Problems

1. Problem Statement: Network Data Packet Processing

Consider a scenario where a program is tasked with processing various types of data packets received from a network. These data packets originate from different sources and contain diverse types of information, including integers, floating-point numbers, and strings.

To optimize memory usage and streamline the processing workflow, the program aims to implement a unified data structure for storing and handling these packets.

Create a vector of packets and 3 functions:

void add_int(PACKET *packets, int value);
void add_float(PACKET *packets, float value);
void add_string(PACKET *packets, const char* value);

2. Write a program to perform the following tasks:

Create a text file named "numbers.txt" containing numbers from 1 to 10, each number on a new line.

- Read the numbers from the file and calculate their sum.
- Write the sum to a binary file named "sum.bin".

Ensure to handle file opening, reading, writing, and closing operations properly.

3. Employee Management

Define a struct named Employee with fields for name, age, and salary. Write a program to create an array of Employee structs, populate it with data from a text file, and print out the details of each employee.

example.txt

John Doe 30 50000
Alice Smith 25 60000
Bob Johnson 35 70000

Try to make it as space efficient as possible.

4. Galactic Gastronomy: Intergalactic Chef's Challenge

Welcome to the culinary capital of the cosmos, where taste buds are as diverse as the stars themselves! As the Intergalactic Chef Extraordinaire, your mission is to concoct otherworldly delicacies that will tantalize the alien palates of the galaxy's most discerning diners.

- Ingredient Inventory:

Your pantry is stocked with an array of exotic ingredients sourced from the far reaches of the universe. Define a struct named GalacticIngredient with fields for ingredient type (Stardust Spice, Celestial Citrus, Nebula Nectar, Lunar Lotus Root, Comet Cumin) and flavour intensity (1 – 5). Populate your ingredient inventory with these cosmic treasures.

- Recipe Revelation:

Every dish tells a story, and yours are no exception. Create a struct named CosmicRecipe (Asteroid Alfredo, Meteorite Meatballs, Black Hole Burgers, Space Soup) to encapsulate the secrets of your culinary creations. Utilize bitfields to specify cooking methods (Grilling, Baking, Frying, Boiling) and seasoning preferences for each recipe (An array of ingredients), ensuring that each bite is a voyage through flavour space. Let your imagination run wild!

Grilling and baking require a temperature, boiling requires a water amount and frying an oil amount. Also, every recipe has a duration.

- Cosmic Cookbook Conversion:

Delve into the Cosmic Cookbook, a compendium of recipes gathered from galaxies far and wide. Access this binary file treasure trove and transmute its contents into binary brilliance for swift retrieval during the heat of culinary combat. Write all of your splendid recipes and ingredients in a binary format.

- User Menu

Create a user interface which allows the user to add a new ingredient to the pantry, a recipe to the recipe book and the option to write to a file or read from a file data. While reading or writing print out information about what is being written or read.

5. Character customization

How would you create a structure for storing options for a video game character's customization, including hair style, hair colour, clothing style, and weapon preference? Add any other fields that you think would make sense.

6. Bitmap Grayscale Conversion

Convert a bitmap image into grayscale. In the bitmap case each pixel is represented by 3 bytes. One for red, one for green, one for blue, each from 0 to 255 (inclusive). We can use the formula P_gray = 0.299 * P_red + 0.587 * P_green + 0.114 * P_blue to convert to grayscale.

Bitmap file header – Link here

The data that needs to be changed is in the Raster Data.

The number of pixels is height x width (Found in the header)

To change the colour, you will need to overwrite P_red, P_green, P_blue with P_gray

Example bitmap will be provided.

7. Custom binary data

Design and implement a binary file format for efficient storage of various types of data structures, such as graphs, matrices, and other complex data representations (come up with anything you want). The primary goal is to develop a standardized format that maximizes storage efficiency, allows for read and write operations.

Provide comprehensive documentation and usage guidelines for developers (aka how the file looks like).