Bert Darnell

Kenneth Dewey

CS225

04/30/23

Unit 7- HW11

1. *Can we declare an object of arrayListType? If yes, what member/non-member functions can this object be used with? If no, why not?*

Yes, we can declare an object of ‘arrayListType’. The functions we can access are isEmpty(), isFull(), listSize(), macListSize(), isItemAtEqual(int, int), removeAt(int), retrieveAt(int, int&), clearList(), seqSearch(int), insertAt(int, int), insertEnd(int), replaceAt(int, int), remove(int).

1. *Can we declare an object of unorderedArrayListType? If yes, what member/non-member functions can this object be used with? If no, why not?*

Yes, we can declare an object of ‘unorderedArrayListType’. The functions we can access include all the public/protected members of the object ‘arrayListType’ which is the functions from the previous question, and also the new overided members, insertAt(int, int), insertEnd(int), replaceAt(int,int), seqSearch(int), remove(int).

1. *For the given main, identify which member/non-member functions of which class will be invoked. Include constructor (specify which constructor if there is more than one constructor in a class) and destructor in your answer.*

The main function creates an object of type ‘unorderedArrayListType intList(25)’ inserts some elements into it ‘intList.insertEnd(int)’ then creates a second object ‘unorderedArrayListType temp(intList)’ and initializes it with the first object, replaces an element in the second object ‘temp.replaceAt(int)’ and then prints both objects ‘std::ostream& operator<<(std::ostream&, const arrayListTYpe&)’.

1. Is it okay to remove the isEmpty() function from queue class and let users use the inherited isEmpty()? That’s what the unorderedArrayListType class does. Explain.

Yes, it is okay to remove the ‘isEmpty()’ function from the queue class and instead inherited ‘isEmpty()’ function from the base class. This is used in the ‘unorderedArrayListType’ class because the ‘isEmpty()’ function is already implemented in the base class

Text

Description automatically generated