**CSC 330  HW2**  **Overloaded Operators in C++**

Given the following class definition:

```
class Array {
public:
   Array( int = 10 );  // default constructor
   ~Array();              // destructor
private:
   int size;    // number of elements in the Array
   int *ptr;   // address of dynamically allocated
memory
};
```

Write a member function that overloads the += operator to merge two sorted Arrays and store the result in the first Array.  You can assume that both of the operands are sorted, and that there are no duplicated values.
For example, if array1 contains
```
   1, 2, 5, 9, 12, 15
```
and array2 contains
```
   3, 4, 7, 13, 19, 21, 25
```
then after execution of the statement
```
   array1 += array2
```
array2 will be unchanged, and array1 will contain
```
   1, 2, 3, 4, 5, 7, 9, 12, 13, 15, 19, 21, 25
```

Your function must not call any other functions.  The class does not include a copy constructor, overloaded assignment operator, or any other overloaded operators. Note; although the [ ] operator is not overloaded for Array objects, you can still use the [ ] operator with int pointers.
The efficiency of your function will be considered.  This means that for full marks you must use the correct method for merging two already-sorted arrays.  You will receive only 80 instead of 100 if you correctly use a method that creates a non-sorted result and then sorts it.

**Posted: March  9**
**Due: March 17**