EE3093 Tutorial: C++ week 5

Instructions to users

This tutorial can be completed up to 3 degrees of complexity (one for each section); make sure you have understood and completed a section (i.e. code implemented and tested successfully) before moving on to the next one.

Part 1: Basic

Follow the instructions given in the second Tutorial and create (in a new directory) a new project (e.g. TUT5).

Implement a class called IntArray; an instance of this class is an array that can accommodate MAX_SIZE integer items (start with MAX_SIZE=50).

Write and include in your project the file(s) implementing the (public) member functions described below:

IntArray(); // constructor; upon construction the array is "empty".

//If position "pos" is a valid position and corresponding item is "empty", return true; false otherwise;
bool **IsItemEmpty**(int pos);

//Check that position "pos" is a valid position for this array; if so return true; false otherwise;
bool **checkPos**(int pos);

//Insert "val" in position "pos" if this is a valid position and
// if the corresponding item is "empty", then return true; false otherwise;
bool **InsertItem**(int val, int pos);

//Pint to screen position and value of all "non-empty" elements in the array.
void **printArrayContent**();

Test your implementation: write & run a test routine analogous to the ones shown in lectures over past weeks.

Part 2: Intermediate

For class IntArray, implement the following (public) member functions:

//The number of "non-empty" items in the array (inserted so far by the user);
int **getTotValidItems**();

//If a "non-empty" item is found in position "pos", its value is written to "val",
// the item becomes "empty" and true is returned; false otherwise;
bool **removeItemVal** (int pos, int& val);

//If position "pos" is a valid position for this array, reset the corresponding item (so that it becomes "empty");
bool **resetItem** (int pos);

Test your implementation: write & run a test routine analogous to the ones shown in lectures over past weeks.

Part 3: Advanced

For class IntArray, implement the following (public) member function:

// sort all valid items (ascendingly or descendingly); any "empty" item is placed at the end of the array

sortArray (bool ascending);

Test your implementation: write & run a test routine analogous to the ones shown in lectures over past weeks.