Computer Programming-1 (BCSC 0071)

ASSIGNMENT 01

SHARED WITH SECTIONS EA AND EB

B.TECH 1ST YEAR CSE (HONS)

DATE OF ASSIGNMENT: 02 SEP 2024.

DATE OF ASSIGNMENT SUBMISSION: 03 SEP 2024

https://github.com/amirkhan1092/Batch2024-25

Problem 1.

Understanding Python and Its Interpretation

Scenario: Imagine you are tasked with developing a small automation script for your college project. Your team suggests using Python because of its simplicity and versatility.

Task: Explain what Python is and how it operates as an interpreted language. Discuss how Python's interpreted nature would benefit you in writing and testing your automation script.

Guidelines:

- Provide a brief overview of Python and it's features.
- Explain the concept of interpretation and how it differs from compilation.
- Discuss practical benefits of using an interpreted language like Python for development and testing.

Problem 2

Practical Applications of Python

Scenario: You have been appointed as a member of a project team that needs to select a programming language for different applications within a startup. Python has been recommended for various tasks.

Task: Identify four real-world applications of Python that would be useful for the startup. Provide a brief explanation of how Python can be applied in each of these areas.

Guidelines:

- Choose applications relevant to the startup's needs (e.g., web development, data analysis, automation).
- Explain how Python's features make it suitable for each application.
- Include real-world examples or case studies where possible.

Problem 3

Commenting in Python Code

Scenario: As you write a Python script for automating tasks in your project, you realize the importance of making your code understandable to other team members.

Task: Define what a comment is in Python programming. Then, demonstrate with examples how to write both single-line and multi-line comments within your code to clarify its functionality for your team.

Guidelines:

- Explain the purpose of comments in programming.
- Provide examples of single-line and multi-line comments.
- Illustrate how comments improve code readability and maintainability.

Problem 4

Exploring Python Keywords

Scenario: While coding, you encounter various reserved words that seem to have special meanings in Python. Your mentor asks you to learn more about these keywords.

Task: Python 3 has a set of reserved keywords. How many keywords are there in Python 3? Select any 10 keywords and provide a brief explanation of each, along with an example that shows how they are used in a practical scenario.

Guidelines:

- List the total number of keywords in Python 3.
- Choose 10 keywords and explain their role in Python programming.
- Provide practical code examples for each keyword to illustrate its usage.

Problem 5

Memory Management and Namespaces in Python

Scenario: You're developing a complex program in Python that handles large amounts of data. To ensure your program runs efficiently, you need to understand how Python manages memory and organizes variable names.

Task: Explain how memory is managed in Python and what a namespace is. Illustrate your explanation with a practical example of how Python handles variable names and memory allocation in a typical program.

Guidelines:

- Describe Python's memory management system, including garbage collection.
- Define what a namespace is and its types (local, global, built-in).
- Use code examples to show how Python manages variable names and memory.

Problem 6

Working with Standard Input and Output in Python

Scenario: As part of a project, you need to create an interactive Python script that collects user information and displays a personalized message based on the input. This involves using Python's standard input and output functions.

Task: Explain how the print() and input() functions are used in Python for standard output and input, respectively. Write a Python script that asks the user for their name and age, and then prints a message such as "Hello, [Name]! You are [Age] years old."

Guidelines:

- Explain the purpose of print() and input() functions in Python.
- Provide code examples demonstrating how to use these functions for user interaction.
- Ensure your script handles user input appropriately and formats the output clearly.

Assignment guidelines for students

- All Questions are compulsary to attend.
- Students are required to prepare assignments in their own handwriting.
- Students must maintain academic integrity and work on the assignments at their own. Direct copying of assignments from another student(s) is prohibited. If found guilty, zero marks will be awarded to the student(s).
- Last Date for Submission of Assignments is **Tuesday, Sep 03, 2024** in there respective lecture timings.
- Late submission of assignments will not be accepted in any case and the students who fail to submit the assignments will be awarded zero marks.
- General Guidelines:
- Paper Size must be A4 (Standard white plain paper)
- Assignment should contain page numbers (1, 2, 3….) starting from first question.
- Page number should be placed at the bottom-center of page.
- Students must mention all the details on the title page as specified

Name of Student: _____ Program Name and Section: _____

Assignment No.: _____ Class Roll No. of Student: _____

Subject: _____

Faculty: _____

Date of Allotment: _____

Date of submission: _____