**Homework 7. Implement an ADT consisting of a singly linked list and a dynamic array of integers**

**Main goal:**  Use pointers to implement a dynamic array and a singly linked list.  You will encapsulate both in an ADT.  Specifically, your ADT will include the following data members:

* int \*array;      //dynamic array
* int size\_array; //number of elements stored in the dynamic array
* int capacity;   //the current number of integer spaces allocated to the dynamic array
* struct Node {

                   int value;

                   Node \*next

          };

* Node \*head; //point to the first node on the linked list
* int size\_linkedList;  //number of nodes on the linked list

Your ADT will also include the following member functions:

* a default constructor to initialize the array and linked list correspondingly
* the "big-3": destructor, copy constructor and overloaded assignment operator
* void push\_back(int val ); This member function inserts the value 'val' to the end of the dynamic array and the end of the linked list. Note that this function will require memory allocation (i.e., need to call the *new* operator).
* void pop\_back();  This member function deletes the last number from the array, and the last number from the linked list.  Note that this function will  require memory deallocation, i,e., need to call the *delete* operator.
* an overloaded put operator (<<) to print out all the data items stored on the dynamic array and the linked list. Note that you are recommended to overload this operator as a friend function of your ADT.

Make sure you test all these above functions in the main() function. Separate compilation is required.

**Total points**:   70 points (10 points per function).

**Submission:** Archive your header file and the two .cpp files into one zip file and submit this zip file