CSC1024 Programming Principles

A Python-based Inventory Management System

Project Description:
The objective of this project is to develop an Inventory Management System using Python. The system should efficiently handle various aspects of inventory management, including product management, stock tracking, order processing, and reporting. The project aims to showcase the student's proficiency in applying fundamental programming concepts and techniques using Python.

Project Requirements:

1. Initial Data Preparation:
   - Create separate text files to store initial data for products, suppliers, and orders.
   - The "products.txt" file should contain product details such as product ID, name, description, and price.
   - The "suppliers.txt" file should store supplier information, including supplier ID, name, and contact details.
   - The "orders.txt" file should maintain records of orders, including order ID, product ID, quantity, and order date.

2. Program Functionality:
- Implement a user-friendly menu system with the following options:

a. Add a new product: Allow the user to add a new product to the inventory.
b. Update product details: Enable the user to update the details of an existing product.
c. Add a new supplier: Allow the user to add a new supplier to the system.
d. Place an order: Facilitate the process of placing an order for a specific product.
e. View inventory: Display the current inventory levels for all products.
f. Generate reports: Generate reports such as low stock items, product sales, and supplier orders.
g. Exit: Provide an option to exit the program gracefully.

3. Programming Techniques:
   a. Utilize input and display functions for effective user interaction and data input/output.
   b. Implement lists and file processing techniques to efficiently store and retrieve data from text files.
   c. Use conditional statements and logical operators to handle decision-making scenarios, such as checking book availability or membership status.
   d. Incorporate loops to perform repetitive tasks, such as displaying book inventory or member information.
   e. Create user-defined functions to modularize the code and enhance code reusability and readability.

Marking Rubrics:

1. Code Organization and Readability (8 marks):
    - Proper indentation and consistent coding style (2 marks)
    - Meaningful variable and function names (2 marks)
    - Appropriate use of comments to enhance code understanding (2 marks)
    - Logical and efficient code structure (2 marks)
2. Functionality and Correctness (30 marks):
    - Accurate implementation of product addition functionality (5 marks)
    - Correct product detail update process (5 marks)
    - Proper handling of supplier addition (4 marks)
    - Accurate order placement functionality (6 marks)
    - Correct display of inventory (5 marks)
    - Accurate generation of reports (5 marks)
3. Data Persistence and File Handling (15 marks):
    - Proper creation and utilization of text files for data storage (4 marks)
    - Accurate reading and writing of data to and from files (7 marks)
    - Efficient file handling techniques (4 marks)
4. User Interface and User Experience (12 marks):
    - User-friendly menu system (4 marks)
    - Clear and concise prompts and messages (4 marks)
    - Smooth navigation and flow of the program (4 marks)
5. Error Handling and Robustness (10 marks):
    - Appropriate error handling for input validation (4 marks)
    - Graceful handling of edge cases and exceptions (3 marks)
    - Overall program stability and reliability (3 marks)
6. Flowchart (5 marks):
    - Clarity and readability of the flowchart (1 mark)
    - Accurate representation of the program flow (2 marks)
    - Inclusion of all major processes and decision points (1 mark)
    - Proper use of flowchart symbols and conventions (1 mark)
7. Report (10 marks):
    - Clarity and organization of the report (2 marks)
    - Detailed explanation of the system design and architecture (2 marks)
    - Discussion of the implementation challenges and solutions (2 marks)
    - Analysis of the system's strengths and limitations (2 marks)
    - Suggestions for future improvements and enhancements (2 marks)
8. Presentation (10 marks):
    - Clarity and effectiveness of the presentation (2 marks)
    - Demonstration of the system's functionality (3 marks)
    - Explanation of the key features and design decisions (2 marks)
    - Ability to answer questions and provide insights (2 marks)
    - Time management and adherence to the allotted presentation duration (1 mark)

Total Marks: 100

Notes:

1. This is a group project with a maximum of 5 members.
2. Important Dates:
   Distribution Date: 14th October 2024 (Thursday)
   Submission Date: 29th December 2024 (Friday, 11.59 p.m.)
3. A recorded presentation will be submitted (Guidelines shall be provided)
4. The group leader will submit the following materials via elearn
   a. Report in .pdf
   b. Programme source file in either .py or .ipynb
   c. Presentation deck in .pdf
5. Please ensure that for both report and presentation deck submitted, each student name and ID are included. Also, do note that marks for the presentation may differ from each members based on their understanding and contribution to the project.