

Data Structure and Algorithm Analysis

COP3530

Program – Unit 3

This assignment will access your skills using C++ strings and dynamic arrays. After completing this assignment you will be able to do the following:

(1) allocate memory dynamically (2) implement a default constructor (3) insert and remove an item from a sorted dynamic array of strings, (4) use the string class member functions, and (5) implement a destructor

Assignment Description:

Call your driver "dynamic_array_driver.cpp", your class implementation file "tlist.cpp", and your class declaration, "tlist.h". Define the following behavior for tlist.

Implement a default constructor which initializes the state to the following: count = 0, capacity = 12, and allocates memory the size of string[capacity]. Read the contents of the file "myData.txt". The contents of the files is given below.

1. Include the following message, "**Default Constructor Invoked**" every time the function is called.
2. Implement the **destructor**. Include the following message, "Destructor Invoked" every time the function is called.
3. Implement **Is_Full()** which returns true if full; otherwise false; Include the message, "Is_Full Invoked" every time the function is called
4. Implement **Is_Empty()** which returns true if empty; otherwise false; include the message, "Is_Empty Invoked" every time the function is called.
5. Implement the member function called "**Search**" to search the dynamic array for an item. The function should print the message, "**Item Found**" or "**Item Not Found**". Which message is print depends on if the item was found or not found in the list. Print the item (search key) you were looking for. Include the following message, "**Search Invoked**" every time the function is called.
6. Implement a function called "**Insert**" to add an item to the dynamic array in order. The function should print the contents of the tlist object before and

after the function has been executed on single separate lines. Include the following message, "Insert Invoked" every time the function is called.

7. Implement a function called "**Remove**" to remove an item from the dynamic array. The function should print the contents of the tlist object before and after the function has been executed on single separate lines. Include the following message, "**Remove Invoked**" every time the function is invoked.
8. Implement a function called "**Display**" that displays the contents of the array.

Your program should COMPLETELY TEST ALL the operations of the class. For this assignment, put the class declaration in the implementation file "tlist.h" and the implementation file in "tlist.cpp". Use the class declaration below to help you implement the class tlist.

Class Declaration:

```
class tlist
{
public:
    //tlist();           //default constructor sets the following: count = 0, capacity =
    //12, and allocates memory the size of string[capacity]

    //~tlist();          //destructor

    //bool IsEmpty();     //return true if empty; otherwise false

    //bool IsFull();      //return true if full; otherwise false

    //int Search(const string &); //returns the location of the string in the dynamic array

    //void Insert(const string & key); //add key to dynamic array if not full; otherwise
    //doubles the size of the array and adds key

    //void Remove(const string & key); //removes key from dynamic array if it is there;
    //otherwise prints a message stating it was not in dynamic array

    //void Display(); //displays the contents of the dynamic array

    //other functions may be implemented if necessary

private:
    // string *DB; //dynamic array

    //int count; //number of strings stored in the dynamic array
```

```
//int capacity; //total number of cells in the dynamic array  
//additional state variables you may wish add  
};
```

Contents for the file “myData.txt”:

```
defghiiijjiiijj  
ab  
stringtrist  
abczzzabcabc  
cdeerxre  
abcdeabaeceabrxre  
cveevcyxq  
pppppreeeeppppp  
kjhlfgdsaw  
fertyxcwt
```

For this assignment, put the class declaration in the implementation file "tlist.h" and the implementation file in "tlist.cpp". Called the driver "dynamic_array_driver.cpp". Put all files (tlist.h, tlist.cpp, and dynamic_array_driver.cpp) in a zip file called "asgn3", and submit to Blackboard before the due date and time in the Assignments area.