COIS3380 Mini-Project Stage 2 – Pokemon Server

Prerequisites

Please review the lectures up to and including Week 10.

Download the csv file containing Pokemon descriptions.

Learning Objectives

- Convert a set of requirements into the program design and implement running code
- Make your stage 1 code responsive and performance via multi-threading (changes in functional behaviour are highlighted in ***bold italic***).

Learning Outcomes

1. You will learn how to perform file I/O and error handling in your C programs

Pokemon Server Program

During the course of our mini-project we will build a Pokemon Server. When finished, our Pokemon server will be a client-server application that allows gamers query pokemon properties by Pokemon type. The goal of the mini-project is to develop a computer program that helps explore the properties of Pokemon:

> *You are an avid gamer. You really like Pokemon games and trading cards. In order to improve your Pokemon gaming abilities you need to explore the properties of different Pokemon. A CSV file listing all Pokemon and their properties is available in the public domain. You would like to explore the file to learn more about different Pokemon and their properties. To do this you want to write a program that will allow you to look up Pokemon by name and store their properties in a separate file.*

We will use the iterative development approach to develop our Pokemon Server. During Stage 1 we will focus on developing

Legal Disclaimer

Pokemon and all related images, trademarks and copyrights are property of their rightful owners. This assignment is intended as a fun learning exercise and is not intended to infringe on any of these copyrights and trademarks. Please consult this site for legal obligations regarding Pokemon: https://www.pokemon.com/us/

Functional Requirements

Use Case 1 – Gamer starts the query program and opens the pokemon.csv file.

Actor: Gamer
Steps:

1. The Gamer starts the Pokemon query program.
2. The program prompts the Gamer to enter the name of the file containing the Pokemon descriptions.
3. The Gamer enters the name (or full directory path to, but we can assume the file is in the same directory) of the pokemon.csv file
    a. **Error flow:**
        i. If the file is not found the query program displays the message: "Pokemon file is not found. Please enter the name of the file again."
        ii. The query program prompts the Gamer to enter the name of the file again, or to exit the program.
4. The query program displays a menu with the following options:
    a. Type search
    b. Save results
    c. Exit the program

Use Case 2 – Search for Pokemon by type (Type 1).

Actor: Gamer
Steps:

1. The Gamer selects option 1 on the menu – Type search.
2. The query program searches all Pokemon data in pokemon.csv, finds those Pokemon records where Type 1 matches the Gamer's input, and stores these in memory.
    a. ***The query program does not wait for the query to complete and immediately returns to the main menu.***
    b. ***While the Gamer selects new choices in the main menu, the query runs in the background.***
    c. ***The Gamer can start a new query while the previous query(ies) run in the background.***
    d. ***Error flow:***

        i. ***If the query program is not able to start the query in the background the query program displays the message: “Unable to start the query!”.***
        ii. ***The query program exits.***

Use Case 3 – Save the accumulated query results.

Actor: Gamer
Steps:

1. The Gamer selects option 2 from the menu – Save results.
2. The query program prompts the Gamer to enter the name of file which will contain the query results accumulated to this point.
3. ***The query program waits for all currently running queries to complete prior to saving the accumulated query results in the file.***
4. The query program saves the results of all completed queries in this file.
    a. **Error flow:**
        i. If the query program is not able to create the new file it displays the message: “Unable to create the new file. Please enter the name of the file again.”
        ii. The query program prompts the Gamer to either enter the name of the file again.
    b. The query program does not include any data generated by queries that are still in progress.
    c. The save operation runs in the background, and the query program immediately displays the menu described in Use Case 1.

Use Case 3 – Exit the query program.

Actor: Gamer
Steps:

1. The Gamer selects option 4 from the menu – Exit the query program.
2. ***The query program terminates any queries that are currently in progress.***
3. The query program displays the total number of queries completed successfully during the session.
4. The query program displays the names of new files created during the session.
5. The query program exits.

Deliverables

This assignment will have two deliverables:

1. **Readme File.**
    a. Submitted in text format.
    b. It should explain how to build and run your program.

   c. It should also explain the key program design and implementation choices to supplement the comments in your code.

2. **C code.**
   a. C and header files that implement the program design.
   b. For full marks the code should:
      i. Run without errors.
      ii. Implement the functional and non-functional above - correctly.
      iii. Should include comments that explain key implementation choices, and follow the ANSI C style.

3. **Guidellines**.
   a. Aim to complete coding for this stage within 3 days.
   b. If you work as a team, try working on your code together (pair programming).
   c. Submit your code and readme when ready.
   d. Organize your code nicely into functions, to make it easier to modify it during Stages 3.
   e. Think about how you will dynamically allocate the memory to hold the results of your queries.
   f. Only the final deliverable will be graded, but if you are not able to complete all stages they will be able to get partial marks for code that you submitted.