CSI 1320 — Final Project Report

Group Number: 33
Mark Dedvukaj G00913386

1. Introduction

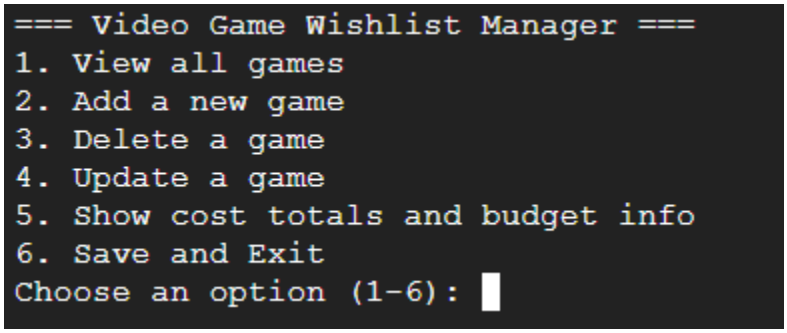
The purpose of my project is to manage a list of games. The problem it solves is keeping track of the games you own and want to buy in the future with their prices and how much they need to buy the games they do not own yet. This is useful in real life because people have large game libraries and want a simple way to organize the games they own and find out how much they need to purchase the games they want. This program uses a list of games from a text file. This allows the user to make changes through a menu, and it saves all updates you make when the program exits.

2. Methodology

My program loads data from a file, allows the user to add, delete, and modify game entries, and then saves all changes back into the file. I used modular structures like load_games() and save_games(). I used a list to store the games and a dictionary to represent each game. For user interaction I used input() and print() commands to allow the user to add, delete, update price or ownership, and to view the total cost of the game library. I used conditionals a lot in the update, delete, and budget functions. For this I used functions like “<, >, ==, and, or, not”. My code uses many loops, I use loops for displaying games, calculating totals, and a main program loop that keeps showing the main menu until the user decides to save and exit. For the data types portion I used data types like “str, float, int” for things like menu input, game prices, and totals. This program fully meets the requirement for adding, modifying, and deleting elements from a data structure because I used functions to add, delete, and update games. My code contains a custom function (calculate_total_cost(games, owned_only=False)) and it's used multiple times. My program reads and writes to games.txt, This satisfies the requirement of handling files containing at least 5 lines. The code includes comments that describe the purpose of major sections, how data is being processed, what each function does, and key operations such as loops, type conversions, and conditionals. My code is structured with spacing, indentation, and descriptive function names so it can be easy to read and follow.

3. Evaluation

Present and analyze the outputs produced by your code.



This is the main menu of my code

```
Choose an option (1-6): 1

--- Current Games ---
1. Forza Horizon 5 - $69.99 (Owned)
2. God of War - $49.99 (Owned)
3. NBA 2K26 - $59.99 (Wishlist)
4. Elden Ring - $59.99 (Wishlist)
5. Minecraft - $29.99 (Owned)
----------------------
```

When you choose option 1 it presents the current list of games you currently want and own.

```
=== Video Game Wishlist Manager ===
1. View all games
2. Add a new game
3. Delete a game
4. Update a game
5. Show cost totals and budget info
6. Save and Exit
Choose an option (1-6): 2
Enter the game title: Fortnite
Enter the price (e.g., 59.99): 0.00
Do you already own this game? (y/n): y
Game 'Fortnite' added.
```

When you choose option 2 you can add a new game with its price and if you own it.

```
=== Video Game Wishlist Manager ===
1. View all games
2. Add a new game
3. Delete a game
4. Update a game
5. Show cost totals and budget info
6. Save and Exit
Choose an option (1-6): 3
Enter the title of the game to delete: Fortnite
Game 'Fortnite' removed.
```

With option 3 you can delete a game from the list.

```
=== Video Game Wishlist Manager ===
1. View all games
2. Add a new game
3. Delete a game
4. Update a game
5. Show cost totals and budget info
6. Save and Exit
Choose an option (1-6): 4
Enter the title of the game to update: Minecraft
Current info: Minecraft - $29.99 - Owned
What would you like to update? (price/status/both): price
Enter the new price: 14.99
Game updated.
```

Option 4 allows you to change the price, status, or both of a game in the list.

```
=== Video Game Wishlist Manager ===
1. View all games
2. Add a new game
3. Delete a game
4. Update a game
5. Show cost totals and budget info
6. Save and Exit
Choose an option (1-6): 5
Enter your budget for games (e.g., 100): 150
Total cost of ALL games: $254.95
Total cost of WISHLIST (not owned) games: $119.98
Your budget: $150.00
Good news! You can afford all wishlist games within your budget.
```

Option 5 tells you how much your library of games is worth and it also shows the total cost of games you do not own. You can add your budget to see if you can buy all the games.

```
=== Video Game Wishlist Manager ===
1. View all games
2. Add a new game
3. Delete a game
4. Update a game
5. Show cost totals and budget info
6. Save and Exit
Choose an option (1-6): 6
Changes saved. Goodbye!
```

Option 6 saves your changes and exits the program.

4. Individual contributions

I did this project on my own.

My contributions include:

- Designing the layout and behavior of the program
- Writing all the code (files, menu system, functions, and data structures)
- Creating the games.txt file

- Testing all the menu options and making sure it gives the correct output
- Writing the project report and README file