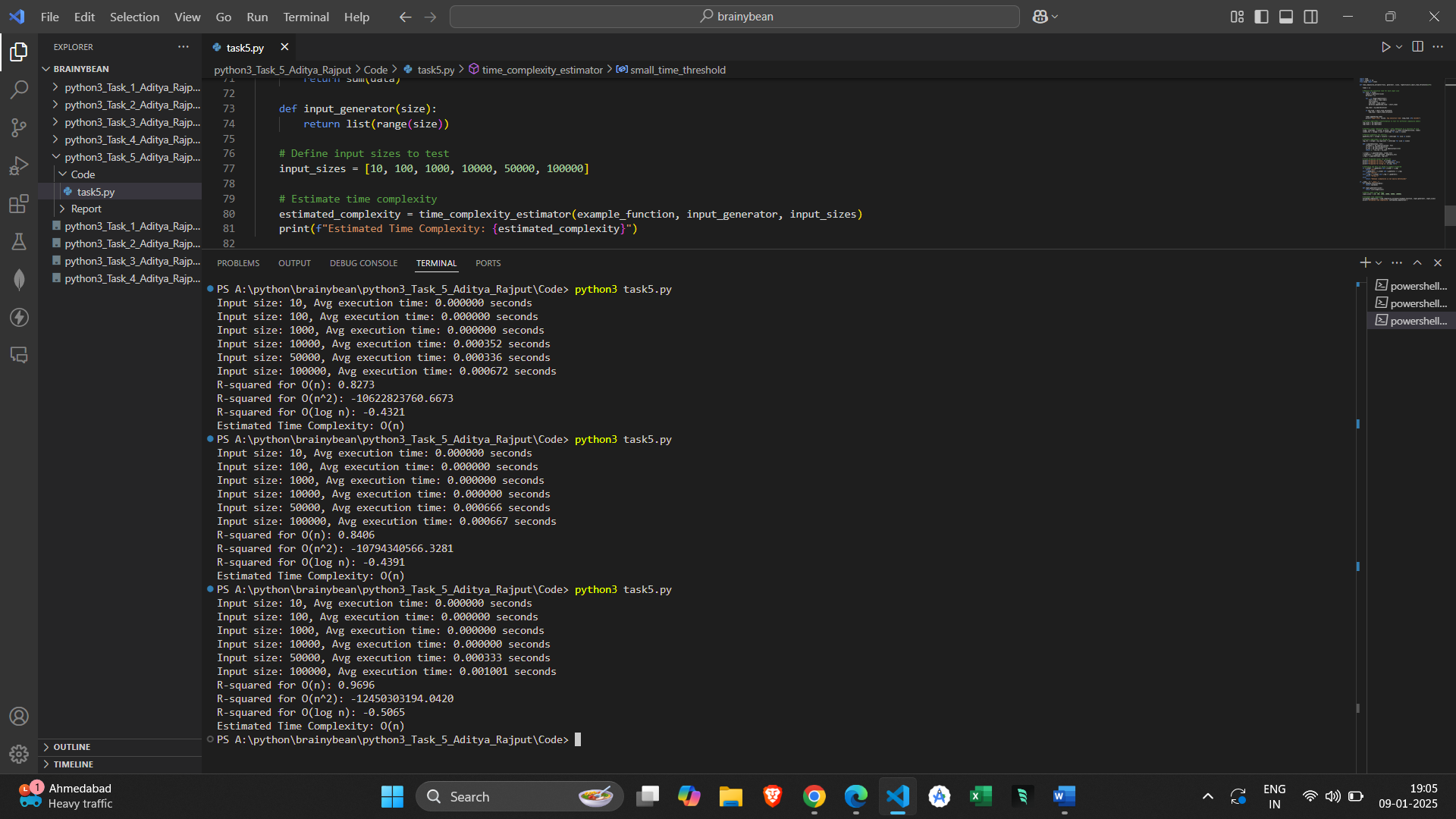
**Technology: python**

**1. Task Description:**

Build a code which can apply to any function and returns their time complexity.

I have to take reference from chatgpt because this was a complex query.

**2. Task Output:**

****

**3. Methods and Alorithms used in task:**

The code measures the actual execution time of a function for multiple input sizes.

It will call the function multiple times for each input size, recording the start and end time.

time.time() is used to record the current time before and after running the function.

np.mean() will calculate the average of the recorded times.

I have used logarithmic transformation of both input sizes and execution times helps in fitting different complexity models and many time complexities(o(n), o(n2), o(log n)) have relationships that can be represented linearly after applying a transformation.

I have used linear regression for model fitting like scipy.stats.linregress() to perform a linear regression.

I have also used to calculate R-Squared value.

R-Squared is used to quantify hoe well a particular model fits the data, a higher R-Squared value indicates a better fit.

The model with R-Squared value is assumed to be the most accurate in representing the time approximation of the time complexity of the function.

The final step will return the time complexity based on Big-O notation, if the linear model gives the best fit it returns “O(n)”, if the quadratic model then returns ‘’O(n^2)”, and if logarithmic model it will return “O(log n)”.

->**Time Measurement**: Record actual execution time for increasing input sizes.

-> **Logarithmic Transformation**: Transform data for better fitting to complexity models.

-> **Model Fitting (Linear, Quadratic, Logarithmic)**: Fit the data to various time complexity models.

-> **R-Squared Calculation**: Assess how well each model fits the data.

-> **Selection of Best Fit**: Compare R-squared values to choose the most appropriate time complexity model.

-> **Big-O Notation**: Return the time complexity in Big-O notation based on the best-fitting model.