COMP 3000: Assignment 4
Due: 11/17/2015 at 11:59 PM
Points: 100

Note: You do not need to submit hard copies.

Goals:

1. Learn to use pointers.
2. Learn to create your own data structure.
3. Create a library class that can be used by other programmers.

Write a program to create a class IntegerArrayList, which provides all the functionality on the array. This class should provide the functionality to create and maintain a list of integers using array as the base data structure. The structure of the class should be as follows:

Member Variables:

intArray	This is the array containing the list of integers.
capacity	It holds the capacity of the array list.
count	It keeps the count of the occupied locations in the array.

Constructors:

IntegerArrayList() Constructs an empty list of initial capacity of 10.
IntegerArrayList(int capacity) Constructs an empty list of initial capacity of *capacity.*

Functions:

int	***add(int item)*** adds the item to the next available location, returns the current count of the elements in the array list.
int	***remove(int location)*** removes the item located at location. Returns the removed item.
bool	***contains(int item)*** returns true if the list contains item, else returns false.
int	***get(int location)*** returns the item located at *location.* If the location is greater than *count. Returns -1*.
bool	***isFull()*** returns true if the list is full, i.e. if capacity== count.
bool	***isEmpty()*** returns true if the array list contains no element.
void	***printList()*** prints all the elements in the list.
bool	***set(int location, int item)*** replaces the item located at *location* with *item.* Returns true if the operation is successful, else returns false.

You have to create a file IntegerArray.cpp, which contains the implementation of this class. I will be having a unit test cases (a test driver) which will call all the functions in your class. Your program should satisfy all the test cases in the test driver.

Requirements:

1. (5 points) use comments to provide a heading containing your name, Auburn userid, and filename. Also, describe any help and sources you have used to find the solution.
2. (5 points) your program should successfully implement the default constructor.
3. (5 points) your program should successfully implement parameterized constructor.
4. (10 points) your program should implement add() function, which takes an integer itme as an argument and adds it to the list if there is any capacity left.
5. (5 points) your program should implement get function which returns the item located at the *location.* If the location is greater than the *count*, your program should display appropriate message to the user.
6. (15 points) your program should implement *remove()* function, which removes the item located at *location* and shifts all the item after the location to the left in order to fill the gap.
7. (5 points) your program should implement *contains()* function which returns true if the list contains the *item,* else returns false.
8. (5 points) your program should successfully implement *get()* function, which returns the item location at *location.* If the *location* is greater than the capacity or count shows appropriate message to the user and returns -1.
9. (15 points) your program should successfully implement *isFull() and isEmpty()* functions.
10. (5 points) your program should successfully implement *printList()* function, which displays entire list. Also, displays appropriate message if the list is empty.
11. (15 points) your program should successfully implement *set()* function which takes two argument- *location and item*- to replace the item at *location* cab be replaced by *item.* If the location is out of the capacity or after the count the function displays appropriate message to the user and returns false.
12. (5 points) usability of your program.
13. (5 points) readability of your source code.

Deliverables:

- Submit your source code file named as "IntegerArrayList.cpp" through Canvas.

Late Submission Penalty:

- 10-point penalty per day for late submission. For example, an assignment submitted after the deadline but up to 1 day (24 hours) late can achieve a maximum of 90% of points allocated for the assignment. An assignment submitted after the deadline but up to 2 days (48 hours) late can achieve a maximum of 80% of points allocated for the assignment.
- Assignment submitted more than 3 days (72 hours) after the deadline will not be graded.

Rebuttal period:

- You will be given a period of 7 days to read and respond to the comments and grades of your homework or project assignment. The TA may use this opportunity to address any concern and question you have. The TA also may ask for additional information from you regarding your homework or project.