CSCE 240 – Advanced Programming Techniques

Project 3

Pointer "ArrayList"

25 Points

Assigned on: September 17th, 2019
Due: October 1st, 2019 @ 11:59 pm

Topics Covered:

- Master the use of functions in C++.
- Utilize call-by-value and call-by-reference implementation of functions.
- Demonstrate mastery of pointer arrays.
- Become familiar with dynamic memory allocation using pointers.
- Basic use of header file.

Description:

Develop a program that mimics some of the functionalities of an ArrayList in Java. Your program should maintain a pointer array of doubles and be able to perform the following functions:

1. `void insert(int index, double num, double *&arr, int &size)`
    a. Adds an element (`num`) to the array (`arr`) at a given position (`index`) and updates the size.
    b. You may allow the user to add to the immediate end of the array (at position n for an array of n elements) but not past. You should print an error message if they try to print beyond these bounds.
2. `void remove(int index, double *&arr, int &size)`
    a. Removes an element from the array (`arr`) at a given position (`index`) and updates the size.
    b. If index is out of the bounds of the arr then an error message should be printed.
3. `int get(int index, double *arr, int size)`
    a. Returns the element at the given position (`index`).
    b. Should check if the index given is outside the bounds of the array. If it is out of bounds an error message should be printed.
4. `void clear(double *&arr, int &size)`
    a. Clears all elements of the array (`arr`) and updates the size (`size`) to be 0.
5. `int find(double num, double *arr, int size)`
    a. Returns the first index in which a given element (`num`) is found in the array (`arr`). If not found -1 is returned.
6. bool equals(double *arr1, int size1, double *arr2, int size2)
    a. Returns true if the contents of the two arrays are equal and false if they are not equal.
7. `void init(double *arr, int size)`

    a. Populates the elements of the array (`arr`) with input from the user (or via file redirection).
8. `void print(double *arr, int size)`
    a. Prints the elements of the array.

Additional Specifications:

- I have provided two files: sampleMain.cpp and myArray.h
    - I will be using sampleMain.cpp to test your program so you should test using that.
    - In myArray.h you should write ALL YOUR CODE. This is the file that you will be required to submit. DO NOT RENAME THIS FILE!
- DO NOT USE GLOBAL VARIABLES.
- You should check your code with Valgrind for memory leaks. You will get points off for both memory leaks as well as errors reported in Valgrind.
- Your program should consist of a header that contains the following information:
    - First name and last name of the programmer.
    - Date and time of the program completion.
    - A brief description of the program function.
    - Input requirements and format.
    - The output of the program.
    - Any additional needed comments (e.g. related to compilation, execution or other requirements).
- Each function needs to be properly commented.
    - Your comments need to include a description of the function.
    - Description of the inputs.
    - Description of the output.
    - Any additional notes assisting future programmers to comprehend the complex portions of your functions.
- Make sure your program compiles and runs on one of the Linux machines in the Linux lab before you submit.
- **Submit the source code (.h) not the executable (a.out).**

DO NOT CHANGE THE NAME OF YOUR FILE! IT SHOULD BE NAMED “myArray.h”

Example Output of sampleMain.cpp:

```
casey@thinki:~/Dropbox/Teaching/CSCE240Fall2019/hws/Assignment3$ g++ Assignment3_Cole.cpp
casey@thinki:~/Dropbox/Teaching/CSCE240Fall2019/hws/Assignment3$ ./a.out
Please enter the size of your array: 5
Please enter 5 elements to populate the array.
1
2
3
4
5
The original array: 1 2 3 4 5
Size: 5
Index is not within the bounds of 0...n.
The new array: 1 9.3 2 3 4 5
Size: 6
Index is not within the bounds of 0...n--1.
The new array: 1 9.3 2 3 4
Size: 5
The element at position 2 is: 2
The element at position 10 is: -1
9.3 was found at position: 1
999 was not found.
0 0 0 0 0
The arr and arr2 are not equal.
The arr and arr are equal.
casey@thinki:~/Dropbox/Teaching/CSCE240Fall2019/hws/Assignment3$
```