Project task

Create a "stack" data structure handler.

Requirements:

1. The project must be done in Microsoft Visual Studio
2. The project must include error and message handling.
3. The project must include stack handling, specifically:

    a. Initializing the stack
    b. Stack release
    c. Adding a new element
    d. Retrieving the first element
    e. Finding the specified item[1]
    f. Write all stack elements to disk in a binary file
    g. Read from disk

4. The handling of the stack should not depend on the type of data, placed in the stack. Use a void* pointer for this.
5. Combining elements into a stack should be done through pointers (Do not use a stack, created from an array!).
6. The design must include an interface. Each procedure from the stack handler should be called from the corresponding interface function.
7. In addition, the interface must output stack and menu items to the monitor.
8. The data must be presented in a structured form and include handling functions such as:

    a. Write/read a single object to/from disk
    b. Output to monitor
    c. Initializing the object
    d. Memory release

    e. Comparison functions according to the specified criterion (for the function of finding the specified item)

9. Each component of the project should be represented as a separate pair of files (.cpp, .h), with data encapsulated in the file and used in other files only through functions.
10. All functions for each component of the project should be separated into external functions (will be used in other files) and internal functions (only in the file that handles the data). Internal functions should be encapsulated in their respective files.
11. Create a sample MY_STUDENT data structure that contains:

    a. student's name (string)
    b. year of birth (int)
    c. field of study (an enum element that points to an array of strings with the names of each field of study).

    "student's name" string should be allocated dynamically with a size, corresponding to the number of characters in the name.

---

[1] The comparison criterion for determining whether the specified element has been found is specified by the function passed through the pointer