CSE102 – Computer Programming

Homework #12

Dynamic Arrays

Due Date: 27/06/2023

Hand in: A student with number 20220000001 should hand in a zip file named 20220000001.zip for this homework.

Homework Description: In this assignment, you will generate a linked list that contains items of four different structure types. The linked list will then be serialized and written into a binary file. Lastly, you must reverse engineer the binary file to generate the linked list back.

Important: Please ensure that you adhere to the given requirements, which include using a void pointer function to fill the linked list and not relying on any tips or tricks in the binary file to indicate the order of the items.

Part 1. [30 pts] Generate and Fill the Linked List

Implement a function **void fillLinkedList(struct Node** head)** that generates a linked list with four different types of assets and fills them with appropriate values. The struct definitions are as follows:

```
typedef struct Node {
    void* data;
    struct Node* next;
} Node;
typedef struct {
    char type[20];
    int ivals[1];
    double svals[1];
} Asset1;
typedef struct {
    char type[20];
    double svals[2];
```

```
    int ivals[2];
}Asset2;
typedef struct {
    char type[20];
    char string1[50];
    char string2[50];
}Asset3;


Typedef struct {
    char type[20];
    double value1;
    float value2;
    double value3;
}Asset4;
```

Write a function that creates a randomly determined number of assets, between 10 to 20 (randomly determined on each call), and connect them in a linked list. The generated linked list should be stored in the provided 'head' parameter.

Part 2. [30 pts] Serialize and Write the Linked List

Implement a function **void serializeLinkedList(struct Node* head)** that serializes the linked list and writes it to a binary file named linkedlist.bin. The serialization should include the type of each asset (as a fixed-length string) and the corresponding data for each asset.

Part 3. [40 pts] Read and Reconstruct the Linked List

Implement a function **void deserializeLinkedList(struct Node** head)** that reads the binary file "linkedlist.bin" and reconstructs the linked list. The function should read the serialized data from the file, create the appropriate elements based on the type information, and connect them to form the linked list. The reconstructed linked list should be stored in the provided **head** parameter.

Notes:

** You are required to create a demo video showcasing the random generation of assets and their writing into a binary file, without revealing the apparent order of asset types in the list. The video should also include the deserialization process and the assets in the reconstructed list. Please upload the videos to YouTube with unlisted visibility and share the link by including it in a text file within the homework document.

****Do not forget to prepare a makefile (-50 points)**

General Rules:

1. Make sure to include appropriate comments and variable names in your code to make it easy to understand.
2. The program must be developed on given version of OS and must be compiled with GCC compiler, any problem which rises due to using another OS or compiler won't be tolerated.
3. Note that if any part of your program is not working as expected, then you can get zero from the related part, even it is working partially.
4. Zip your homework files before uploading them to MS Teams. The zip file must contain the C file with your solution and screenshots of the valid outputs of the program.
5. You can ask any question about the homework by sending an email to b.koca@gtu.edu.tr or by using the homework channel on MS Teams page of the course.